

Open Geospatial Consortium

Publication Date: 2019-01-14

Approval Date: 2014-12-01

Submission Date: 2014-11-13

External Identifier: <http://www.opengis.net/doc/IS/MovingFeatures/xmlcore/1.0.2/>

URL for the normative version: <http://docs.opengeospatial.org/is/18-075/18-075.html>

Reference number of this OGC® project document: OGC18-075

Version: 1.0.2

Category: OGC® Implementation Standard

Editors: Akinori Asahara, Ryosuke Shibasaki, Nobuhiro Ishimaru, David Burggraf

OGC® Moving Features Encoding Part I: XML Core

Copyright notice

Copyright © 2015 Open Geospatial Consortium

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is an OGC Member approved international standard. This document is available on a royalty free, non-discriminatory basis. Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: OGC® Approved Standard
Document subtype: Encoding
Document stage: Approved for Public Release
Document language: English

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it. None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable

Contents

i.	Abstract.....	v
ii.	Keywords	v
iii.	Submitting organizations	v
iv.	Submission contact points	v
v.	Changes to the OGC® Abstract Specification	vi
	Foreword.....	vii
	Introduction.....	viii
1	Scope.....	1
2	Conformance	1
3	Normative references.....	1
4	Terms and definitions	2
5	Conventions	4
5.1	Symbols (and abbreviated terms).....	4
5.2	UML Notation	4
6	Data Model Overview	7
6.1	Concepts and standard modularity	7
6.2	Foliation and prisms	7
6.3	XML Implementation.....	10
6.4	Time ordered encoding.....	11
7	Root Element	11
7.1	mf:MovingFeatures.....	11
7.2	mf:stBoundedBy.....	13
7.3	mf:member	14
7.4	mf:MovingFeature	14
8.	Header data	15
8.1	mf:Header.....	15
8.2	mf:VaryingAttrDefs	15
8.3	mf:AttrDef	16
8.4	mf:AttrAnnotation.....	18
8.5	mf:Hints	18
8.6	mf:Hint.....	19
9.	Foliations.....	20
9.1	mf:Foliation.....	20
9.2	mf:AbstractTrajectory	21
9.3	mf:LinearTrajectory	22
9.5	mf:Attr.....	23
10	Example instances	24
10.1	Example 1 Pedestrian’s tracks.....	24

10.2 Example 2 Vehicles	25
Annex A (Normative) Abstract test suite.....	28
A.1 Conformance Class:.....	28
A.1.1 XML Validity.....	28
A.1.2 Root Element	29
A.1.3 Attribute Definition.....	29
A.1.3 Time order	29
A.1.4 Hints	30
A.1.5 sTBoundedBy	30
Annex B (Informative) Application examples.....	32
B.1 Transportation survey with smart phones	32
B.2 Maritime sector use case	33
B.3 Aviation sector use case	35
B.4 A Use case for Hurricanes.....	38
B.5 Predicting Crime patterns by simulating movement of criminals, victims and police on map	40
B.6 Use-Cases for Soccer Matches	42
Annex C (Informative) XML Schema for MovingFeatures.....	45
Annex D Bibliography	46
Annex E (Informative) Revision History	47

i. Abstract

This OGC® Standard specifies standard encoding representations of movement of geographic features. The primary use case is information exchange.

ii. Keywords

ogcdoc, ogc document, moving features, gml

iii. Submitting organizations

The following organizations submitted this Implementation Standard to the Open Geospatial Consortium

- a) Central Research Laboratory, Hitachi Ltd.
- b) Center for Spatial Information Science, The University of Tokyo
- c) Defense Systems Company, Hitachi, Ltd.

iv. Submission contact points

All questions regarding this submission should be directed to the editor or the submitters:

CONTACT	COMPANY
Akinori Asahara	Central Research Laboratory, Hitachi, Ltd.
Ryosuke Shibasaki	Center for Spatial Information Science, The University of Tokyo
Nobuhiro Ishimaru	Defense Systems Company, Hitachi, Ltd.
Hiroshi Kanasugi	Center for Spatial Information Science, The University of Tokyo
David Burggraf	Individual associate member
Gianluca Luraschi	EMSA (European Maritime Safety Agency)
Frank Suykens	Luciad
Chris Little	UK Met Office
Nadeem Anjum	Google Summer of Code representative through Apache

Martin Desruisseaux	GEOMATYS
Ki-Joune Li	Pusan National University

v. Changes to the OGC® Abstract Specification

The OGC® Abstract Specification does not require changes to accommodate this OGC® standard.

Foreword

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights. However, to date, no such rights have been claimed or identified.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

OGC Moving Features consists of the following parts under the general title *Moving Features Encoding*:

- *Part I: XML Core*
- *Part II: Simple CSV*

and the following specification:

- *Moving Features Access*

Introduction

Applications using **moving feature** data, typically representing vehicles or pedestrians, are rapidly increasing. Innovative applications are expected to require the overlay and integration of moving feature data from different sources to create greater social and business value. Examples of applications that require integrated simulation are disaster risk management, traffic information services, security services, navigation for robots, aviation or maritime traffic monitoring, and wildlife tracking and conservation. Moreover, systems relying on single-source moving feature data are now evolving into more integrated systems. Integration of moving feature data from different sources is a key to developing more innovative and advanced applications.

Efforts in this direction therefore should be encouraged by ensuring smoother data exchange because handling and integrating moving feature data will broaden the market for geo-spatial information such as Geospatial Big Data Analysis. ISO 19141:2008 is an existing abstract standard to model moving features; however encoding method is not provided.

Therefore, an XML (and GML) style encoding for Moving Features is standardized. The defined encoding standard is *ISO 19141:2008 Geographic Information – Schema for Moving Features* compliant implementation standard applicable for massive data handling. ISO 19141:2008 defines a method to describe the geometry of a feature that moves as a rigid body.

OGC® Moving Feature Encoding Part I: XML Core

1 Scope

This OGC™ Standard specifies an encoding format to describe the geometries of many features that move as rigid bodies. This OGC® Standard is applicable to features having the following characteristics.

- a) Each moving feature can be described with Schema for Moving Features (ISO19141: 2008).
- b) The number of features simultaneously encoded with this format can be massive (many thousands of features).
- c) All features can be described using common space-time coordinates.

This standard does not address all types of moving features. Examples of features that are out of scope include the following:

- Deforming features, such as flood water.
- Feature’s geometric representation should not be encoded inline in a geometric complex that contains the geometric representations of other features, since this would require the other features’ representations to be updated as the feature moves.

This standard is focused on the encoding format, thus Web interface and inertial data models are out of scope also.

2 Conformance

Conformance with this specification shall be checked using all the relevant tests specified in Annex A (normative).

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of OGC Moving Features. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of OGC Moving Features are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

[OGC 07-036] OGC 07-036, OpenGIS® Geography Markup Language (GML) Encoding Standard

[ISO 19141:2008] ISO 19141:2008, Geographic information – Schema for moving features

[ISO 19101:2014] ISO 19101:2014, Geographic information -- Reference model -- Part 1: Fundamentals

[IETF RFC 2396] IETF RFC 2396, Uniform Resource Identifiers (URI): Generic Syntax. (August 1998)

[W3C XML] W3C XML, Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation (4 February 2004)

[W3C XML Namespaces] W3C XML Namespaces, Namespaces in XML, W3C Recommendation (14 January 1999)

[W3C XML Schema Part 1] W3C XML Schema Part 1, XML Schema Part 1: Structures, W3C Recommendation (2 May 2001)

[W3C XML Schema Part 2] W3C XML Schema Part 2, XML Schema Part 2: Datatypes, W3C Recommendation (2 May 2001)

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1

Moving feature

representation, using a local origin and local ordinate **vectors**, of a **geometric object** at a given reference time

[ISO 19141:2008]

4.2

Foliation

one parameter set of geometries such that each point in the **prism** of the set is in one and only one **trajectory** and in one and only one **leaf**

[ISO 19141:2008]

4.3

geometric object

spatial object representing a geometric set

[ISO 19107:2003]

4.4

leaf

<one parameter set of geometries>

geometry at a particular value of the parameter

[ISO 19141:2008]

4.5

trajectory

path of a moving point described by a **one parameter set of points**

[ISO 19141:2008]

4.6

one parameter set of geometries

function f from an interval $t \in [a, b]$ such that $f(t)$ is a geometry and for each **point** $P \in f(a)$ there is a one parameter set of points (called the trajectory of P) $P(t) : [a, b] \rightarrow P(t)$ such that $P(t) \in f(t)$

[ISO 19141:2008]

4.7

prism

<one parameter set of geometries>

set of **points** in the union of the geometries (or the union of the **trajectories**) of a **one parameter set of geometries**

[ISO 19141:2008]

4.8

period

one-dimensional **geometric primitive** representing extent in time

[ISO 19141:2008]

4.9

point

0-dimensional **geometric primitive**, representing a position

[ISO 19107:2003]

4.10

geometric primitive

geometric object representing a single, connected, homogeneous element of space

[ISO 19107:2003]

4.11

vector

quantity having direction as well as magnitude

[ISO 19123:2005]

5 Conventions

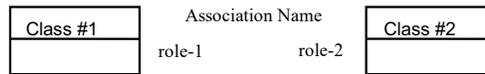
5.1 Symbols (and abbreviated terms)

ISO	International Organization for Standardization
OGC	Open Geospatial Consortium
UML	Unified Modeling Language
XML	Extensible Markup Language
1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
CSV	Comma Separated Values

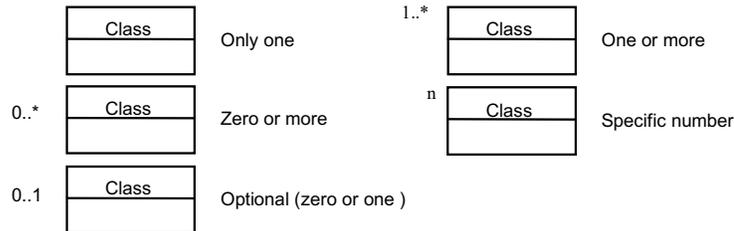
5.2 UML Notation

The diagrams that appear in this standard are presented using the Unified Modeling Language (UML) static structure diagram. The UML notations used in this standard are described in the diagram below.

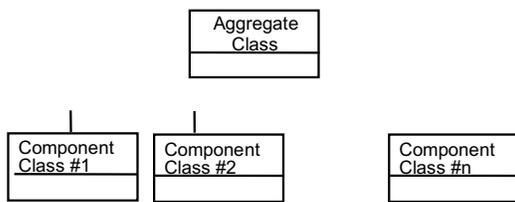
Association between classes



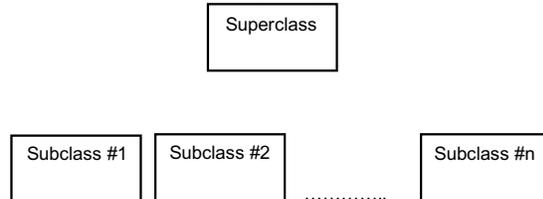
Association Cardinality



Aggregation between classes



Class Inheritance (subtyping of classes)



UML notation

In this standard, the following three stereotypes of UML classes are used:

- <<Interface>> A definition of a set of operations that is supported by objects having this interface. An Interface class cannot contain any attributes.
- <<DataType>> A descriptor of a set of values that lack identity (independent existence and the possibility of side effects). A DataType is a class with no operations whose primary purpose is to hold the information.
- <<CodeList>> is a flexible enumeration that uses string values for expressing a list of potential values.

In this standard, the following standard data types are used:

- CharacterString – A sequence of characters
- Integer – An integer number
- Double – A double precision floating point number
- Float – A single precision floating point number

5.2 XML Namespaces

All components of the Moving Feature XML schema are defined in the namespace with the identifier "<http://www.opengis.net/movingfeatures/1.0>", for which the prefix mf or the default namespace is used within this Standard.

6 Data Model Overview

6.1 Concepts and standard modularity

Figure 1 illustrates the standard components for encoding and communicating Moving Features data. The four boxes drawn in the figure portray the four modules defined in the OGC® Moving Features standard.

OGC® Moving Features Simple CSV Encoding and Simple Binary Encoding are fundamental and are data formats to describe moving points and their trajectories. The formats are designed to encode trajectories of moving points compactly as application schemas of IETF RFC 4180 (CSV), NetCDF Discrete Sampling Geometry and so on.

OGC® Moving Features Core, defined in this document, is represented by an XML data encoding for moving points and their trajectories. The format is an extensible format to model and encode movement data for features encoded as point geometries.

OGC® Moving Features 1D/2D Geometry and OGC® Moving Features 3D Geometry are defined to encode moving features that have more complex geometries.

Two UML packages are defined in ISO 19141:2008 [ISO 19141:2008]: “Geometry Types” and “Prism Geometry”. “Geometry Types” defines the description model of movements including the paths. “Prism Geometry” defines the moving geometry. The two squares drawn with broken lines in Fig. 1 correspond to these packages. The square labeled as “<<Prism Geometry>>” is an implementation of “Prism Geometry”; the square labeled as “<<Geometry Types>>” is an implementation of “Geometry Types.”

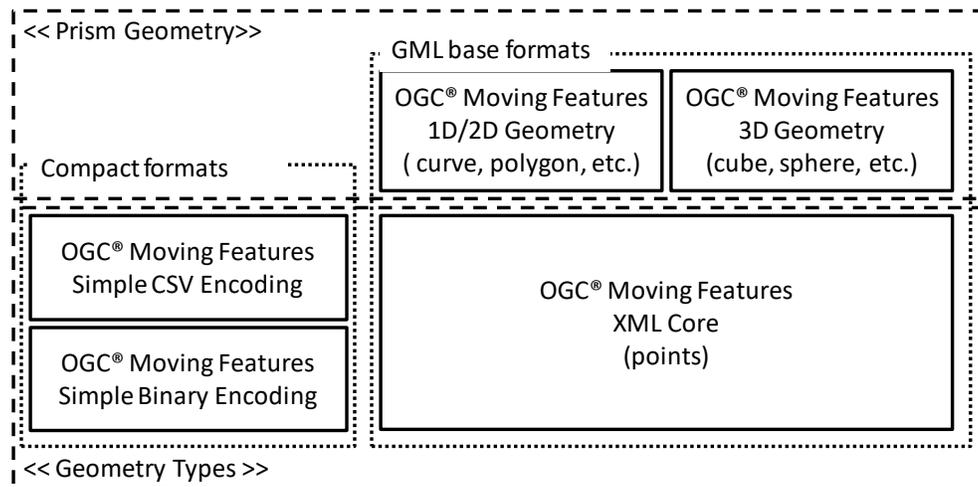


Figure 1. OGC® Moving Features modularity

6.2 Foliation and prisms

Figure 2 illustrates the concepts of foliation, prism, trajectory, and leaf, which are defined in ISO19141:2008[ISO 19141:2008]. In this illustration, a 2D rectangle moves and

rotates. Each representation of the rectangle at a given time is a leaf. The path traced by each corner point of the rectangle is a trajectory. The set of points contained in all of the leaves, and in all of the trajectories, forms a prism. The set of leaves also forms a foliation.

The prism of the moving feature can be viewed as a bundle of trajectories of points on the local engineering representation of the feature's geometry. If viewed in a 4-dimensional spatio-temporal coordinate system, the points on the feature at different times are different points.

Each moving feature has information to indicate the movement path, which is the trajectory shown in the prism model. MF_TemporalTrajectory is defined in ISO 19141:2008 [ISO 19141:2008] to describe the trajectory. OGC Moving Features supports the structure (mf:AbstractTrajectory shown in 9.2 corresponds to MF_TemporalTrajectory).

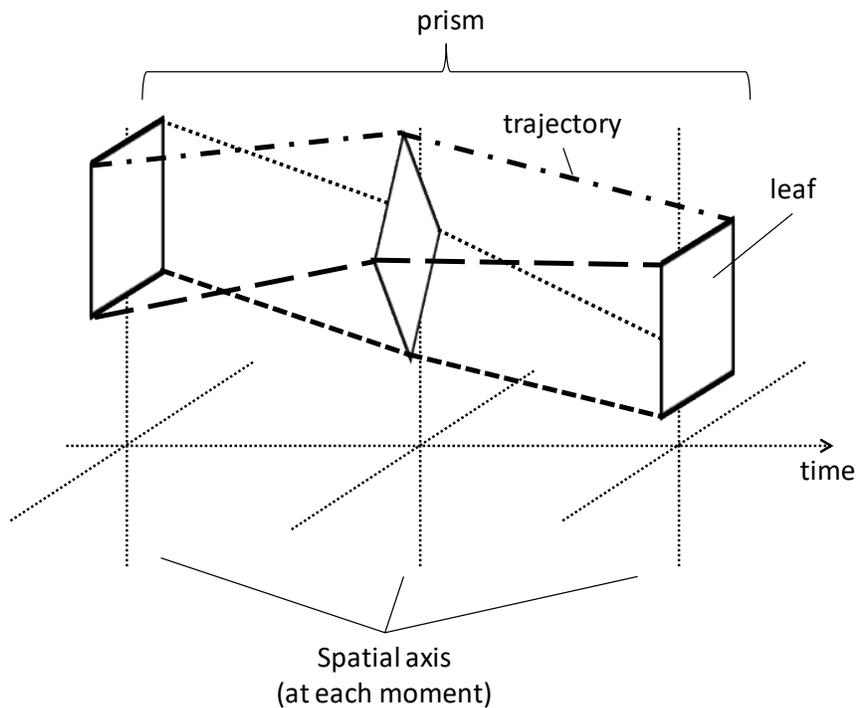


Figure 2. Foliation, prism and trajectory

More generally, a moving feature is a feature which has dynamic attributes. The dynamic attributes indicate attributes changing dynamically. If the value type of a dynamic attribute is geometric point, it is a trajectory. Accordingly, the feature type models for the OGC Moving Features are defined as shown in Figure 3. `MovingFeatureType` inherits `GF_FeatureType` and aggregates a group of `DynamicAttributeType`.

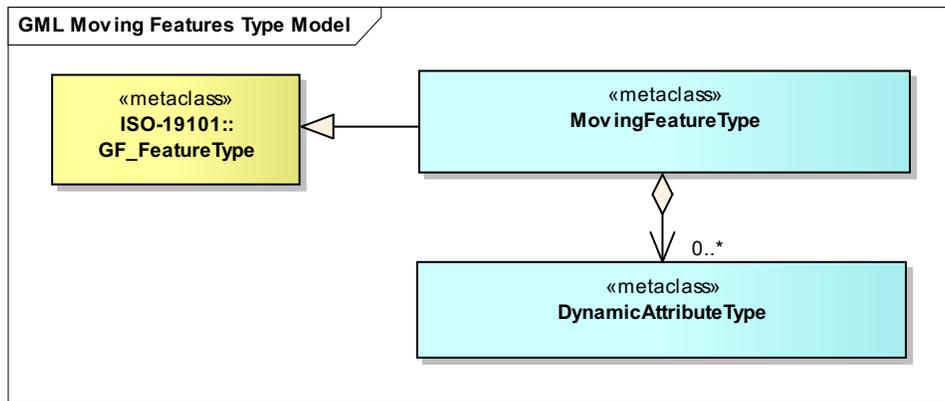


Figure 3. Conceptual model for the moving features

6.3 XML Implementation

An XML data encoding is defined in this standard for implementing the foliation data model. Note that massive amounts of moving feature data may need to be described with this format. Therefore, the data encoding should be compact and easy to handle. Figure 4 illustrates an importing process for a document written in the XML data format. When dealing with massive numbers of moving features, the entire data set may not be able to be loaded into memory. Therefore, the XML data should be parsed on-the-fly. This means that all information needed for interpretation of the XML data should appear at the head of the document. Thus, the XML document written has two main parts: *header part* and *body part*. The header part, which is implemented as meta elements, includes meta information for the XML data. The body part, which is implemented as foliation elements, includes the movement information and dynamic properties for moving features.

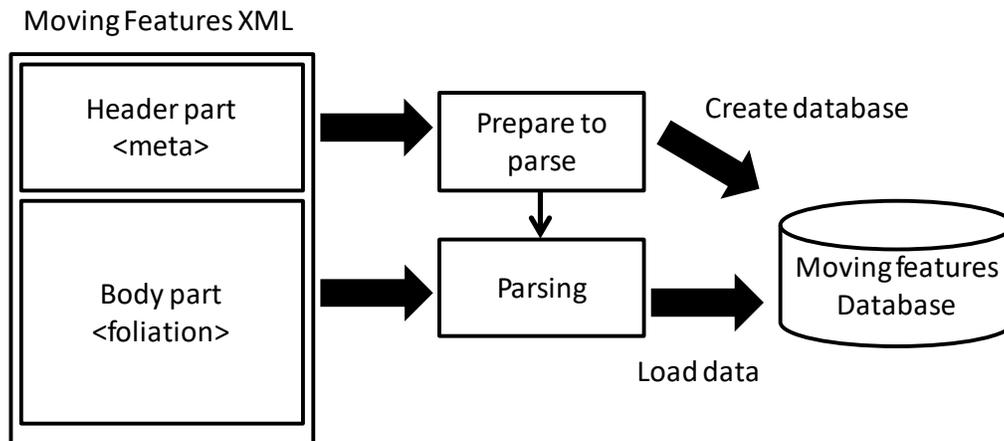


Figure 4. XML parsing

6.4 Time ordered encoding

Figure 5 shows an example for trajectories of three moving points A, B and C. Each trajectory has the start time and the end time. At $t=0$ (start of all data), all points start moving. Then, at $t=1$, the movement of A is changed. In the case, the trajectory of A from $t=0$ to $t=1$, the trajectory of A from $t=1$ to $t=2$, the trajectory of B from $t=0$ to $t=2$, and the trajectory of C from $t=0$ to $t=2$ are encoded. This means that only changes of state, including moving-speed, direction, existence, and attributes, are encoded in the format. The encoded changes of state are ordered by time in order to determine the states of all features even if only the first half of data is loaded.

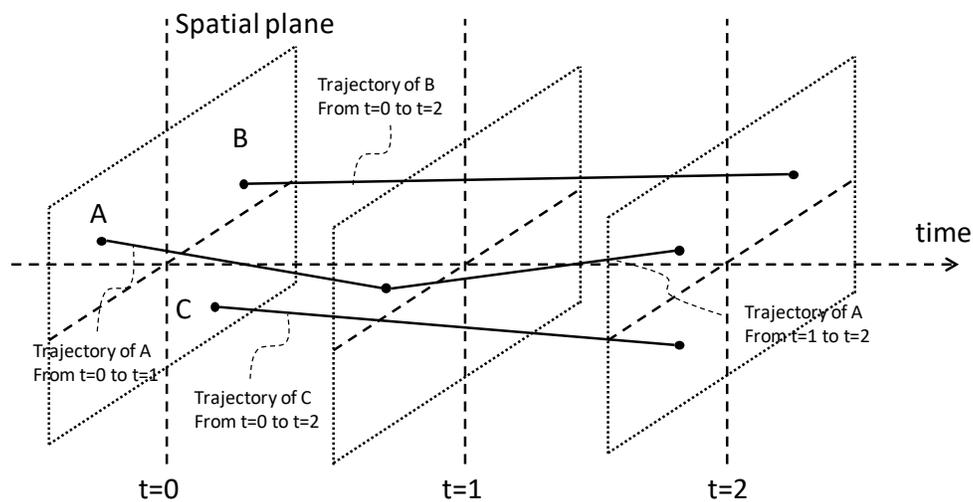


Figure 5. Example of trajectories

7 Root Element

7.1 mf:MovingFeatures

7.1.1 Structure

```
<mf:MovingFeatures gml:id="ID">  
  <mf:sTBoundedBy> ...</mf:sTBoundedBy>[1]  
  <mf:member> ...</mf:member>[0,...,n]  
  <mf:Header> ...</mf:Header>[0,1]  
  <mf:Foliation> ...</Foliation>[0,1]  
</mf:MovingFeatures>
```

Superclass: gml:AbstractFeature [OGC 07-036]

7.1.3.2 **mf:member (optional)**

mf:member elements give the properties of moving features.

7.1.3.3 **mf:Header**

mf:Header element describes the meta information of the XML document.

7.1.3.4 **mf:Foliation**

mf:Foliation includes movements of moving features. The movements are expressed as spatio-temporal curves, which are included in trajectory elements.

7.2 **mf:STBoundedBy**

7.2.1 **Structure**

```
<mf:STBoundedby offset="time unit">
  <gml:EnvelopeWithTimePeriod>...</gml:EnvelopeWithTimePeriod>
</mf:STBoundedBy>
```

Superclass: gml: BoundingShapeType [OGC 07-036]

7.2.2 **Description**

The mf:STBoundedBy element indicates the boundary of the moving features. All trajectories are contained in the spatio-temporal boundary specified at mf:STBoundedBy.

7.2.3 **Content**

7.2.3.1 **gml:EnvelopeWithTimePeriod**

The gml:EnvelopeWithTimePeriod [OGC 07-036] specifies the envelope of the boundary. All moving features included in the foliation should be in the envelope. The used spatial reference system in the envelope (specified by srsName and srsDimension attributes) is default in the foliation. The coordinate system used for the trajectories should be specified by srsName

7.2.4 **Attributes**

7.2.4.1 **offset**

The *offset* attribute shows the unit of time used in the mf:Foliation. “start” attribute and “end” attribute of mf:AbstractTrajectory is described the offset from the gml:beginPosition. The unit of time is defined as the offset attribute.

Attribute values	Definitions	Default
sec	the offset described in seconds is encoded in xsd:decimal style	yes

minute	the offset described in minutes is encoded in xsd:decimal style	
absolute	an absolute time description used in the gml:beginPosition	

7.3 mf:member

7.3.1 Structure

```

<mf:member>
  <gml:AbstractFeatureType gml:id="ID">
    ...
  </gml:AbstractFeatureType>
</mf:member>

<mf:member xlink:href="URI" />

```

7.3.2 Description

The contents of mf:member specifies properties of moving features.

7.3.3 Content

7.3.3.1 gml:AbstractFeature

The gml:AbstractFeature gives the properties of the moving feature. The gml:id can be referred to associate the element with mf:mfldRef.

7.3.4 Attributes

7.3.4.1 xlink:href (optional)

The attribute specifies an URI at which mf:AbstractFeature is given.

7.4 mf:MovingFeature

7.4.1 Structure

```

<mf:MovingFeature gml:id="ID">
  <gml:name>...</gml:name>
</mf:MovingFeature>

```

Superclass: gml:AbstractFeature [OGC 07-036]

7.4.2 Description

The mf:MovingFeature specifies the property of moving features as a kind of gml:AbstractFeature. Normally, elements inheriting gml:AbstractFeature

are defined in an application of Moving Feature XML Core, for example, `app:Vehicle`, `app:Pedestrian`, and so on. The `mf:MovingFeature` is predefined for ease to providing properties of features.

8. Header data

8.1 `mf:Header`

8.1.1 Structure

```
<mf:Header>
  <mf:VaryingAttrDefs> ...</mf:VaryingAttrDefs>[0,1]
  <mf:Hints> ... </mf:Hints> [0,1]
</mf:Header>
```

8.1.2 Description

The `mf:Header` element contains the meta-information for the XML data.

Requirement 8.1 (req/xmlcore/header/structure):

The contents of `mf:Header` **shall** be `mf:VaryingAttrDefs` or `mf:Hints` or both. If both are included, `mf:VaryingAttrDefs` **shall** appear earlier than `mf:Hints`.

8.1.3 Content

8.1.3.1 `mf:VaryingAttrDefs`

This element defines attributes used in the foliation (see 8.3). If the moving features do not have any variable attributes, this element can be omitted.

8.1.3.2 `mf:Hints`

Hints element shows the optional information which is helpful for data processing.

8.2 `mf:VaryingAttrDefs`

The `mf:VaryingAttrDefs` element specifies the available attributes in the XML document. The `mf:VaryingAttrDefs` element contains `mf:AttrDef` elements to enumerate the attribute name and the type of contents.

8.2.1 Structure

```
<VaryingAttrDefs>
  <AttrDef ... />[0...*]
</VaryingAttrDefs>
```

8.2.2 Description

`mf:VaryingAttrDefs` states the definition of attributes used in the foliation.

Requirement 8.2 (req/xmlcore/varyingattrdefs):

`mf:VaryingAttrDefs` **shall** include `mf:AttrDef`, otherwise the content is empty.

8.2.3 Content

8.2.3.1 `mf:AttrDef`

The element indicates a definition of an attribute used in the foliation.

8.3 `mf:AttrDef`

8.3.1 Structure

```
<mf:AttrDef>
  <xsd:simpleType name="">
    ...
    <xsd:annotation> ... </xsd:annotation>
  </xsd:simpleType>
</mf:AttrDef>

<mf:AttrDef name="name of attribute" type="type string">
  <mf:AttrAnnotation>
    ...
  </mf:AttrAnnotation>
</mf:AttrDef>
```

8.3.2 Description

The `AttrDef` element defines the attribute name and the type of contents.

Requirement 8.3 (req/xmlcore/attrdef/attributes)

Any attributes of moving features appearing in `mf:Foliation` **shall** be defined by `mf:AttrDefs` element.

Requirement 8.4 (req/xmlcore/attrdef/datatypes)

The types of the attributes **shall** be consistent with the data descriptions of the attributes.

8.3.3 Content

The type of the attribute is defined in `mf:AttrDef` by `xsd:simpleType`. An example is following.

```

<mf:AttrDef>
  <xsd:simpleType name="speed_meter_record">
    <xsd:union>
      <xsd:simpleType name="km_per_hour">
        <xsd:restriction base="xsd:positiveInteger">
          <xsd:minInclusive value="0"/>
          <xsd:maxInclusive value="100"/>
        </xsd:restriction>
      </xsd:simpleType>
      <xsd:simpleType name="speed_rank">
        <xsd:restriction base="xsd:NMTOKEN">
          <xsd:enumeration value="slow"/>
          <xsd:enumeration value="medium"/>
          <xsd:enumeration value="fast"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:union>
  </xsd:simpleType>
</mf:AttrDef>

```

The attributes name and type can be used instead of the content for ease of the description.

```

<mf:AttrDef name="speed_meter_km_per_hour"
  type="xsd: positiveInteger" />

```

Either of the two styles should be used to define the type of attributes.

Requirement 8.5 (req/xmlcore/attrdef/datatypes_definition)
 A `xsd:simpleType` shall be contained by `mf:AttrDef` if the attributes name and type of `mf:AttrDef` are omitted.

8.3.4 Attributes

8.3.4.1 name (optional)

The name attribute shows the name of the attribute defined in the `mf:AttrDef` element.

8.3.4.2 type (optional)

The type attribute shows the data type of the attribute defined in the `mf:AttrDef` element. The data type is specified by `xsd` data type. Examples of the data types are listed in the following table.

#	Type	Contents
1	<code>xsd:boolean</code>	"True" or "false"

2	xsd:decimal	Real number
3	xsd:integer	Integer number
4	xsd:string	Text data like "I am here." with double quotation marks ("").
5	xsd:dateTime	Text specifying the date time like "1999-05-31T13:20:00.000-05:00".
6	xsd:AnyURI	URI style text

8.4 mf:AttrAnnotation

8.4.1 Structure

```
<mf:AttrAnnotation>
  Annotation Text
</mf:AttrAnnotation>
```

8.4.2 Description

The contents of `mf:AttrAnnotation` is the detailed text for describing the attribute. An example of the annotations is following.

```
<mf:AttrDef name="speed_meter_record" type="xsd:string">
  <mf:AttrAnnotation>
    This attribute shows the speed meter records of the
    vehicles.
    Slow : less than 20km/h
    Medium: more than or equal to 20km/h, and less than 100km/h
    Fast: more than or equal to 100km/h
  </mf:AttrAnnotation>
</mf:AttrDef>
```

Note that `xsd:Annotation` is also available if `xsd:simpleType` is used.

8.5 mf:Hints

8.5.1 Structure

```
<mf:Hints>
  <mf:Hint> - </mf:Hint>[0...*]
</mf:Hints>
```

8.5.2 Description

The `mf:Hints` element includes the information helpful for processing. If the dataset is huge, special techniques to handle the dataset are needed. For such case, the `mf:Hints`

element provides the additional information. For example, the size of dataset is helpful for determining whether the entire data are loadable on the DRAM.

Requirement 8.6 (req/xmlcore/hints):

Contents of the `mf:Hints` element **shall** be `mf:Hint` describing hints for interpretation processes.

8.5.3 Content

8.5.3.1 `mf:Hint`

`mf:Hint` gives predefined hints to process the dataset.

8.6 `mf:Hint`

8.6.1 Structure

```
<mf:Hint name="hintname">value text</mf:Hint>
```

8.6.2 Description

The `mf:Hint` element includes the information helpful for processing. Pair of name attribute and text describes one hint (this is a key-value-pair data).

Requirement 8.7 (req/xmlcore/hint):

Any hints specified by `mf:Hint` **shall** be consistent with the contents of `mf:Foliation`.

8.6.3 Content

The `mf:Hint` element value provides the value of the “key-value-pair”.

8.6.4 Attributes

8.6.4.1 `name`

The name attribute specifies the key of the key-value pair. The content shows the value related to the key. For example, if name is “TrajectoryAppearance” and content is “periodic” like

```
<mf:Hint name=" TrajectoryAppearance">periodic</mf:Hint> ,
```

the trajectory elements will appear periodically. The defined pairs of name and value are listed in the **table 1**.

Table 1. Key-value pairs for mf:Hint element

Name	Value
TrajectoryAppearance	periodic: trajectory elements appear periodically. random: trajectory elements appear randomly.
TrajectoryLifetime	The maximal time length of the trajectories is shown.

9. Foliations

9.1 mf:Foliation

9.1.1 Structure

```
<mf:foliation order="...">
  <mf:AbstractTrajectory> ...</mf: AbstractTrajectory>[0...*]
</mf:foliation>
```

9.1.2 Description

The foliation element, which contains the spatio-temporal geometries for moving features, supports a foliation structure.

Requirement 9.1 (req/xmlcore/foliation)

The foliation element **shall** include `mf:STBoundedBy` with a `srsName` attribute and elements inheriting `mf:AbstractTrajectory` using a coordinate system specified by the `srsName` of `mf:STBoundedBy`.

Requirement 9.2 (req/xmlcore/foliation/order)

`mf:STBoundedBy` **shall** be the first element in `mf:Foliation`. The following elements **shall** be ordered by the rule specified by `order` attribute of the `mf:Foliation`.

9.1.3 Contents

9.1.3.1 mf:AbstractTrajectory

Trajectories of moving features are expressed as subclasses of the `mf:AbstractTrajectory`. Because the subclasses of `mf:AbstractTrajectory` will be defined in the later versions of OGC Moving Features XML Core, softwares supporting the current version should work correctly even if any subclasses of `mf:AbstractTrajectory` are used (i.e. the unsupported types might be ignored),

9.1.4 Attributes

9.1.4.1 “order”

“order” defines appearing order of `mf:AbstractTrajectory` elements. The possible values are enumerated in the following code table.

Table 2. code-table for “order” attribute

Attribute values	Definitions	Default
Time	<code>mf:AbstractTrajectory</code> elements ordered by time strictly	yes
Sequential	<code>mf:AbstractTrajectory</code> elements which are parts of one track are ordered by time	

9.2 `mf:AbstractTrajectory`

9.2.1 Structure

```
<mf:AbstractTrajectory mfIdRef="id" start="t1" end="t2"  
interpolation="interpolation type">  
  <mf:Attr>Attributes text</mf:Attr> [0, 1]  
</mf:AbstractTrajectory>
```

Superclass: `gml:AbstractCurve` [OGC 07-036]

9.2.2 Description

`mf:AbstractTrajectory` is an abstract class to implement the trajectory.

`mf:AbstractTrajectory`, of which superclass is `gml:AbstractCurve` [OGC 07-036], has an attribute to specify the method for temporal interpolation.

Requirement 9.3 (req/xmlcore/abstracttrajectory):

All elements in `mf:Foliation` shall be inheritances of `mf:AbstractTrajectory`.

Each `mf:AbstractTrajectory` describes a segment consisting of a trajectory. If the `mf:AbstractTrajectory` elements with common “`mfid`” attribute are collected and sorted by time, the sequence of the segments constitutes the entire spatio-temporal trajectory of the moving feature identified by the “`mfid`” attribute.

9.2.2 Contents

9.2.2.1 mf:Attr

Attributes variable by time are described in this element. The values of the attributes are constant while the feature moves along the mf:AbstractTrajectory.

9.2.3 Attributes

9.2.3.1 “mfIdRef”

The text specifying the moving feature (i.e. person ID, vehicle ID, etc.)

9.2.3.2 “start”

“start” attribute specifies the start time of the segment.

9.2.3.3 “end”

“end” attribute specifies the end time of the segment.

9.2.3.2 “interpolation” (optional)

The attribute specifies the interpolation method.

“linear” (default): The point during the movement is linearly interpolated. The linearly interpolated points between two end points P1 and P2 are derived by:

$$(1-t)*P1+t*P2, \text{ where } 0 < t < 1.$$

9.3 mf:LinearTrajectory

9.3.1 Structure

```
<mf:LinearTrajectory>
  <gml:PosList>...</gml:PosList>[0, 1]
  <mf:Attr>Attributes text</mf:Attr>[0, 1]
</mf:LinearTrajectory>
```

Superclass: mf:AbstractTrajectory

9.4.2 Description

mf:LinearTrack element is the spatio-temporal line string, which is similar to gml:LineString [OGC 07-036]. The mf:LinearTrack is a special trajectory that consists of a single segment with linear interpolation.

Requirement 9.4 (req/xmlcore/lineartrajectory):

mf:LinearTrajectory **shall** include gml:PosList having more than two points (their dimension is specified by gml:EnvelopeWithTimePeriod in

`mf:sTBoundedBy`).

9.4.3 Contents

9.4.3.1 `gml:posList`

Two or more coordinate tuples are specified in `gml:posList`, with linear interpolation between them. The described line string is the trajectory of the moving featurer.

9.4.3.2 `mf:Attr`

Attributes variable by time are described in this element. The values of the attributes are constant while the feature moves along the `mf:LinearTrajectory`.

9.5 `mf:Attr`

9.5.1 Structure

```
<mf:Attr>Attributes Text</mf:Attr>
```

9.5.2 Description

The `mf:Attr` element contains the attribute information as a text.

Requirement 9.5 (req/xmlcore/attr):

`mf:Attr` **shall** include a CSV text line of which columns are consistent with `mf:AttrDefs`.

9.5.3 Contents

A CSV text which describes the attribute values is contained. An example is the following:

```
12.0,313.1,4,5,a<b,123
```

Some characters may need to be escaped here. `<` (less than), `>` (greater than), `"` (double quotation), `'` (single quotation), and `&` (ampersand) must be replaced with the entity references defined in XML. Space, tab, and comma are written in escape sequences `\s`, `\t`, and `\b`, respectively. If the value equals the previous value, the text for the value can be omitted.

Requirement 9.6 (req/xmlcore/attr/entity):

`<` (less than), `>` (greater than), `"` (double quotation), `'` (single quotation), and `&` (ampersand) contained by `mf:Attr` **shall** be replaced with the entity references.

10 Example instances

10.1 Example 1 Pedestrian's tracks

```
<?xml version="1.0" encoding="UTF-8"?>
<mf:MovingFeatures
  xmlns:mf="http://www.opengis.net/movingfeatures/1.0/mf-
  xmlcore"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

  xsi:schemaLocation="http://www.opengis.net/movingfeatures/
  1.0/mf-xmlcore
  http://schemas.opengis.net/movingfeatures/1.0/mf-xmlcore.xsd"
  gml:id="MFC_0001">
  <mf:STBoundedBy offset="sec">
    <gml:EnvelopeWithTimePeriod
      srsName="urn:ogc:def:crs:EPSG:6.6:4326">
      <gml:lowerCorner>50.23 9.23</gml:lowerCorner>
      <gml:upperCorner>50.31 9.27</gml:upperCorner>
      <gml:beginPosition>2012-01-17T12:33:41Z
      </gml:beginPosition>
      <gml:endPosition>2012-01-17T12:37:00Z </gml:endPosition>
    </gml:EnvelopeWithTimePeriod>
  </mf:STBoundedBy>
  <mf:member xlink:href="#pointerToGmlIdOfMovingFeature_a"/>
  <mf:member xlink:href="#pointerToGmlIdOfMovingFeature_b"/>
  <mf:member>
    <mf:MovingFeature gml:id="a">
      <gml:name>Joe Blow</gml:name>
    </mf:MovingFeature>
  </mf:member>
  <mf:member>
    <mf:MovingFeature gml:id="b">
      <gml:name>Jane Doe</gml:name>
    </mf:MovingFeature>
  </mf:member>
  <mf:header>
    <mf:VaryingAttrDefs>
      <mf:attrDef>
        <xsd:simpleType name="state">
          <xsd:restriction base="xsd:NMTOKEN">
            <xsd:enumeration value="walking"/>
            <xsd:enumeration value="staying"/>
            <xsd:enumeration value="running"/>
          </xsd:restriction>
        </xsd:simpleType>
      </mf:attrDef>
      <mf:attrDef>
        <xsd:simpleType name="typecode">
```

```

        <xsd:restriction base="xsd:integer">
            <xsd:enumeration value="1"/>
            <xsd:enumeration value="2"/>
            <xsd:enumeration value="97"/>
        </xsd:restriction>
    </xsd:simpleType>
</mf:attrDef>
</mf:VaryingAttrDefs>
</mf:header>
<mf:foliation>
    <mf:LinearTrajectory gml:id="LT0001" mfIdRef="a" start="10"
        end="150">
        <gml:posList>11.0 2.0 12.0 3.0</gml:posList>
        <mf:Attr>walking,1</mf:Attr>
    </mf:LinearTrajectory>
    <mf:LinearTrajectory gml:id="LT0002" mfIdRef="b" start="10"
        end="190">
        <gml:posList>10.0 2.0 11.0 3.0</gml:posList>
        <mf:Attr>walking,2</mf:Attr>
    </mf:LinearTrajectory>
    <mf:LinearTrajectory gml:id="LT0003" mfIdRef="a"
        start="150" end="190">
        <gml:posList>12.0 3.0 10.0 3.0</gml:posList>
        <mf:Attr>walking,2</mf:Attr>
    </mf:LinearTrajectory>
</mf:foliation>
</mf:MovingFeatures>

```

10.2 Example 2 Vehicles

```

<?xml version="1.0" encoding="UTF-8"?>
<mf:MovingFeatures
    xmlns:mf="http://www.opengis.net/movingfeatures/1.0/mf-
    xmlcore"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:gml="http://www.opengis.net/gml/3.2"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

    xsi:schemaLocation="http://www.opengis.net/movingfeatures/
    1.0/mf-xmlcore
    http://schemas.opengis.net/movingfeatures/1.0/mf-xmlcore.xsd"
    gml:id="MFC_0002">

    <mf:sTBoundedBy offset="sec">
        <gml:EnvelopeWithTimePeriod
            srsName="urn:ogc:def:crs:EPSG:6.6:4326">
            <gml:lowerCorner>150 0</gml:lowerCorner>
            <gml:upperCorner>250 50</gml:upperCorner>
            <gml:beginPosition>2013-05-01T10:33:41Z
            </gml:beginPosition>

```

```

    <gml:endPosition>2013-05-01T10:37:41Z </gml:endPosition>
  </gml:EnvelopeWithTimePeriod>
</mf:STBoundedBy>

<mf:member xlink:href="#pointerToGmlIdOfMovingFeature_a"/>
<mf:member xlink:href="#pointerToGmlIdOfMovingFeature_b"/>
<mf:member>
  <mf:MovingFeature gml:id="a">
    <gml:description>Nissan Sentra - License plate ABC 123.
    Five passengers.</gml:description>
    <gml:name>NissanA</gml:name>
  </mf:MovingFeature>
</mf:member>
<mf:member>
  <mf:MovingFeature gml:id="b">
    <gml:description>Nishiki 21 speed racer</gml:description>
    <gml:name>BicycleB</gml:name>
  </mf:MovingFeature>
</mf:member>
<mf:header>
  <mf:VaryingAttrDefs>
    <mf:attrDef name="direction" type="xsd:double">
      <mf:AttrAnnotation>
        The direction of the vehicle: the attribute value
        is the angle between the north direction and the vehicle
        direction. Unit of angle is radian.
        For example, North is 0[rad], East is 1.57[rad] and
        West is -1.57[rad].
      </mf:AttrAnnotation>
    </mf:attrDef>
  </mf:VaryingAttrDefs>
</mf:header>
<mf:foliation>
  <mf:LinearTrajectory gml:id="LT0001" mfIdRef="a" start="0"
  end="50">
    <gml:posList>161 5 172 5</gml:posList>
    <mf:Attr>0.0</mf:Attr>
  </mf:LinearTrajectory>
  <mf:LinearTrajectory gml:id="LT0002" mfIdRef="b" start="0"
  end="50">
    <gml:posList>158 20 158 50 159 60</gml:posList>
    <mf:Attr>1.57</mf:Attr>
  </mf:LinearTrajectory>
  <mf:LinearTrajectory gml:id="LT0003" mfIdRef="a" start="50"
  end="100">
    <gml:posList>172 5 172 1</gml:posList>
    <mf:Attr>-1.57</mf:Attr>
  </mf:LinearTrajectory>
  <mf:LinearTrajectory gml:id="LT0004" mfIdRef="b"
  start="50" end="100">
    <gml:posList>159 60 166 50</gml:posList>
    <mf:Attr>0</mf:Attr>
  </mf:LinearTrajectory>

```

```
</mf:foliation>  
</mf:MovingFeatures>
```

Annex A (Normative)

Abstract test suite

This Annex specifies an Abstract Test Suite which shall be passed in completeness by any implementation claiming conformance with this Moving Features XML Core. Test identifiers below are relative to

<http://www.opengis.net/spec/MovingFeatures/1.0/conf/>

A.1 Conformance Class:

The OGC URI Identifier of this conformance classes is

<http://schemas.opengis.net/movingfeatures/1.0/mf-xmlcore.xsd>.

A.1.1 XML Validity

Test id: [conf/xmlcore/xmlcore-valid](#)

Requirements:

- [8.1 req/xmlcore/header/structure](#)
- [8.2 req/xmlcore/varyingattrdefs](#)
- [8.6 req/xmlcore/hints](#)
- [9.1 req/xmlcore/foiliation](#)
- [9.3 req/xmlcore/abstracttrajectory](#)
- [9.6 req/xmlcore/attr/entity](#)

Test purpose:

Verify that any XML element defined in namespace “mf” is well-formed and valid.

Test method:

Validate the XML document using the XML schema document <http://schemas.opengis.net/MovingFeatures/1.0/xmlcore.xsd>. Pass if no errors reported. Fail otherwise.

A.1.2 Root Element

Test id: [conf/xmlcore/movingfeatures](#)

Requirements:

- [7.1 req/xmlcore/movingfeatures](#)

Test purpose:

Verify that the root element is “mf:MovingFeatures”.

Test method:

Validate that the root element of the document is mf:MovingFeatures.

A.1.3 Attribute Definition

Test id: [conf/xmlcore/attributes](#)

Requirements:

- [8.3 req/xmlcore/attrdef/attributes](#)
- [8.4 req/xmlcore/attrdef/datatypes](#)
- [8.5 req/xmlcore/attrdef/datatypes_definition](#)
- [9.5 req/xmlcore/attr](#)

Test purpose:

Verify that attribute definitions are consistent with the attributes shown in the trajectories.

Test method:

At First, the contents of mf:AttrDef is checked. If it is empty, the attributes of mf:AttrDef are checked. If either the name attribute or the type attribute is empty, the test is failed.

After that mf:AttrDef is checked, the contents of mf:attr are parsed as a CSV data. Then the number of columns and the datatype of the columns are checked. If they are consistent with the definition in mf:attrdefs, the test is passed. If not, it is failed.

A.1.3 Time order

Test id: [conf/xmlcore/order](#)

Requirements:

- [9.2 req/xmlcore/foiliation/order](#)

Test purpose:

Verify that “order” attribute of mf: foliation is consistent with the contents of the foliation.

Test method:

The “start” attributes of mf:AbstractTrajectory (and the inheritances) are checked. If the order is consistent with the “order” attributes, the test passes.

A.1.4 Hints**Test id:** [conf/xmlcore/hints](#)**Requirements:**

- [8.7 req/xmlcore/hint](#)

Test purpose:

Verify that mf:Hints is consistent with the contents of the foliation.

Test method:

The contents of mf:Hint are checked.

- If TrajectoryAppearance is specified, the “start” and “end” attributes of mf:AbstractTrajectory (and the inheritances) are checked. If they are consistent, the test passes.
- If TrajectoryLifetime is specified, the difference between “start” and “end” attributes of mf:AbstractTrajectory (and the inheritances) are checked. If they are less than the specified life time, the test passes.

A.1.5 sTBoundedBy**Test id:** [conf/xmlcore/lineartrajectory](#)**Requirements:**

- [8.7 req/xmlcore/lineartrajectory](#)

Test purpose:

Verify that the mf:LinearTrajectory is valid.

Test method:

If gml:PosList included by mf:LinearTrajectory has more than two points which are contained by the spatio-temporal envelope determined by mf:sTBoundedBy, the test passes.

Annex B (Informative)

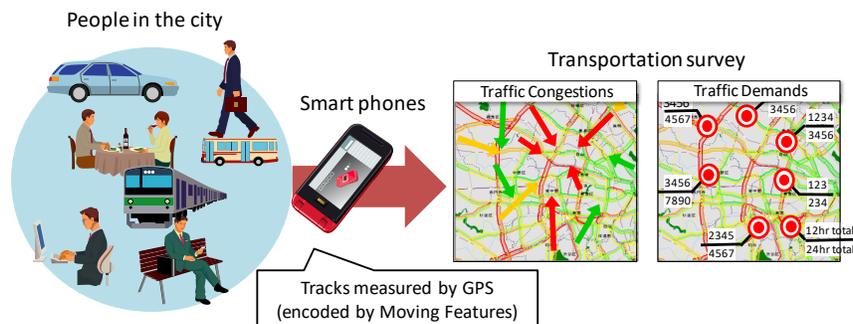
Application examples

B.1 Transportation survey with smart phones

Traffic congestion is a serious problem in many cities. Basically, two types of solutions are applied to resolve congestion: road-construction and enhancement of public transportation systems. Both of the solutions require information on trip demands: the number of people's movements between two places, because roads and public transportation systems should make connection between the pairs of places that have the highest trip demand. Therefore, the trip demands should be figured out by transportation survey.

Questionnaires are traditionally used to survey trip demands. However, because GPS enabled smart-phones are now available, GPS based research is applicable as a sampling alternative. The GPS tracks are encoded by Moving Features standard in order to be shared by many stakeholders like local governments, bus companies, and so on.

Features				
Objects	Detail	Movement properties	Attributes	
Pedestrians	Walking people to go somewhere	Speed: 4km/h	State: Stay or Walking	
Vehicles	People driving his/her vehicle	Speed: 60km/h		
Data sources				
Data sources	Localization method	Sampling rate	Accuracy	Number of data
Smart phones	GPS/Wi-Fi positioning	1 point per 1s	100-10m	Over 100 tracks



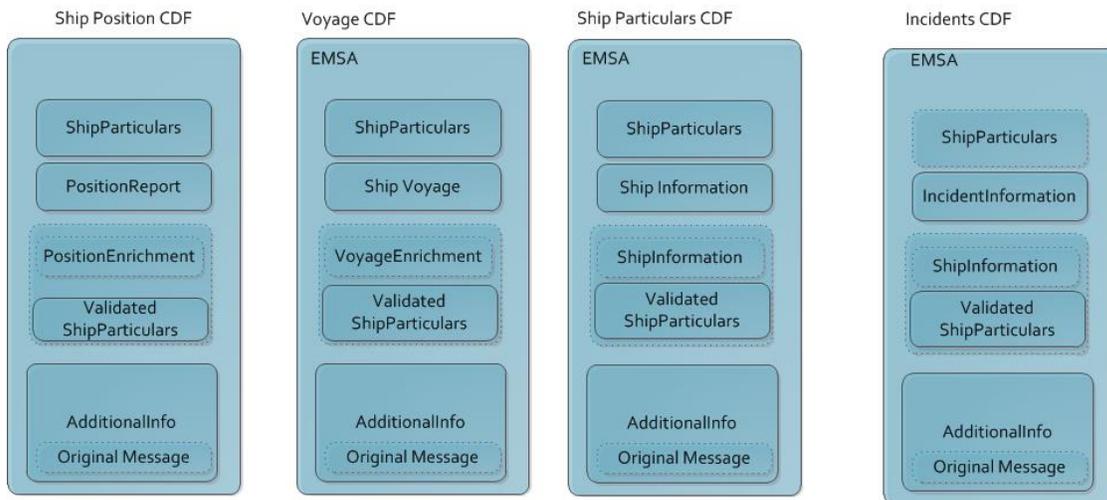
Background map: ©Open Street Map

B.2 Maritime sector use case

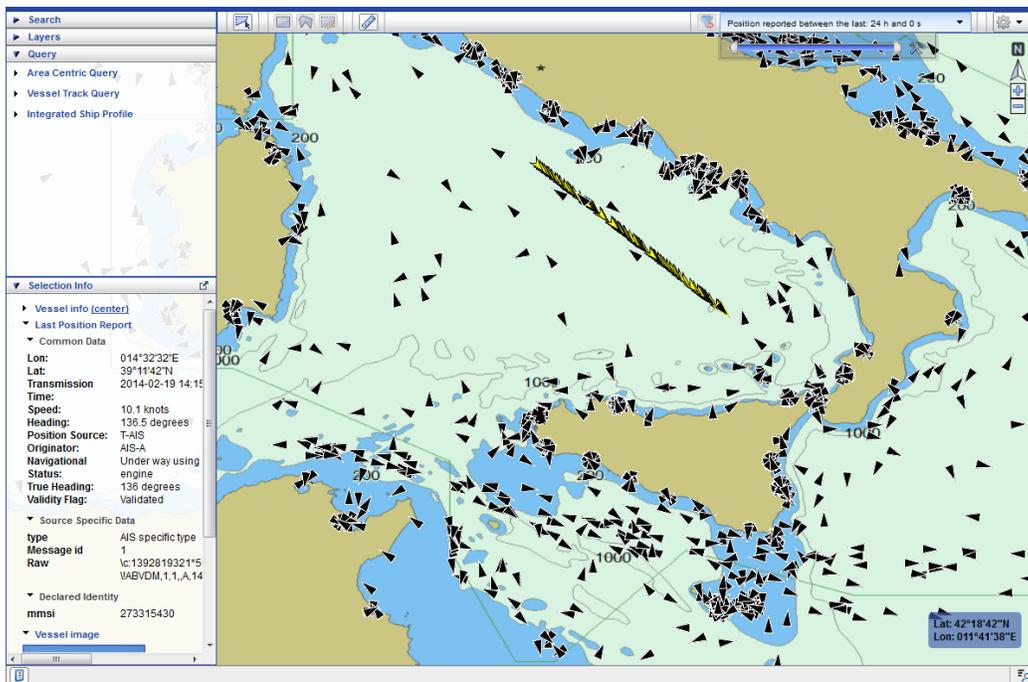
Several devices have been used by vessels to track their positions. Some of these devices are mandatory as required by regional/international standardization initiatives (i.e. EU Directives, IMO conventions, etc ...).

The scope to track the position of the vessels is twofold: increase the safety and security within the maritime sector.

In order to address maritime use cases the Moving Feature standard has to include at least the following information: (i) ship position, which can be provided by different data source (AIS, LRIT, etc ...); and (ii) ship voyage, which describe the track of vessels. Additional information about the vessel incidents and the ship particulars has to be considered complementary, therefore as possible extensions to the Moving Feature standard. For example, the following figure summarize the information collected by the Canonical Data Format produced by the European Maritime Safety Agency



The following picture reports the last position of vessels within a specific area (black triangle), and the track of a specific vessel (yellow triangles). In this case the yellow triangles report the position of a specific vessel every 6 minutes in the last 24 hours. Furthermore, additional information as: heading, true heading and speed is recorded for enrich the description of the vessel track.



A vessel has a unique identification at a worldwide scale (IMO number, or MMSI), the position of a vessel (ship position) is a point feature over the time. The voyage feature of a vessel (ship voyage) is characterized by several attributes such as: speed over the ground, course over the ground, heading and true heading.

To build the trajectory of a vessel, two main abstract feature types are relevant for Moving Feature standard: ship position and ship voyage.

Possible uses of the information collected by maritime authorities are:

Features				
Objects	Detail	Movement properties	Attributes	
Vessels	enter/leave/inside a particular area X	Speed 11 knots in/out a Feature of Interest		
Vessels	Suspicious manoeuvres	Change the speed rapidly and/or varying		
Vessels	Discrepancy between declared Next Port or arrival time and current position/course	Current position and voyager declared		
Vessels	Approaching dangerous Feature Of Interest	Distance from a rocks less than 3 miles		
Data sources				
Data sources	Localization method	Sampling rate	Accuracy	Number of data

Automatic Identification System (AIS) / Satellite AIS	GPS/Wi-Fi positioning	1 point every 6 minutes	100-10m	Over 10 tracks per hour
---	-----------------------	-------------------------	---------	-------------------------

B.3 Aviation sector use case

Aircraft and other airborne vehicles are obviously moving features as well. The aviation sector has a (large) number of methods to track aircraft and to represent their position.

The main sources of data include the following

- Primary surveillance radar: Radar equipment measuring position and heading of aircraft.
- Secondary surveillance radar: position detection is augmented with an active response from the aircraft's transponder. It supplies additional information such as its identity.
- ADS-B: aircraft with on board GPS or other GNSS equipment may also provide position information based on the GPS measurement. This mode of position and information broadcasting is called ADS-B and is seen as an important future tool for increasing air traffic capacity and safety.

The position and related data of aircraft is encoded according to different standards. One standard is Eurocontrol ASTERIX (<http://www.eurocontrol.int/asterix>). ASTERIX defines different categories, most categories containing position data according to a different detection method.

For instance, category 48 defines the following fields (copied from the Cat 48 specification on <http://www.eurocontrol.int/services/specifications-documents>)

Data Item	Description	Description System Units
I048/010	Data Source Identifier	N.A.
I048/020	Target Report Descriptor	N.A.
I048/030	Warning/Error Conditions	N.A.
I048/040	Measured Position in Slant Polar Co-ordinates	RHO: 1/256 NM, THETA: 360°/(2 16)
I048/042	Calculated Position in Cartesian Co-ordinates	X, Y: 1/128 NM
I048/050	Mode-2 Code in Octal Representation	N.A.
I048/055	Mode-1 Code in Octal Representation	N.A.
I048/060	Mode-2 Code Confidence Indicator	N.A.
I048/065	Mode 1 Code Confidence Indicator	N.A.
I048/070	Mode-3/A Code in Octal Representation	N.A.
I048/080	Mode-3/A Code Confidence Indicator	N.A.
I048/090	Flight Level in Binary Representation	1/4 FL

I048/100	Mode-C Code and Confidence Indicator	N.A.
I048/110	Height Measured by a 3D Radar	25 ft
I048/120	Radial Doppler Speed	(2-14) NM/s
I048/130	Radar Plot Characteristics	N.A.
I048/140	Time of Day	1/128 s
I048/161	Track/Plot Number	N.A.
I048/170	Track Status	N.A.
I048/200	Calculated Track Velocity in Polar Representation	Speed: (2-14) NM/s, Heading:360°/(2 16)
I048/210	Track Quality	N.A.
I048/220	Aircraft Address	N.A.
I048/230	Communications / ACAS Capability and Flight Status	N.A.
I048/240	Aircraft Identification	N.A.
I048/250	Mode S MB Data	N.A.
I048/260	ACAS Resolution Advisory Report	N.A.

The items in bold relate to the position, heading, time, and identity of the aircraft. Other information fields provide metadata on quality, data source, etc.

Other ASTERIX categories provide different sets of information where position and other information may also be encoded in different ways. Some interesting aspects to ASTERIX:

- The positions may be expressed in relation to the radar equipment position. A (separately available) position of the radar is thus needed to correctly geo-reference the aircraft.
- The data has a binary encoding optimized to obtain a small message size. Verbose GML/XML encoding may be a barrier for adoption of the Moving Feature standard for large data sets in the aviation world.
- Many data sets do not include all fields. Optional elements in the Moving Features may accommodate representing such data sets.
- Some categories or data sets do not include an identification of the moving features. Such data is often referred to as ‘plots’. Since multiple plots cannot be related to a single aircraft, the data set basically is a bag of time-stamped points.

FAA has an Aircraft Situation Display to Industry (ASDI) feed, which provides aircraft position reports as a convenient data feed to industry. The feed also includes information on flight plans (the planned route of the aircraft). See <http://www.fly.faa.gov/ASDI/asdi.html> for more information.

In order to address aviation use cases at least this information must be managed in the moving feature standards. Other information is important as well and may be encoded as use-case specific metadata.

Possible uses of the information collected by aviation authorities are:

Features				
Objects	Detail	Movement properties	Attributes	
Aircraft	Performing a flight	Speed: 300-1000km/h	Many	
Aircraft	Taxiing on a airport	Speed: 0-300km/h	Many	
Data sources				
Data sources	Localization method	Sampling rate	Accuracy	Number of data
Primary/Secondary Surv. Radar	Radar detection	1 to several seconds per position	Depends on radar & distance to radar	Thousands of concurrent flights world-wide
ADS-B	GNSS/GPS	1 to several seconds per position	100-10m	Thousands of concurrent flights world-wide

Use cases can also involve analysis afterwards (incident analysis, statistics, etc.) or simulated air traffic. The moving features standard may be useful to convert raw feeds into a standard moving features representation for use in generic analysis and processing tools.

B.4 A Use case for Hurricanes

Tropical storms, hurricanes and typhoons are extremely energetic and dangerous weather phenomena with a pronounced centre of rotation, the 'eye'. However, their position can be forecast, with increasing accuracy, several days in advance, and even more accurately only a day in advance. These forecasts are normally presented to the general public and emergency services as a simple graphical map, often using PNG or JPEG formats.

Hurricanes and typhoons are the same phenomenon, but traditionally the names in the western Pacific and western Atlantic are different. A tropical storm is a weaker version, causing less damage, but can become more energetic. Currently a wind speed threshold of 65 knots (120Km/hour, 75 miles/hour) is the only distinguishing attribute.

Typically a deterministic chart will show a single forecast storm trajectory as a single track with the location of the storm centre or 'eye' indicated at regular intervals, say every 6 or 12 hours. The location at a specific time is usually marked by specific symbols, standardised across all countries for both meteorology and aviation.

Sometimes a probabilistic chart is preferred, and a range of possible trajectories, or an envelope of these trajectories is used. The set of trajectories may come from a single forecast centre using an ensemble of possible forecasts, or the set may be from several forecast centres which all produce slightly different forecasts. Generally, if the spatial, and temporal, spread of the set of trajectories is small, then more confidence is attached to them than if the spread is great.

Expert users, such as forecasters, may also use a chart that displays ranges of trajectories from successive forecast times, and the actual trajectory so far. Again, similarity in these successive forecasts gives increased confidence in the accuracy of the forecasts.

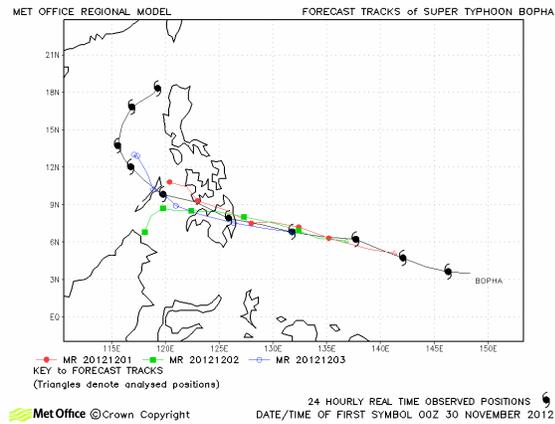
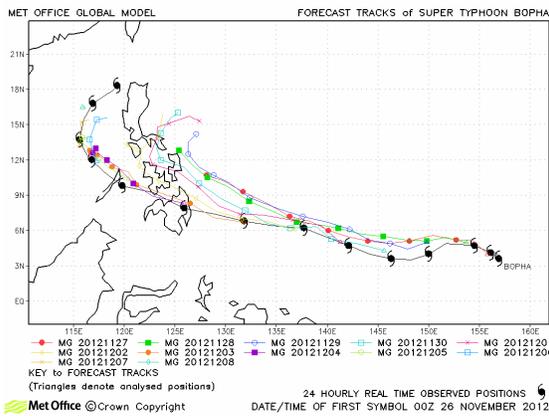
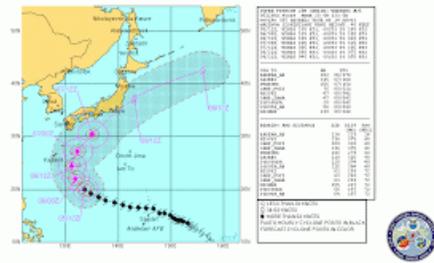
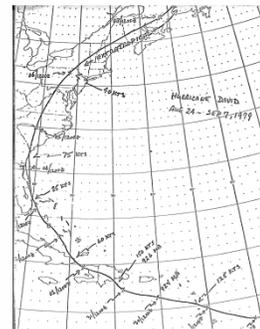
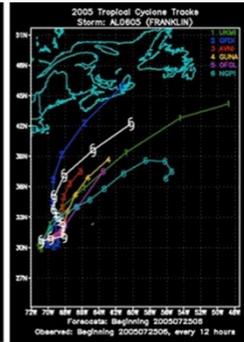
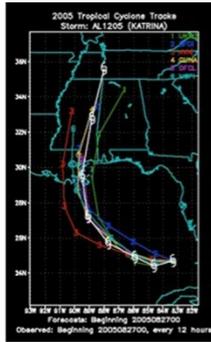
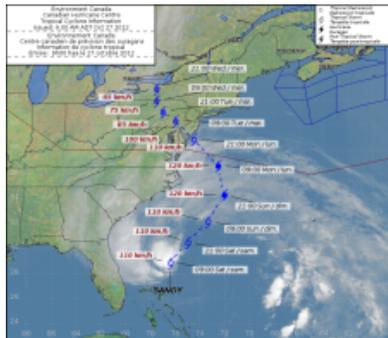
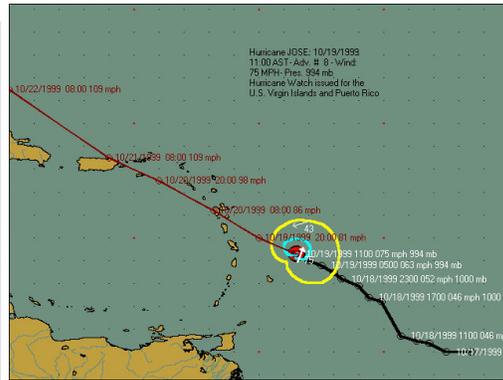
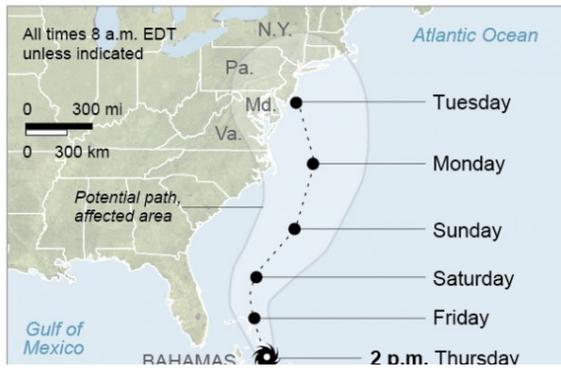
If graphical output is inappropriate, there is a standardised simple text message used to transmit the current and forecast locations across emergency communications channels. These messages also include the actual and forecast wind speeds at various distances and directions from the 'eye'. The atmospheric pressure near the eye may also be included.

The central pressure is an important indicator of damage, like the wind speed, as the reduced pressure causes the local sea surface to rise by a few metres causing flooding and destructive waves.

The sea surface temperature in the path of the storm is also important, as this determines whether the storm will increase or decrease in strength.

So there are a number of changing attributes need to be associated with the storm and its trajectory.

to reach the East Coast, as a tropical storm, next week.



Features

Objects	Detail	Movement properties	Attributes	
Data sources				
Data sources	Localization method	Sampling rate	Accuracy	Number of data

B.5 Predicting Crime patterns by simulating movement of criminals, victims and police on map

Goal: To replicate the conditions under which crime is committed and predict crime patterns, for example predicting regions more vulnerable to crime based on Routine Activity Theory.

Setup: All these phenomena move only along roads on the map. There are certain fixed marked spots on the map. These include Criminal's Home, Office, Mall, Bar, Restaurants. There is a weekly schedule and the criminal agent moves between these spots based on the schedule.

Forms of Motion:

- Beeline Movement: The agent chooses one of the shortest 3 paths between two spots randomly and moves along it. This is time bound - if there are less than 3 routes that can be covered in the amount of time the agent has - only that number of routes are considered for random selection.
- Random Movement: Agent starts from source and at each road intersection, agent chooses a road randomly and moves along it. This is also time bound - when time runs out, the agent moves to the destination along the shortest path from its current location.
- Hybrid Movement: Agent moves between source and destination along the shortest path with deviations. Agent randomly chooses two points on the shortest route and takes a longer different route between these two points. If there is time remaining, agent repeats this, otherwise moves to the destination along the shortest path.

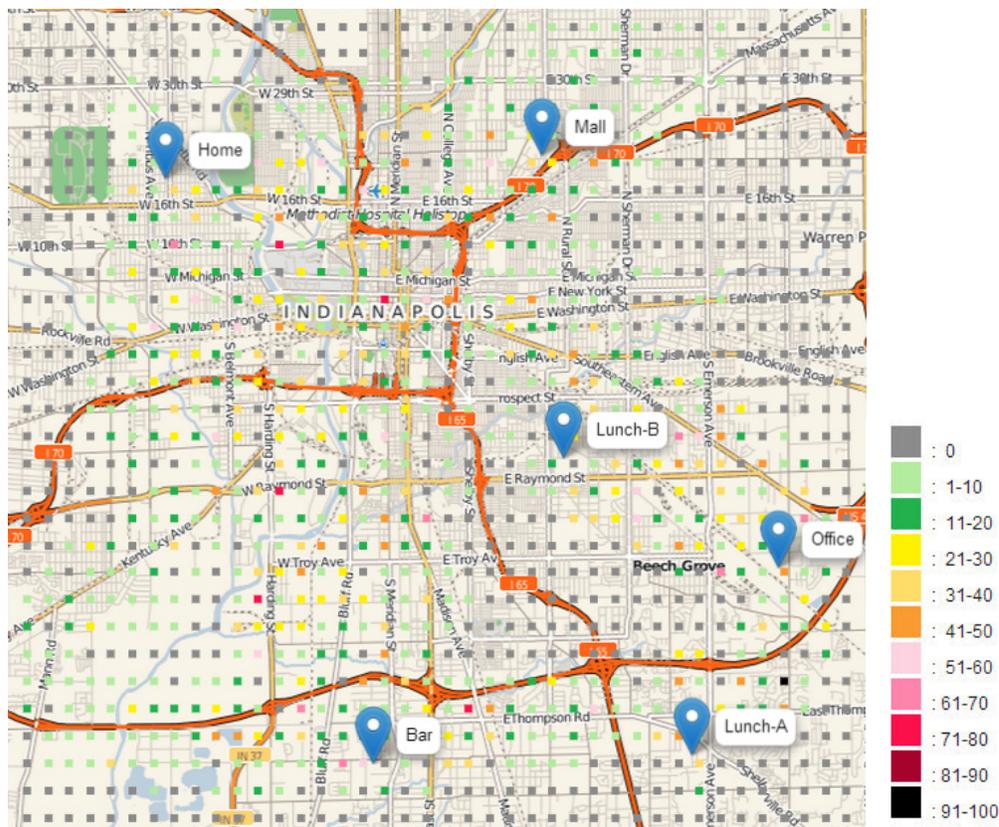
Prediction of Crime Patterns: Criminal Agent, Victim Agents and Police all keep moving along the map as per their schedules. There are a number of designated crime spots on the map. When the criminal agent passes through the vicinity of a crime spot, it decides whether or not to commit crime based on factors like profit value associated with crime spot, risk involved, presence of police nearby, absence of guardian of spot and the number of times the spot has been visited before. Crime Patterns are predicted by running this simulation a very large number of times.

Discussion:

- The simulation needs a city description with roads and buildings of interest. CityGML could be used here.
- The simulation needs to represent the agent's trajectory in the above-cited city. Moving Features standard would be used here.
- The trajectory is repeated weekly with random variations. That may be represented as a "base line" moving feature. This base line would not describe a real moving feature, but would be used as a template.
- Interactions between the environment (CityGML) and the moving features may trigger events. For example when <mf:Trajectory> value become close to a CityGML feature or interest having a low "presence of police" attribute, a crime may occur depending on the value of that attribute and other attributes like "absence of guardian", "profit", and "risk".

Features				
Objects	Detail	Movement properties	Attributes	
Criminal Agent	The one who commits crime			
Victim Agent	The one whose property is under risk of theft			
Police Agent	Police patrol the city to prevent crime			
Data sources				
Data sources	Localization method	Sampling rate	Accuracy	Number of data
Simulation	Movement simulation along Road map (Open Street Map)			

Example: The criminal agent moves between Home, Office, Bar, Lunch-A, Lunch-B, Mall marked on the map as per a weekly schedule. The result corresponds to the weekly schedule run for 500 times on the city: Indianapolis. The small squares on the map correspond to crime spots. The color of each crime spot represents the number of times crime has been committed at that spot, normalized on a scale of 0-100.



B.6 Use-Cases for Soccer Matches

Background: It is an important task of soccer coaches to analyze matches for developing proper tactics and preparing for future matches. With recent progress of several video analysis and sensor technologies, it is now possible to gather match data including trajectories of both players and the ball.

Feature Modelling [1]: A soccer match is composed of 23 moving features including 22 players and ball, where each moving feature has $p(x,y)$ or $p(x,y,z)$ coordinates for each time instance during a match. A sequence of moving point for a player or ball forms a trajectory. A basic feature model for soccer match in terms of moving feature is given as the following figure.

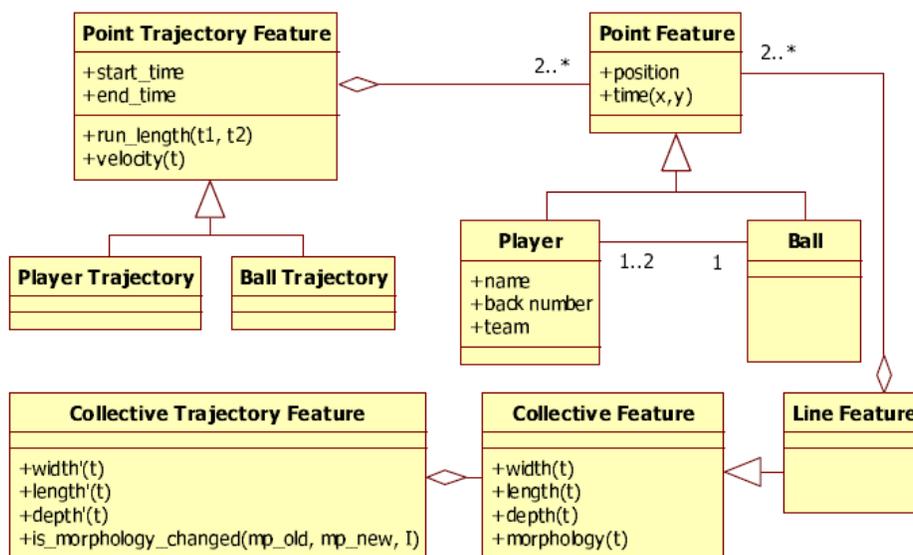


Figure B.6-1. Basic Feature Model for Soccer Match

We define two abstract feature types Point Feature and Point Trajectory Feature as the root classes of the model of figure 1, which correspond to spatial and spatiotemporal objects respectively. And Ball and Player, which inherit from Point Feature, have the coordinates of the ball and players at each sampling time instance. Ball Trajectory and Player Trajectory represent the feature types of ball and player trajectories respectively. Each object of Point Feature has (x, y) coordinates at each time instance t . Note that we employ (x, y) -coordinate reference system to specify the position of spatial objects from the left-bottom corner of the field. And we assume the continuity of temporal domain even though there may be several breaks of match such as ball-out. Due to the breaks, we cannot assure the continuity of the ball trajectory. Since soccer is a collective sport, it is required to analyze the positions of players as a collection as well as position of each individual player. For example, the defensive tactics would be better understood by analyzing the shape of the defensive players in a collective way, rather than the position of each individual defender separately. For this reason, we define Collective Feature and Collective Trajectory. We may define additional properties for player such as direction.

Based on the information of this basic feature model, we can derive more useful information from primitive behaviours such as kicks, ball possession, dribbles (see figure 2), intercepts, passes, to complicated strategies such as formation, defence or attack systems.

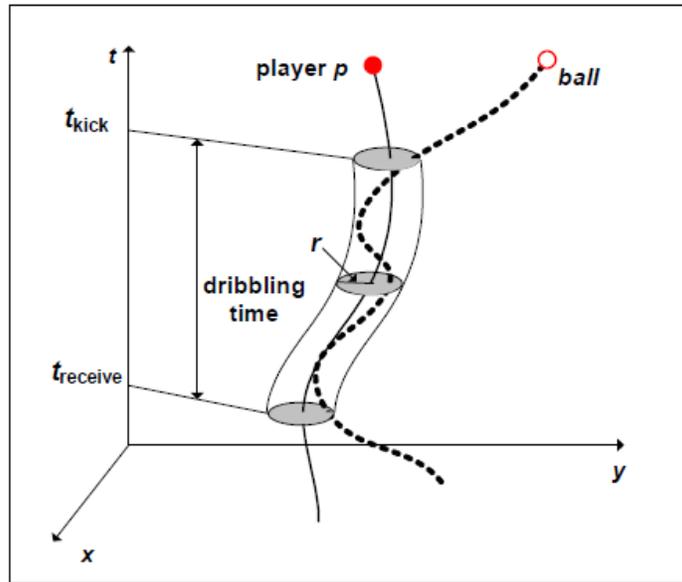


Figure B.6-2. Dribbles [2]

Annex C (Informative)

XML Schema for MovingFeatures

In addition to this document, this standard includes some normative XML Schema Documents. These XML Schema Documents may be included in this document. These XML Schema Documents are also posted online at the URL <http://schemas.opengis.net/movingfeatures/1.0/>. In the event of a discrepancy between this online document and online versions of the XML Schema Document files (posted at schemas.opengis.net), the online schema files SHALL be considered authoritative (**normative**).

Annex D

Bibliography

[1] Ho-Chul Kim, Oje Kwon and Ki-Joune Li, "Spatial and Spatiotemporal Analysis of Soccer", Proceedings of ACM SIGSPATIAL GIS '11, November 1-4, 2011. Chicago, IL, USA

[2] Chan-Hyun Kang, Jung-Rea Hwang, and Ki-Joune Li, "Trajectory analysis for soccer player", Spatial and Spatio-temporal Data Mining (SSTDM), the IEEE Computer Society Press

Annex E (Informative)

Revision History

Date	Release	Author	Paragraph modified	Description
2014/11/13	OGC 14-084r2	Akinori Asahara	3	References were added
2014/11/18		Carl Reed	Numerous	Final edits prior to publication
2017/12/XX		Akinori Asahara	6.2, 6.3	Figure 3 and its description were added