INTERNATIONAL STANDARD

ISO 17572-3

First edition
2008-12-15

Intelligent transport systems (ITS) — Location referencing for geographic databases —

Part 3 Dynamic location references (dynamic profile)

Systèmes intelligents de transport (SIT) — Localisation pour bases de données géographiques —

Partie 3: Localisations dynamiques (profil dynamique)

Reference number
ISO 17572-3:2008(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

ISO 17572-3 was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

ISO 17572 consists of the following parts, under the general title *Intelligent transport systems (ITS) — Location referencing for geographic databases*:

— *Part 1: General requirements and conceptual model*

— *Part 2: Pre-coded location references (pre-coded profile)*

— *Part 3: Dynamic location references (dynamic profile)*

Introduction

A Location Reference (LR) is a unique identification of a geographic object. In a digital world, a real-world geographic object can be represented by a feature in a geographic database. An example of a commonly known Location Reference is a postal address of a house. Examples of object instances include a particular exit ramp on a particular motorway, a road junction or a hotel. For efficiency reasons, Location References are often coded. This is especially significant if the Location Reference is used to define the location for information about various objects between different systems. For Intelligent Transport Systems (ITS), many different types of real-world objects will be addressed. Amongst these, Location Referencing of the road network, or components thereof, is a particular focus.

Communication of a Location Reference for specific geographic phenomena, corresponding to objects in geographic databases, in a standard, unambiguous manner is a vital part of an integrated ITS system, in which different applications and sources of geographic data will be used. Location Referencing Methods (LRM, methods of referencing object instances) differ by applications, by the data model used to create the database, or by the enforced object referencing imposed by the specific mapping system used to create and store the database. A standard Location Referencing Method allows for a common and unambiguous identification of object instances representing the same geographic phenomena in different geographic databases produced by different vendors, for varied applications, and operating on multiple hardware/software platforms. If ITS applications using digital map databases are to become widespread, data reference across various applications and systems must be possible. Information prepared on one system, such as traffic messages, must be interpretable by all receiving systems. A standard method to refer to specific object instances is essential to achieving such objectives.

Japan, Korea, Australia, Canada, the US and European ITS bodies are all supporting activities of Location Referencing. Japan has developed a Link Specification for VICS. In Europe, the RDS-TMC traffic messaging system has been developed. In addition, methods have been developed and refined in the EVIDENCE and AGORA projects based on intersections identified by geographic coordinates and other intersection descriptors. In the US, standards for Location Referencing have been developed to accommodate several different Location Referencing Methods.

This International Standard provides specifications for location referencing for ITS systems (although other committees or standardization bodies may subsequently consider extending it to a more generic context). In addition, this version does not deal with public transport location referencing; this issue will be dealt with in a later version.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this document may involve the use of a patent concerning procedures, methods and/or formats given in this document in Clauses 8 and 9 and Annexes A, B and C.

ISO takes no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured ISO that he/she is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with ISO. Information may be obtained from:

PANASONIC, Matsushita Electric Co., Ltd.	OBP Panasonic Tower, 2-1-61 Shiromi, Chuo-ku, Osaka, 540-6208, Japan
Blaupunkt GmbH	Robert-Bosch-Str. 200, 31139 Hildesheim, Germany
Siemens AG	Philipstr. 1, 35576 Wetzlar, Germany
Tele Atlas NV	Reitscheweg 7F, 5232 BX 's-Hertogenbosch, Netherlands
Toyota Motor Co. (et al)	1 Toyota-Cho, Toyota City, Aichi Prefecture 471-8571, Japan

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. ISO shall not be held responsible for identifying any or all such patent rights.

Intelligent transport systems (ITS) — Location referencing for geographic databases —

Part 3: Dynamic location references (dynamic profile)

1 Scope

This International Standard specifies Location Referencing Methods (LRM) that describe locations in the context of geographic databases and will be used to locate transport-related phenomena in an encoder system as well as in the decoder side. This International Standard defines what is meant by such objects, and describes the reference in detail, including whether or not components of the reference are mandatory or optional, and their characteristics.

This International Standard specifies two different LRMs:

- pre-coded location references (pre-coded profile);
- dynamic location references (dynamic profile).

This International Standard does not define a physical format for implementing the LRM. However, the requirements for physical formats are defined.

This International Standard does not define details of the Location Referencing System (LRS), i.e. how the LRMs are to be implemented in software, hardware, or processes.

This part of ISO 17572 specifies the dynamic location referencing method, comprising:

- attributes and encoding rules;
- logical data modelling;
- TPEG physical format specification for dynamic location references;
- coding Guidelines for Dynamic Location References;
- compressed Data Format Specification.

It is consistent with other International Standards developed by ISO/TC 204 such as ISO 14825, *Intelligent transport systems — Geographic Data Files (GDF) — Overall data specification*.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 17572-1, *Intelligent Transport Systems (ITS) — Location referencing for geographic databases — Part 1: General requirements and conceptual model*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 17572-1 and the following apply.

3.1
bearing
angle between a reference direction and the direction to an object measured clockwise

NOTE Unless otherwise specified, the reference direction is generally understood to be geographic north.

3.2
connection angle
CA
difference between **side road bearing** and **bearing** at a point

3.3
connection point
location point captured in the location reference core, which forms the start point of a path external to the location

NOTE 1 Connection points are used to connect a location reference extension to a location reference core and to connect linear locations to form a subnetwork. The connection point is identified using its connection point index.

NOTE 2 The connection point index is implicitly defined by the order of the points in a location reference.

3.4
connectivity
status of being topologically connected

NOTE In a graph two or more edges are said to be connected if they share one or more nodes.

3.5
coordinate pair
set of two coordinates (one longitude value and one latitude value), representing a position on the earth model

NOTE Within the scope of this International Standard the earth model is embodied by ITRS and by ITRF coordinates.

3.6
core point
CP
point belonging to the location reference core

3.7
destination location
location to be used as the end location of a journey for a route guidance application

3.8
extension point
EP
point belonging to the location reference extension

3.9
great circle
circle on the surface of a sphere that has the same circumference as the sphere

NOTE The connection between two points on a sphere along the great circle passing through said two points is the shortest connection (airline distance, or distance 'as the crow flies').

3.10
intersection point
IP
core point representing an intersection, located at places where the road section signature at the location changes

NOTE The intersection point is one of the three defined core point types.

3.11
location point
LP
core point that bounds or is located on the location

NOTE Location points may coincide with intersection points or routing points. The start and end of the location is always represented by a location point. Additional intermediate location points may be created to represent the shape of the location. The location point is one of the three defined core point types.

3.12
location reference core
point or set of points that is available in any location reference

NOTE The rules in Clause 8 control the data to be stored in the location reference core.

3.13
location reference extension
additional point or set of points, not belonging to the location reference core, available in a location reference under special conditions

NOTE The rules in Clause 8 specify the conditions under which a location reference extension is to be used and control the data to be stored in a location reference extension.

3.14
next point
point that is directly (topologically) connected to a given point, in a direction that is defined by the defined direction of the location

NOTE A point may have zero or more next points.

3.15
next point relation
ordered pair of points (A, B) for which a direct connection exists from A to B along the path of the referenced location

NOTE In the road network, a direct connection between points A and B exists when point B can be reached from point A via part of the road network, without visiting intermediate points in the location reference. This excludes points connected in a GDF graph via a node representing an intersection-not-at-grade. Such points are not considered to be directly connected.

3.16
parallel carriageway indicator
non-negative integer which indicates if a road segment contains more than one carriageway in parallel in the direction of interest, and how many

3.17
precise geometry description
shape along the location, coded on the most detailed level of the digital map, lying in a corridor with a defined perpendicular distance to the great circle connection between two successive points on a location

3.18
road descriptor
full road number, or a significant substring of the official road name

NOTE The road descriptor is ideally three to five characters in length.

3.19
road network location
location which has a one-dimensional and continuous structure, being part of a road network

NOTE It is a continuous stretch of that road network as realized in the database, which may cover different roads, and may be bounded on either side by an intersection. Alternatively it may be bounded on either side by a position on a road.

3.20
road section signature
road signature
value of the attribute quadruple {functional road class, form-of-way, road descriptor, driving direction}

3.21
routing point
RP
point used to reconstruct the location by route calculation.

NOTE RPs are intended to allow point-based matching to the map database of the end user. When such an RP match is found, the location then can be further reconstructed using the connectivity of the road network as represented in the map database of the end user. The routing point is one of the three defined core point types.

3.22
side road section
road section which is not part of the location to be referenced, but connected to it via an at least trivalent junction

3.23
side road bearing
bearing of the side road section

3.24
side road direction
driving direction of the side road section

3.25
side road signature
road section signature of a side road section

3.26
status location
location to be used to position location-based status information

EXAMPLE A location for speed limit information or traffic level information.

4 Abbreviated terms (and attribute codes)

4.1 Abbreviations

AGORA	Name of a European project 1999-2002 Implementation of Global Location Referencing Approach
DLR	Dynamic Location Reference — also known as DLR1 because this is the first LRM under dynamic profile
GDF	Geographic Data Files — data model, data specification and exchange standard for geographic data for road transport applications
ISO	International Organization for Standardization
ITRF	International Terrestrial Reference Frame
ITRS	International Terrestrial Reference System
ITS	Intelligent Transport System
LR	Location Reference
LRM	Location Referencing Method
LRS	Location Referencing System
NLR	Network Location Reference
RDS	Radio Data System — digital data channel on FM sub carrier
RFU	Reserved for Future Use
SSF	Syntax, Semantics and Framing Structure (TPEG ISO/TS 18234-2)
TMC	Traffic Message Channel — system for broadcast of (digitally encoded) traffic messages on RDS
UML	Unified Modelling Language
VLC	Variable Length Coding
XML	Extensible Markup Language

4.2 Attribute codes

AFR	Accessible For Routing flag
BR	Bearing
CA	Connection Angle
CPI	Connection Point Index
DCA	Distance measure CA
DD	Driving Direction
DMB	Distance Measure Bearing
DSF	DeStination Flag
FC	Functional road Class
FCM	Functional road Class Minimal
FW	Form of Way
IT	Intersection Type
PCI	Parallel Carriageway Indicator
PD	Point Distance
PDM	$D_{perp\text{-}max}$ — Attribute to measure distance on shapes
RD	Road Descriptor
RDI	Road Descriptor of Intersection
RP	Routing Point
SNI	SubNetwork Index

5 Objectives and requirements for a location referencing method

For details, see ISO 17572-1:2008, Clause 4.

For an Inventory of Location Referencing Methods, see ISO 17572-1:2008, Annex A.

6 Conceptual data model for location referencing methods

For details, see ISO 17572-1:2008, Clause 5.

For Examples of Conceptual Data Model Use, see ISO 17572-1:2008, Annex B.

7 Specification of dynamic location references

7.1 General Specification

Dynamic Location Referencing is also known as the AGORA-C method and relies on specific attributes that are mostly available in current digital map databases. Consequently this LRM is adequate for LRSs that have a physical format specification based on GDF. The method relies on real-time access by the software to the original or translated values of the relevant attributes from its own digital map. This LRM will also be called "on-the-fly referencing" because the location reference code can be immediately discarded after internal definition of the location has been decoded. The dynamic location referencing concept is designed to compensate for differences that may exist between the map used at the sending system (the encoding side) and the map on board of the receiving system (the decoding side). Such map differences can be caused by the receiving system using an older map dataset of the same supplier, or vice versa, or the receiving system using a map dataset from a different supplier.

Dynamic Location Referencing is often not as compact as pre-coded location coding. However, it is generally accepted that if dynamic location reference codes can on average stay within 50 bytes for problem and status locations, this would be acceptable in terms of bandwidth occupation. The specification focuses on LRSs for two purposes, and hence provides two building blocks:

Location reference core
The location reference core is applicable to problem and status locations, e.g. road traffic messages. The location reference core is intended to provide location information much like ALERT-C location referencing [10] for which this specification actually intends to provide a light weight Dynamic Location Reference (not requiring pre-coding and the use of location tables). The location reference core prepares a function for additional robustness called precise geometry description in cases where a lack of information elements in the decoders map is expected or under conditions defined in the following clause.

Location reference extension
The location reference extension is applicable to use in routing to destination locations, i.e. the location of interest is to be used as the destination of a route guidance application. The location reference extension augments the location reference core to an extended location reference, in which, the location reference extension is provided to ensure that a path from the location of interest to the nearest part of the road network defined in the location reference core exists.

A dynamic location reference is constructed as a set of information elements, which consists of points and related attributes. All points in both building blocks of the location reference (location reference core and location reference extension) together constitute a linear set, i.e. they form a list where each point in the list except the last one relates to the next point in the list, and to no other points. Each point may have one or more attributes.

On reception of this location reference the receiving system needs to reconstruct the location as intended by the sending system. The encoding rules provided in Clause 8 provide the necessary semantics both for creating the location code at the sending system and for interpreting this code in the receiving system. Thus, the role of the encoding rules is both to provide constraints for selecting and creating this set of information elements at the sending system, and to provide a consistent interpretation basis for the receiving system to reconstruct the location reference as intended by the sending system.

This clause describes the building blocks for the Dynamic Location Reference and specifies different types of attributes. Clause 8 defines the Dynamic Profile LRM as a set of rules. These rules are mandatory, and any Dynamic Profile Location Reference shall adhere to these. Clause 9 defines the minimum requirements for any physical data format, which is for storing Dynamic Profile Location References of this LRM. Annex B describes hints to add optional attributes to the Dynamic Location Reference and proposes a coding procedure, which can serve as a basis for the creation of a coding algorithm. Through application of the rules and the coding procedure the sending system should be able to create a location reference that can be interpreted consistently by a variety of receiving systems if the physical format is well-known. For this reason a first physical format is defined in Annex A giving the opportunity to have at least one exchange format usable for the variety of LRS. If application of an LRM cannot implement the physical format of Annex A, the LRS might specify its own proprietary physical format, still fulfilling all format requirements defined by Clause 9 of this specification. A second physical format is defined in Annex C is specifically optimized for implicit areas and location references with precise geometry description and allows storing the location references in a very size efficient way.

Robustness of the codes is acquired by uniqueness. The information elements used and (certain aspects of) their combination shall be unique for this different parameters are defined as thresholds: e.g. the certain area around a point by the default distance D_{search_area}. These parameters are specified in different rules and the best known values are given in Table 3.

7.2 Location referencing building blocks

7.2.1 General

In 7.2.2 to 7.2.5 the building blocks for dynamic location reference encoding are defined and specified. These are points and attributes.

7.2.2 Points

The basis of the dynamic location reference is a set (or list) of points, which can be described as follows:

Point in general
A point may reference an intersection or may reference a position on the road network away from intersections. The set of points in a location reference constitute a next point relationship such that each point except the last one refers to one and only one other point (its 'next point').

Furthermore points are distinguished as to which part of the location reference they belong to: the location reference core or the location reference extension:

Core Point (CP)
Point belonging to the location reference core, which consists of a combination of three types of core points: location points, intersection points, and routing points.

1. **Location Point (LP)**
   A core point that represents the start, an intermediate, or the end point of the real-world location to be referenced.

2. **Intersection Point (IP)**
   A core point representing an intersection, located at places where the road section signature at the location changes.

3. **Routing Point (RP)**
   A core point used to reconstruct the location by route calculation.

Each core point in the set of points in the location reference core shall represent at least one of the three core point types defined in this International Standard.

Extension Point (EP)
Point belonging to the location reference extension. All points in the set of points in the location reference extension are by definition extension points.

7.2.3 Attributes

7.2.3.1 General

Table 1 lists the defined attribute types for dynamic location references, and their possible values. Note that some attributes relate to points, and other attributes to stretches of road network between points (possibly the whole length of the referenced location). An attribute that describes a characteristic of a linear stretch is linked in the location code to the point that is at the start point of the stretch. The following subclauses define some attributes in detail.

7.2.3.2 Functional Road Class

GDF defines this attribute with the purpose of “a classification based on the importance of the role that the road section or ferry connection performs in the connectivity of the total road network.” It is an enumerated list of 10 different values [5] as follows:

- Main roads: the most important roads in a given network.
- First class roads.
- Second class roads.
- Third class roads.
- Fourth class roads.
- Fifth class roads.
- Sixth class roads.
- Seventh class roads.
- Eighth class roads.
- Ninth class roads: the least important roads in a given network.

NOTE 1 Dynamic Location Referencing uses this classification for distinguishing the parts of the road network, having a certain higher probability of existence in different databases. A standard Map database will deliver this attribute as defined in GDF; however, the attribute is quite differently used between countries and providers. The location referencing method considers this in the rules by leaning only on categorisations with high congruence between different map databases.

NOTE 2 The attribute FC may be not stored in some databases, but the encoder and decoder need to be able to derive it from other information available (speed, lanes, routing tables, etc.). For this purpose, Table B.1 provides an interpretation of the most used functional road classes in Clause 8.

7.2.3.3 Bearing at a point

The bearing at a given point along a location is the angle between the geographic north and the straight line connection from the given point to the intersection of the location with a measuring circle in the location direction (Pm), as depicted in Figure 1. The radius of the measuring circle is defined with the attribute "Distance for Measure of Bearing" (DMB), and if it is not provided, with the (parameter) $D_{m\text{-}bearing}$[1].

The bearing is measured in clockwise direction. The measuring distance of at least attribute DMB, respectively (parameter) $D_{m\text{-}bearing}$ ensures robustness for observed interpretation differences[2].

Once a point along a location has a bearing assigned then there is a natural way of associating one and only one road segment with the point. The road segment associated with the point except for the last point is the road segment leading away from the point in the location direction of the point's bearing. Therefore, if the point is not a junction, then the associated road segment is simply the road segment on which the point is located. If the point represents a junction, then the associated road segment is one of the road segments incident at this junction and leading away from it in the direction of the point's bearing (see Figure 1).

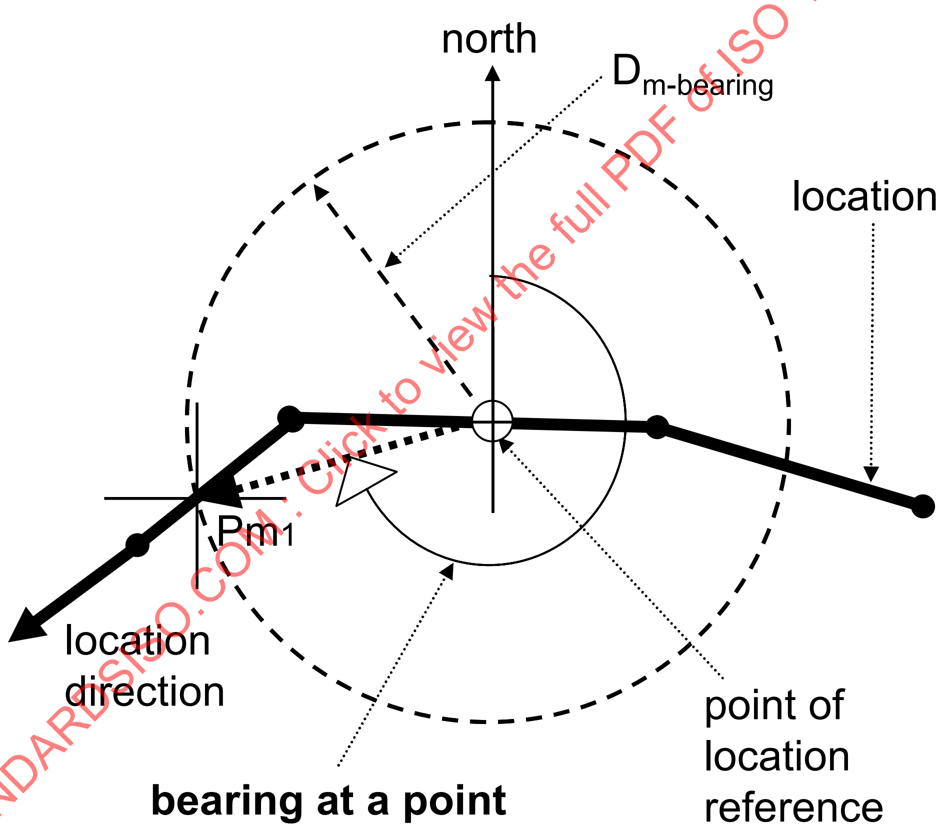

Figure 1 — Bearing at a point (general case)

In a case where a point is the last point of the location, the bearing is the angle from the intersection of the circle and the location reverse to the location direction (see Figure 2).

1) See Table 3 for values of defined parameters.

2) Road segments of less than 10m length do not provide sufficient real-world semantics. Frequently differences occur between maps of different map vendors due to interpretation differences allowed by GDF.

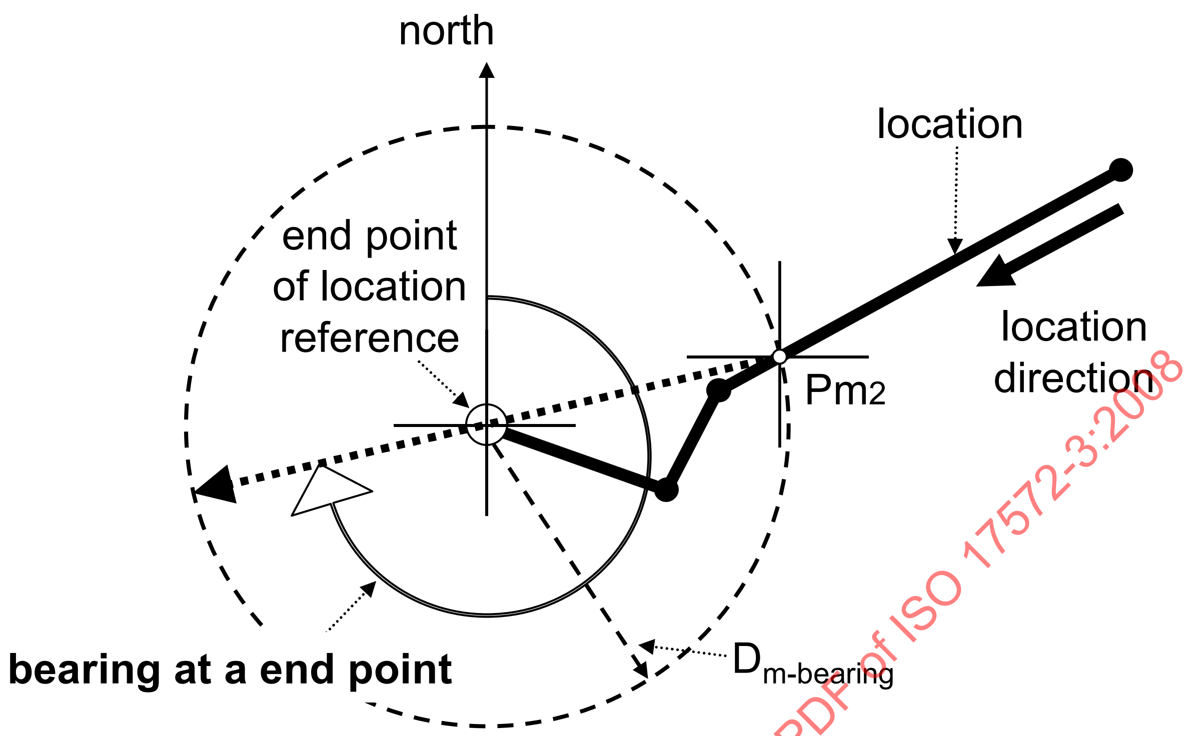

Figure 2 — Bearing at a point (special case for last point)

7.2.3.4 Connection Angle

In a case where in addition to the bearing a side road bearing is calculated, the attribute stored is the connection angle. The connection angle is the difference of side road bearing (Ps) and bearing at a point (Pm). See Figure 3. Because especially in intersections differences in maps are potentially bigger the connection angle is calculated with a higher measuring distance DCA than for the bearing. The attribute "Distance for measuring of Connection-Angle" (DCA) respectively (parameter) $D_{m\text{-co-angle}}$ [3)] ensures robustness for observed interpretation differences[4)] at intersections.

7.2.3.5 Location Direction

A location reference has an implicit direction, which is defined by the order of the points in the set of points that constitute the location reference core. The positive direction or from-to direction is from the first point in the set to the last point in the set. Note that this direction shall coincide with the direction to be referenced, if only one direction is referenced.

The attribute Location Direction (LD) has the following values:

— *Aligned* The location has one direction, corresponding to the implicit direction defined by the order of the points.

— *Both* The location has two directions, both in the implicit direction and in the reverse direction.

3) See Table 3 for values of defined parameters.

4) Road segments of less than a given length do not provide sufficient real-world semantics. Frequently differences occur between maps of different map vendors due to interpretation differences allowed by GDF.

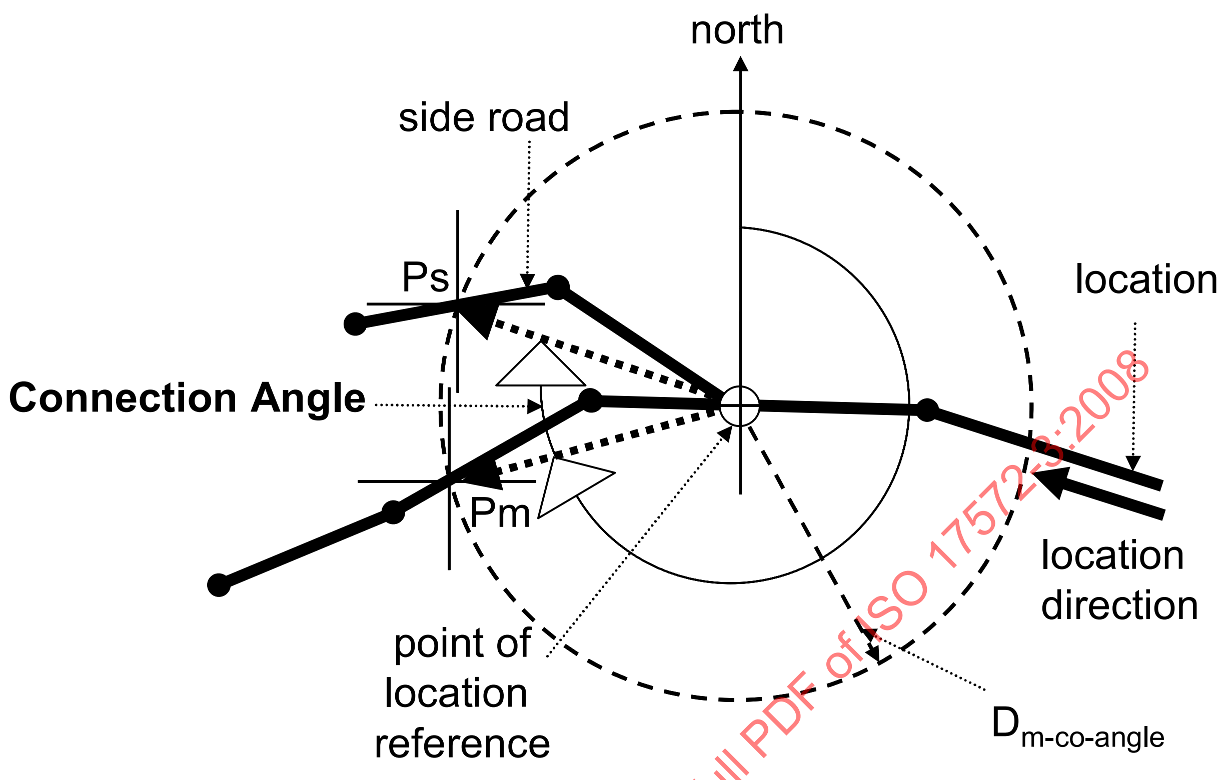

Figure 3 — Connection angle at a point

7.2.3.6 Location Type

A location reference defines the type of the object to be referenced. The decoder will use this information to optimise its effort of decoding the location by a first presumption what type of object it is supposed to search for in its database. For this reason the following values are defined (based on GDF and adapted for this context) with given purpose:

— *Intersection* The location is part of an intersection.

— *Restricted access road* The location is part of a road that is subject to restricted access for certain categories of road users (e.g. pedestrian zone).

— *Ferry* The location is part of a ferry connection.

— *Settlement* The location is part of a settlement (i.e. area with a residential, recreational, industrial or military character) .

— *Point-of-interest* The location is a point of interest (e.g. a specific route destination for a service).

— *Road* The location is a part of a road, without specific character, including motorways or other types of limited access roads.

7.2.3.7 Driving Direction at a Point

The driving direction (DD) at a point (along the location) expresses the legally permissible driving direction under normal conditions for passenger cars[5] at the point relative to the bearing at the point. Stated in terms of the rule above it expresses the driving direction of the road segment leading away from the point along the

5) In GDF 4.0 terminology the attribute "Direction of Traffic Flow" for vehicle type passenger cars. In a revision of the GDF standard (GDF5.0) this attribute is replaced by the attribute "Conditional Traffic Flow" which specifies direction of traffic flow per direction separately.

path in terms of the point's bearing. Alternatively, using the definition of the bearing of a point along a path of non-zero length, it can be determined as follows:

If the point is not the endpoint of the path, then the driving direction pertains -in terms of most detailed level geometry- to the driving directions of the road segment pointing away from the point in location direction.

If the point is the endpoint of the path, then the driving direction pertains -in terms of the most detailed level geometry- to the driving directions of the road segment pointing towards the point in location direction.

The driving direction can have the following values:

— *none* driving is not allowed on the road segment.
— *aligned* the only allowed driving direction coincides with the bearing.
— *reverse* the only allowed driving direction is opposite to the bearing.
— *both* both driving directions are allowed on the road segment.
— *undefined* driving direction information is not available in the database.

7.2.3.8 Accessible For Routing flag

The Accessible For Routing (AFR) flag at a point represents information for routing on road segments connected to a point. The flag is set to 0 (false) if the segment following the point is not accessible in the location direction. The flag is set to 1 (true) if the segment following the point is accessible in the location direction.

NOTE The attribute Driving Direction is related to the Accessible For Routing flag as follows: DD values 'none' and 'reverse' map to AFR value 0 (false), DD values 'aligned', 'both' and 'undefined' map to AFR value 1 (true)

7.2.3.9 Parallel Carriageway Indicator

In case of a road, that has multiple carriageways for the same driving direction, which cannot be distinguished by other attributes of the given rules, the parallel carriageway indicator (PCI) is used to discriminate that particular part of the road. Parallel carriageway indicator counts the number of carriageways (PCIN) either in the horizontal direction or in the vertical direction (indicated by PCIT, values either "horizontal" or "vertical") and defines the sequence of parallel carriageways and a pointer to which of them is taken (PCII, the PCI index).

Some databases may lack of the complex information of a road section constructed out of different carriageways. For this case PCI defines the search area (D_{search_area}) as discriminating selector. All carriageways having all road section signature attributes equal are taken into account for the sequence of parallel carriageways. The PCII is counted starting with value 1 for the first carriageway.

Where the national convention is for all vehicular traffic proceeding in the same direction to be on the right side of the road, counting is performed with an angle of 90 degrees clockwise with respect to the bearing of the carriageway in question and from the leftmost carriageway for type "horizontal", or from the bottom most carriageway for type "vertical". Where the national convention is for all vehicular traffic proceeding in the same direction to be on the left side of the road, counting is performed with an angle of 90 degrees counter-clockwise with respect to the bearing of the carriageway in question and from the rightmost carriageway for type "horizontal", or from the top most carriageway for type "vertical".

The search area enhancement indicates how much the default search area of (parameter) D_{search_area} is enhanced to find the all carriageways in question. The enhancement ensures, that for roads having a larger extent than D_{search_area}, the whole sequence of carriageways are counted. See Figure 4 as an example.

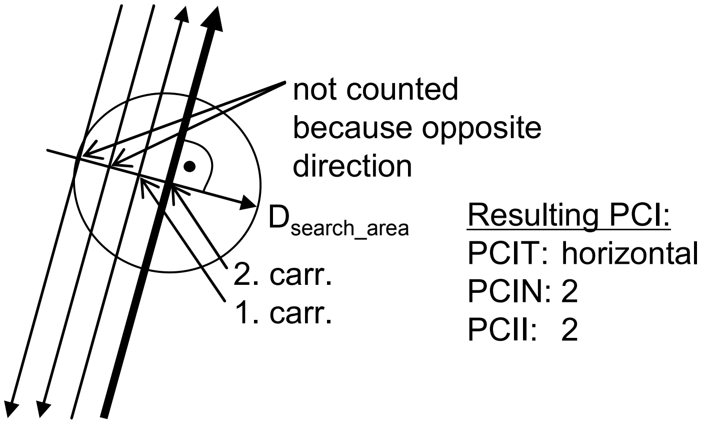

Figure 4 — Example of parallel carriageway indicator

7.2.3.10 Perpendicular distance

To follow the shape of the road in a narrow corridor the attribute perpendicular distance is measured. The perpendicular distance is defined as the maximum distance between the straight line connection, between two given points on a road segment and the shape of the road segment given from the most detailed level geometry of the database. In RULE-10 and RULE-31 the perpendicular distance is required to be less than a certain value.

Figure 5 illustrates for any point of the road elements on the road section between successive points A and B of the original path of the location, the perpendicular distance D_{perp} to the straight line connecting points A and B shall be less than attribute (PDM) respectively (parameter) $D_{perp\text{-}max}$. If not, then one or more intermediate location points need to be inserted. The open circles reflect shape points which are given from the most detailed level geometry of the map database of the encoder side.

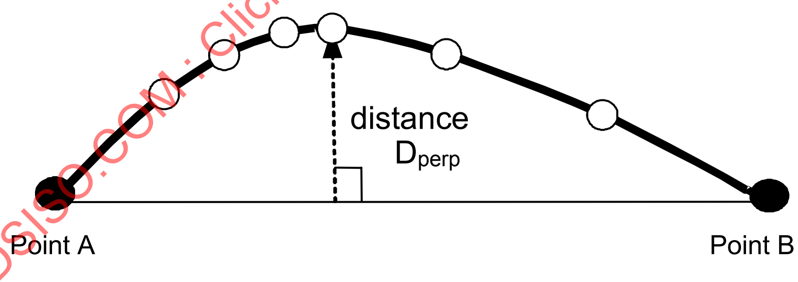

Figure 5 — Calculation of perpendicular distance

7.2.3.11 Distance to Next Point

Between a point and the next point along the path of the location reference the driving length is accumulated. The attribute Point Distance (PD) at a point indicates the driving length from this point along the path to that point having again the attribute PD or until the last point of the location reference.

7.2.4 Next-point relationship

A point B is a next point to another point A only if there is a direct connection between A and B along a path of a location reference. Refer to Figure 6 for illustration. Point A is connected with B and B is connected with C but A is not directly connected with C. The transitive closure of the next point relation indicates whether from a point A the point C can be reached following the path as defined by a sequence of one or more next point relations. For example in Figure 6 the transitive closure also includes the directed relation from point A to point C.

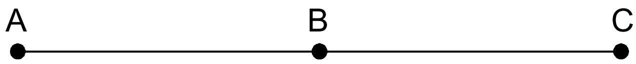

Figure 6 — Explanation of direct connection between points

7.2.5 Attribute type list

The following table defines the general attribute types and their value specification.

Note that in the annexes describing a physical format the value definition for some of the attributes may be more compact (reduced resolution or reduced value set) than in the table below. This is done to achieve small code size in the physical format, while maintaining high hit rate.

Table 1 uses the following notations and base data types:

- *[x-y]* definition of range minimum includes x and maximum includes y; missing x or y defines open range.
- *Boolean* value having only two states namely true or false.
- *string* string of textual characters, usage is given by amended format description.
- *enumeration* enumeration of fixed definitions distinctly indicated by one integer value.
- *integer* signed integer value within a range from negative to positive values.
- *unsigned* unsigned integer value with non-negative values only.
- *structure* set of several sub data types grouped together.

Table 1 — Defined attribute types and attribute values

Abbreviation	Attribute name	Data type	Description	Description of values and range
AFR	accessible for routing	Boolean	Indicates whether the road segment following a point is accessible for routing or not (see 7.2.3.8 for detailed definition).	- *false*, if not accessible - *true*, if accessible
BR	bearing	unsigned	The attribute defines at a given point along a location, the angle between the geographic north and the straight line connection from the given point to the intersection of the location with a measuring circle in the location direction. (See 7.2.3.3 for further definition)	(in degree)
CA	connection angle	integer	The attribute defines the angle difference between side road bearing and bearing at a point. (See 7.2.3.4 for further definition)	(in degree)
CPI	connection point index	unsigned	The attribute defines the index of the core point in a location reference that a location reference extension or subnetwork location connects to.	
DCA	distance for measure of connection angle ($D_{m\text{-}co\text{-}angle}$)	unsigned	Optional value in case that not parameter $D_{m\text{-}co\text{-}angle}$ was used.	(in metres)
DD	driving direction	enumeration	The attribute defines the allowed legal driving direction (see 7.2.3.7).	- *none*, if driving is not allowed on this road segment. - *aligned*, if the only allowed driving direction coincides with the bearing. - *reverse*, if the only allowed driving direction is in opposite direction of the bearing. - *both*, if both driving directions are allowed on the road segment. - *undefined*, in case that the attribute is not existing in the used database [6].
DMB	distance for measure of bearing ($D_{m\text{-}bearing}$)	unsigned	Optional value in case that not parameter $D_{m\text{-}bearing}$ was used.	(in metres)

6) Note: The values 0-3, i.e. none, aligned, reverse and both, are defined as like GDF. For compatibility reason “undefined” has been specified as 4, which may be stored as another bit in the physical format.

Table 1 (*continued*)

Abbreviation	Attribute name	Data type	Description	Description of values and range
DSF	destination flag	Boolean	Destination flag is set to true in case that a linear location reference is used as a destination location to show which point determines the intended destination object.	- *true* in case of destination shall be positioned at this point.
FC	functional road class	enumeration	Road classification based on the importance of a road or ferry connection in the connectivity of the total road network as defined 7.2.3.2. If the database lacks of this attribute a subsiding, similar classification of importance is needed, refer then to Annex B.3.	- FC 0 - Main road - FC 1 - First class road - FC 2 - Second lass road - FC 3 - Third class road - FC 4 - Fourth class road - FC 5 - Fifth class road - FC 6 - Sixth class road - FC 7 - Seventh class road - FC 8 - Eight class road - FC 9 - Ninth class road
FCM	minimal functional road class (FC_{min})	enumeration	Most important road class of road elements comprising the original path of the location extension for at least a stretch of length of $D_{search\text{-}area}$ as defined in attribute FC.	(values as defined for attribute FC)
FW	form of way	enumeration	Physical road type. This attribute is related to the corresponding GDF attribute. However, the attribute here is restricted to a smaller list of definitions. In case that the database lacks this attribute the value “undefined” shall be used.	- *undefined* - *motorway* - *multiple carriageway (which is not a motorway)* - *single carriageway* - *roundabout circle* - *traffic square* - *enclosed traffic area* - *slip road* - *service road* - *car park entrance or exit* - *service entrance to or exit* - *pedestrian zone* - *walkway*

Table 1 (*continued*)

Abbreviation	Attribute name	Data type	Description	Description of values and range
IT	intersection type	enumeration	Categorisation of an intersection at the given point. If the database lacks of this information the value “undefined” shall be used.	- *undefined*[7)] - *interchange involving motorways or other limited access roads (complex)* - *roundabout* - *complex road crossing not of type "interchange involving motorways or other limited access roads" or "roundabout"* - *simple road crossing* - *traffic square* - *bivalent intersection* (i.e. point where one or more attributes change)
LD	location direction	enumeration	The attribute defines the direction of the location (see 7.2.3.5)	- *aligned*, if only the direction coinciding with the direction of the location is referenced - *both*, if both directions of the location are to be referenced
LPF	location point flag	Boolean	Indicates, that the corresponding point is flagged as location point	- *true*, in case that point is a location point.
LT	location type	enumeration	Specifies the location type of the location that is referenced (defined in 7.2.3.6)	- *intersection* - *restricted access road* - *ferry* - *settlement* - *point-of-interest* - *road*
NIT	number of intermediate intersections	unsigned	Number of intersections that follow a point until the next point of same type excluding the intersection of that point itself.	
PCI	parallel carriageway indicator	structure	The attribute indicates which carriageway is chosen if there are several physically separated carriageways leading in the same direction. The attribute consists of five parts combined in one structure refer to 7.2.3.9 for detailed description:	structure see sub sections:

7) Note: The physical format specification may define “undefined” as absent.

Table 1 *(continued)*

Abbreviation	Attribute name	Data type	Description	Description of values and range
PCISF		Boolean	1. Indicator whether default search area is to be used or optional search area enhancement (item #5) is specified	- *false*, if no search area enhancement is defined - *true*, if the search area enhancement is defined
PCIT		Boolean	2. Parallel carriageway indicator type.	- *false* for type *horizontal* - *true* for type *vertical*
PCII		unsigned	3. Parallel carriageway sequence number.	
PCIN		unsigned	4. Number of carriageways	
PCIS		unsigned	5. (Optional) Search area enhancement distance in metres indicating how much the default search area (specified as the parameter $D_{search\text{-}area}$) has been enlarged (radius = $D_{search\text{-}area}$ + search area enhancement distance).	(in metres)
PD	point distance	unsigned	Network (driving) distance as defined by the next point relationship of a given point to the next point along the path that has a PD attribute or until the end of the location.	(in metres)
PDM	perpendicular distance maximum ($D_{perp\text{-}max}$)	unsigned	Perpendicular distance as maximal distance between the approximated points and the original path of the most detailed level of the map in metres (see 7.2.3.10).	(in metres)
RD	road descriptor	string	All characters of the full national road number of the road, if it exists, otherwise at most five characters of the (GDF attribute) official name of the road. Less than five characters may be chosen, with a minimum of three, if the resulting descriptor is a unique string in the search area of a radius of $D_{search\text{-}area}$. In case that neither road name nor road number is available the string is left empty.	For string format definition refer to part 1 annex E or to the corresponding physical data format definition.
RDI	road descriptor of intersection	string	Sometimes a complex intersection like a roundabout has a name of its own. This may be represented by this attribute in an attributes list for the point.	For string format definition refer to the according physical format definition.
SNI	subnetwork index	unsigned	The attribute defines the index of the location reference that a subnetwork location connects to.	

8 Encoding rules

8.1 Introduction

In this clause the Dynamic Profile Encoding Rules are specified. In 8.2 the general point representation and selection rules are specified, common to both the core and location reference extension. In 8.3 and 8.4 respectively, the rules for the location reference core and the location reference extension are specified.

The location reference core provides a minimum specification for a location reference, common for all locations. For use in intelligent transport systems, the location reference core provides a complete and sufficient set of rules for all locations to be used as problem and/or status locations. In the location reference extension then additional rules are provided especially for locations to be used as destination locations. These additional rules complement and extend the rules for the location reference core for such locations.

Each point used in the location reference can be given additional attributes in Table 1 to additionally discriminate the location reference more precisely than defined in the following rules.

NOTE In some cases parts of the road signature may be not available in the encoding map. For such case a value like “undefined” is provided per attribute. In case that the encoder lacks of attributes it clearly reduces the feasibility for the decoder to interpret the location reference dramatically. However, it might be the only way to enable referencing, it resulted in the definition of “mandatory if exist” attributes, but with a clear disclaiming of liability of 95% decoding success rate.

8.2 General point representation and selection rules

RULE-01 Each point shall be represented by a coordinate pair in ITRS longitude and latitude coordinates, reflecting its position on the surface of the earth.

NOTE In this version of this International Standard the coordinates resolution for common ITS applications is in the order of magnitude of 10 micro degrees. However, for future applications (e.g. pedestrian navigation) a higher resolution may be required. This International Standard specifies therefore a high resolution format (in the order of magnitude of 1 micro degree). The resolution shall be signalled in the physical format.

RULE-02 Points are selected along the location in one direction, such that they form a next-point-relationship. If only one direction is to be referenced, the point selection direction shall coincide with the direction to be referenced.

NOTE Each point in a location reference can be addressed using an index. This index is implicitly defined by the order of the points and counted from the first point of a location reference core, starting at value zero. The index is used by the attribute PCI in RULE-29, RULE-41 and RULE-42 to connect a location reference extension or subnetwork location to a point of a location reference core.

RULE-03 Each point shall be located on the path which constitutes the nominal or assumed centreline of the location. Each point shall be selected from the most detailed level of the digital map database.

RULE-04 In case of a road, which is composed of a number of carriageways that are physically divided, the path follows the carriageway to be referenced, and points are located on that carriageway.

8.3 Location reference core encoding rules

In this clause the rules for the location reference core are listed. The location reference core provides a minimum specification for a location reference, which will form the common part for all locations to be referenced. The location reference core is encoded in the location reference core representation.

8.3.1 Location selection

RULE-05 The location reference indicates the referenced location direction or directions (attribute LD). In case of absence of attribute LD in the location reference, location direction is expected to be aligned. If the location to be referenced or part of it is separately digitised for each direction, or contains complex intersections, each direction shall be considered as a separate location, and be separately coded if both directions are to be referenced.

RULE-06 The location type (attribute LT) of the location to be referenced shall be indicated.

8.3.2 Location reference core point selection

RULE-07 Each core point of the location reference core shall be at least one of the three point types defined in 7.2.2: a location point, an intersection point, or a routing point, or a combination of these three core point types.

NOTE Each point is characterised by the used attributes. The location point is signalled explicitly, routing and intersection points are signalled by their attributes.

RULE-08 All points of the location reference core together shall constitute a logically ordered set (list), from the defined start point of the location to the defined end point, where each point has topologically (i.e. in the network) a direct connection (i.e. not via another point in the list) with the next point in the list. In this way the list implicitly defines the next-point-relationship (see 7.2.4).

8.3.3 Core point selection - location points

RULE-09 All selected points lying on the path of the location to be referenced shall be indicated (attribute LPF) as location point. As a minimum the start and end point of the real-world location to be referenced shall be represented by a location point.

NOTE The location reference may include points on the path of the location (like interim intersections), that do not originate from the rules. Such points are coded with type "location point" only.

RULE-10 The segment length along the original path of the location between successive location points shall not exceed the great-circle (airline) distance by more than the greater of 10 m or 5 % of the great-circle (airline) distance. In case that a point needs to be inserted, it shall be placed at the position of greatest perpendicular distance and both sub segments shall apply to this rule again.

In situations according to the following conditions:

— If attributes FW, RD and DD are missing (i.e. not present or not reliable or not usable) for the given segment

and

— for an LP, another road section exists within (parameter $D_{search\text{-}area}$), or

— for an IP, another IP of a different intersection exists within (parameter $D_{search\text{-}area}$), or

— for an RP, another RP is identifiable on a different road section being within (parameter $D_{search\text{-}area}$) and not having a unique bearing as defined in RULE-25,

precise geometry description shall be used to determine the need of inserting location points, to avoid ambiguity.

Precise geometry description requires that any part of the actual segment between successive location points shall have a perpendicular distance D_{perp} to the straight line connection of the points of at most of value $D_{perp\text{-}max}$. The value of $D_{perp\text{-}max}$ defined as attribute PDM along that segment shall be provided at the first point belonging to the segment and has a default value of (parameter) $D_{perp\text{-}max}$.

The precise geometry description shall at least consider a stretch of (parameter) D_{margin} length. In case of complex intersections the margin for precise geometric description shall be counted away from the IP or RP chosen. The end of the precise geometry description segment is defined by that point setting attribute PDM to 0.

NOTE The shape of the location hence is restricted to lie within a narrow corridor around the road sections represented in the location.

8.3.4 Core point selection - intersection points

RULE-11 Each point along the location at which the road section signature changes shall be represented by an intersection point. If the last point of the location is an intersection, it constitutes an intersection point, even if the road section signature does not change at that point. In this particular case the intersection point may not carry all attributes that did not change, for example, for bandwidth saving reasons. The first location point of the set of points is in any case an intersection point, even if it is not an intersection, to provide the signature for the first road section. In case that the database lacks of attributes the value “undefined” as specified per attribute shall be used. If the attribute values used change to “undefined” due to the database, there shall not be an insertion of an intersection point. If only the attribute FW change to “roundabout circle” or "traffic square" and the location continues after these intersections with a road section having the same signature as the section before the intersection, there shall be no insertion of an intersection point.

NOTE 1 The road section signature generally changes at an intersection, but it may also change without an intersection being present, e.g. a name changes part way along a road, this intersection point is then defined as a bivalent intersection. In case that a road has more than one name or number, the one taken does determine if a new IP is to be set.

NOTE 2 In case that an intersection has a name different from any road connected to it (e.g. the name of a roundabout), the attribute RDI may be added as an optional attribute.

RULE-12 An intersection point is located at the intersection which it represents. If the intersection is a simple crossing, the position of its junction is chosen as the intersection point. If the intersection is a complex intersection, a point is chosen within the intersection, on the path of the location, at the first junction of the intersection counted in the direction of the location, where the signature changes.

NOTE Figure 7 shows examples where to apply intersection points in complex intersections.

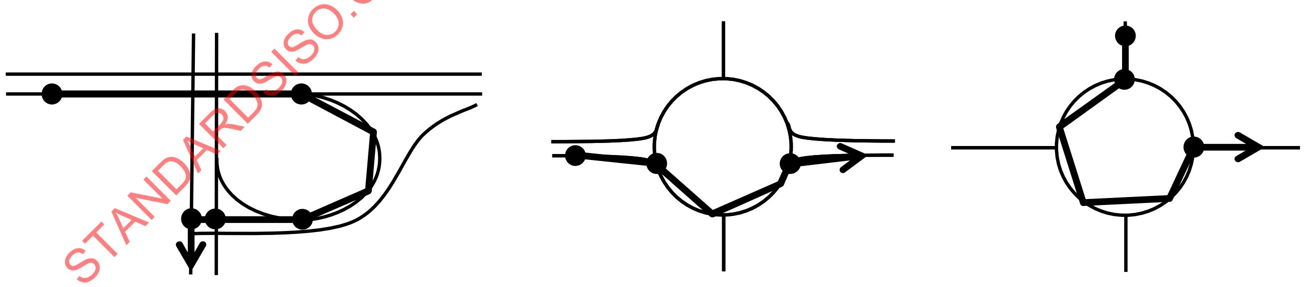

Figure 7 — Examples for positioning of intersection points

RULE-13 If the end point or the start point of the location to be referenced is less than distance (parameter) $D_{search\text{-}area}$ away, along a path in the road network, from an intersection, which is not part of the location (measured along a path in the road network in the location direction or in reverse location direction), then this intersection point shall be included in the location reference as the last point, or the first point respectively.

NOTE 1 Figure 8 shows an example how to measure the distance at the beginning and end of a location; the distance being smaller than $D_{search\text{-}area}$ would cause inserting the intersection point as first point of the location reference.

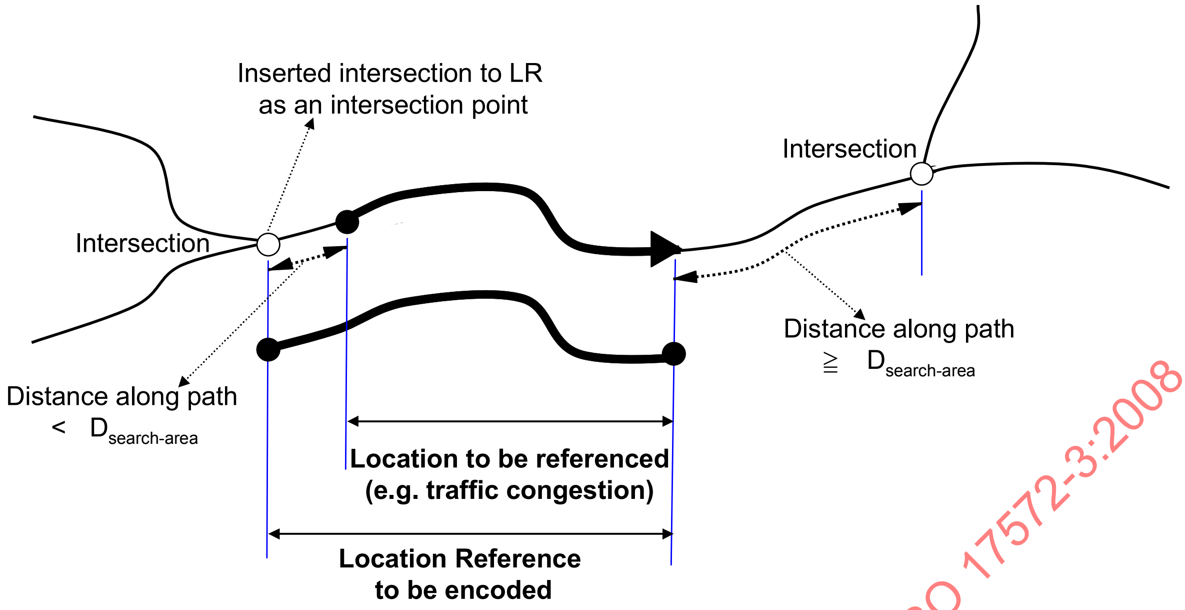

Figure 8 — Example for RULE-13

NOTE 2 Intersections have clear real-world semantics, and provide an added measure of robustness for geometric inaccuracies. RULE-13 is intended to enable relative positioning of start and end point of the desired location. It is a known issue that maps may differ especially in the representation of intersections. For the decoding it is important not to start on the wrong side of the intersection, because this would wrongly include (or exclude) the intersection. Figure 9 shows a case where an intersection is located at a different position in the decoder map. For this situation the decoder should decide to include (or exclude) the remainder of the referenced location on the wrong side from the intersection relative to its internal position of the intersection. Attribute PD may be used to explicitly define known length (offset) between IP and begin of location. Shifting of location is done relative to the matched intersection either with the given offset or by cutting the extent such that the location does not refer to the wrong side of the intersection unintentionally.

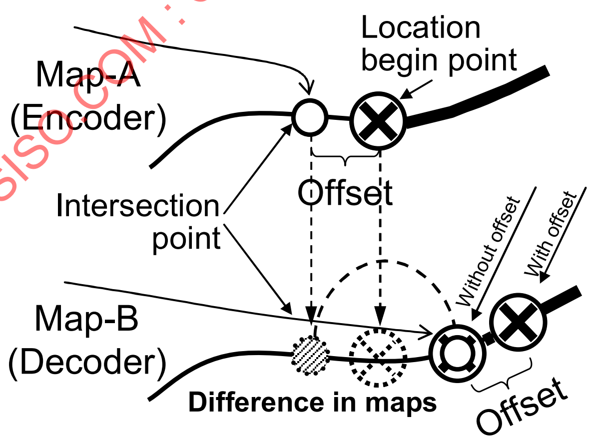

Figure 9 — Example for moving beginning of location reference core relative to the intersection

8.3.5 Core point selection - routing points

RULE-14 A routing point shall be chosen only there where the road segment determining the routing point bearing, and in case of an intersection point, also the road segment determining the side road bearing, have a length of at least for bearing (parameter) $D_{m\text{-bearing}}$ and for side road bearing (parameter) $D_{m\text{-co-angle}}$.

NOTE See definition of bearing, 7.2.3.3. Road segments of less than a given length do not provide sufficient real-world semantics. Frequently differences occur between maps of different map vendors due to interpretation differences allowed by GDF.

RULE-15 At least the first and the last core point shall be routing points.

NOTE 1 The location reference shall contain a sufficient and minimal number of routing points to allow unambiguous reconstruction of the location in the map of the sending system.

NOTE 2 In case that the road segments of the start or end point of the location do not meet the requirements of RULE-14, The RP is chosen at the nearest point before the start or behind the end point that meets the requirements of RULE-14.

NOTE 3 Additional core points can be introduced following RULE-13.

RULE-16 Each routing point shall identify a unique point on a unique road segment on the map of the sending system. Uniqueness is determined on the combination of all relevant attributes defined in 7.2.3 in the encoder map. Uniqueness shall hold within a default radius of (parameter) $D_{search\text{-}area}$, or, if present, the enhanced value as specified in RULE-27 (the parallel carriageway indicator, attribute PCI). The functional road class attribute shall only be considered a unique identification when there is at least a road class difference of two levels with other candidate road segments.

NOTE The interpretation of importance of a road section may differ in countries between different areas or even between different map databases. To make the rule more robust it is sufficient if a two-level difference is used.

RULE-17 The weight factor per functional road class to be used in calculation of weighted distance for decoding purposes shall be defined as in Table 2. The weighted distance is defined as weight factor × distance. In case of lacking attribute FC as defined by GDF a similar cost calculation shall apply, which ensures that the routing does not use shortcuts through unusual, small streets. The alternative definition in B.3 can be used to derive the functional road class.

Table 2 — Distance weight factors per functional road class

FUNCTIONAL ROAD CLASS	main roads	first class roads	second class roads	⩾ third class roads
WEIGHT FACTOR	2	3	4	6

NOTE The rationale for this weight factor is to allow very succinct reference codes for long highway segments, since all other non-highway segments, even when parallel to the main road, will have at least a 50% larger weighted distance.

RULE-18 The segment of the referenced location between two successive routing points shall be considered uniquely defined if the following three criteria are met:

1. The segment length shall not exceed twice the great-circle (i.e. the shortest route between two points located on a sphere) distance between the two successive routing points.

2. The segment has the lowest weighted distance between those routing points.

3. Any completely disjoint alternative (except for start and end point) for part (or whole) of the segment shall have a weighted distance that is at least 25% greater than the weighted distance of the part that it provides an alternative to.

NOTE Figure 10 illustrates criterion 3 of this rule. The black dots in the depicted road network represent intersections. For any segment A to F, any alternative segment (B-D-E) that does not travel over any part of the path of the location to be referenced (A-B-C-E-F), is expected to have a weighted distance of at least 25% more than the intended segment.

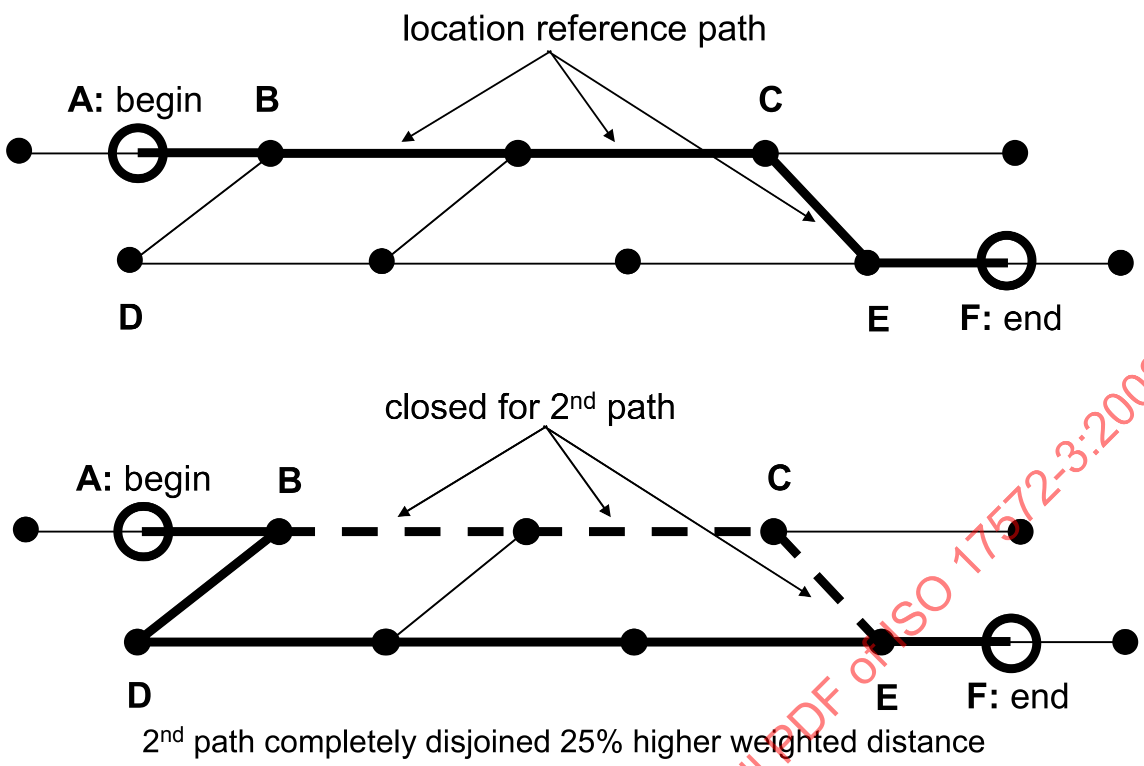

Figure 10 — Illustration of RULE-18 part 3

8.3.6 Intersection point attributes

RULE-19 For each intersection point but the last one if this last one is at an intersection at the end of the location, the road section signature of the following road section shall be given (attributes FC, FW, RD, and DD). In case that the database lacks of attributes the value "undefined" as specified per attribute shall be used.

RULE-20 If a national road number exists for a road section (i.e. following country-wide numbering scheme, not internationally defined), the road descriptor shall contain all characters of the full road number. Otherwise the official name of the road section shall be encoded as a significant substring of three to five characters. The part of the official name that is used to encode the road descriptor shall be unique within a distance of (parameter $D_{search\text{-}area}$) around the extent of the part of the road section that belongs to the location to which the road descriptor pertains (excluding other road sections being the direct extension of that road section). If both, road number and name do not exist, the attribute shall be kept empty.

NOTE 1 In Germany for example the road number would be A40, or B11. The road numbering scheme of the country is to be followed. Thus A40 is to be used, not E25 (the European highway road number). In case of multiple applicable road names and/or numbers, the most commonly used should be taken into account.

NOTE 2 The significant substring is not necessarily the first three to five characters of the road name.

NOTE 3 If there are different names or numbers for each road side or direction of one single road, the name in the direction of the location shall be taken.

RULE-21 For each intersection point but the last one if this last one is at an intersection at the end of the location, the number of intermediate intersections (attribute NIT) along the road section until the next intersection point shall be given.

RULE-22 For each intersection point the intersection type (attribute IT) shall be given, except of the case that the first IP is not positioned on an intersection or a road signature change. In case the database lacks the attribute IT, the value "undefined" as specified for this attribute shall be used.

8.3.7 Routing point attributes

RULE-23 For each routing point direction information shall be given, i.e. the bearing (attribute BR) and accessible for routing flag (attribute AFR). In case that the database lacks of direction information the value "true" shall be used. In this case IP attribute driving direction (DD) of the location reference designates the fact that direction information is not usable.

NOTE Subclauses 7.2.3.3, 7.2.3.7 and 7.2.3.8 provide the exact definitions of bearing, respectively driving direction and accessible for routing flag.

RULE-24 For each routing point located on an at least trivalent junction, side road signature (connection angle and accessible for routing flag) shall be given (i.e. attributes CA and AFR). If more than one side road section is connected to the junction, side road signature shall be given for that side road section which ever has the smallest absolute value of the connection angle either in or against direction of bearing (see Figure 11).

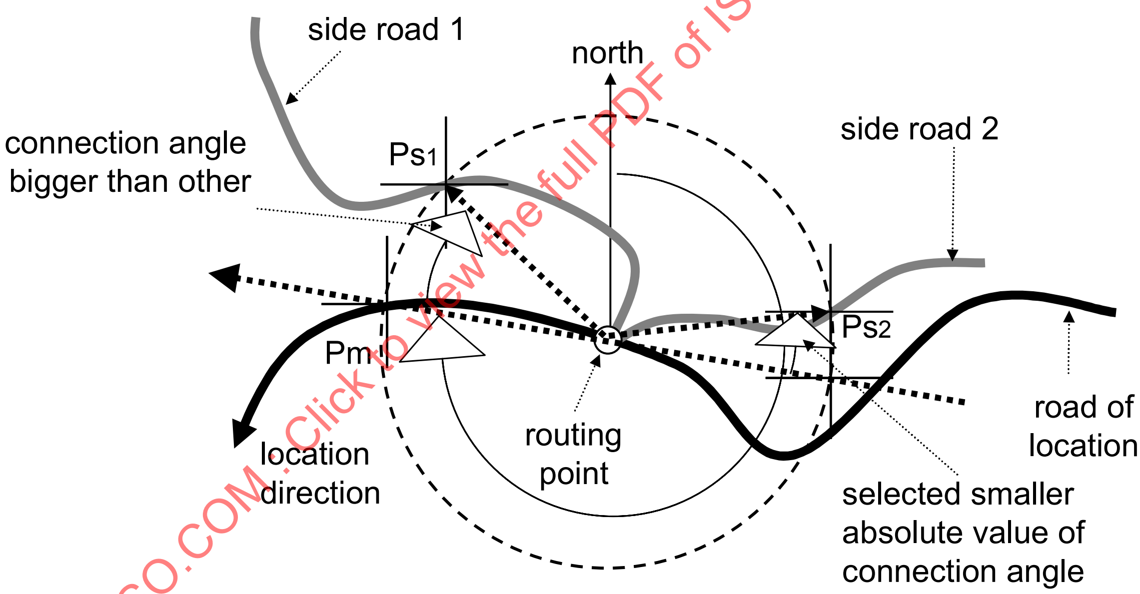

Figure 11 — Illustration of selection of connection angle with smallest absolute value either in or against direction of bearing

RULE-25 The routing point bearing can be considered to be unique as an identifying attribute only within the tolerance (parameter) α. All other locations with a bearing within the range of the bearing of the intended location $\pm\alpha$ should be considered equivalent with respect to the routing point bearing.

RULE-26 In case of a linear location, the location reference shall contain for each, but the last, routing point the distance along the location to the next routing point, as the attribute point distance (attribute PD), associated with the routing point.

NOTE 1 The above mentioned attribute point distance represents the travelling distance along the road network, i.e. the distance travelled on the segment(s) between successive routing points, and NOT the great-circle ('airline') distance.

NOTE 2 In case intermediate core points between two successive routing points have the attribute PD, the distance between these routing points is the sum of point distances signalled by the intermediate core point PDs.

RULE-27 If a road has multiple carriageways for the same driving direction, which cannot be distinguished by other attributes (e.g. functional road class, form of way or road descriptor) and only one of these carriageways shall be coded, a parallel carriageway indicator shall be given (attribute PCI see 7.2.3.9) otherwise the location reference indicates to refer to all carriageways mentioned.

8.3.8 Location reference core encoding parameters

The dynamic profile encoding rules for the location reference core as specified in previous subclauses of this clause contain six parameters.

Table 3 — Parameter value specification for the location reference core

Parameter	Specification value		Used in rule
$D_{perp\text{-}max}$	16 m	if FC has the value “main road”	RULE-10, RULE-31
	8 m	if FC has the value “first or second class road”	
	4 m	if FC has a value less important than “second class road”	
D_{margin}	300 m		RULE-10
$D_{search\text{-}area}$	150 m		RULE-13, RULE-16, RULE-20
$D_{m\text{-}bearing}$	25 m		RULE-14, RULE-23, RULE-25
$D_{m\text{-}co\text{-}angle}$	50 m		RULE-14, RULE-24
α (tolerance)	45°		RULE-25

8.4 Location reference extension encoding rules

8.4.1 General

This clause specifies the additional rules for the *location reference extension* for location referencing. The *location reference extension* is intended for referencing *destination locations*, i.e. locations which are to be used as destination of a route guidance application. In this case it is important to allow a location, which may not be present in the decoder's map, to be connected to the major road network, which will always be present in the decoder's map, such that at least a rudimentary guidance can be given. Figure 12 shows an example of a description guiding to a destination being in an area of e.g. a national park. The receiver shall be guided to find its route through the not digitized area by following the path coded in a precise geometry description shape.

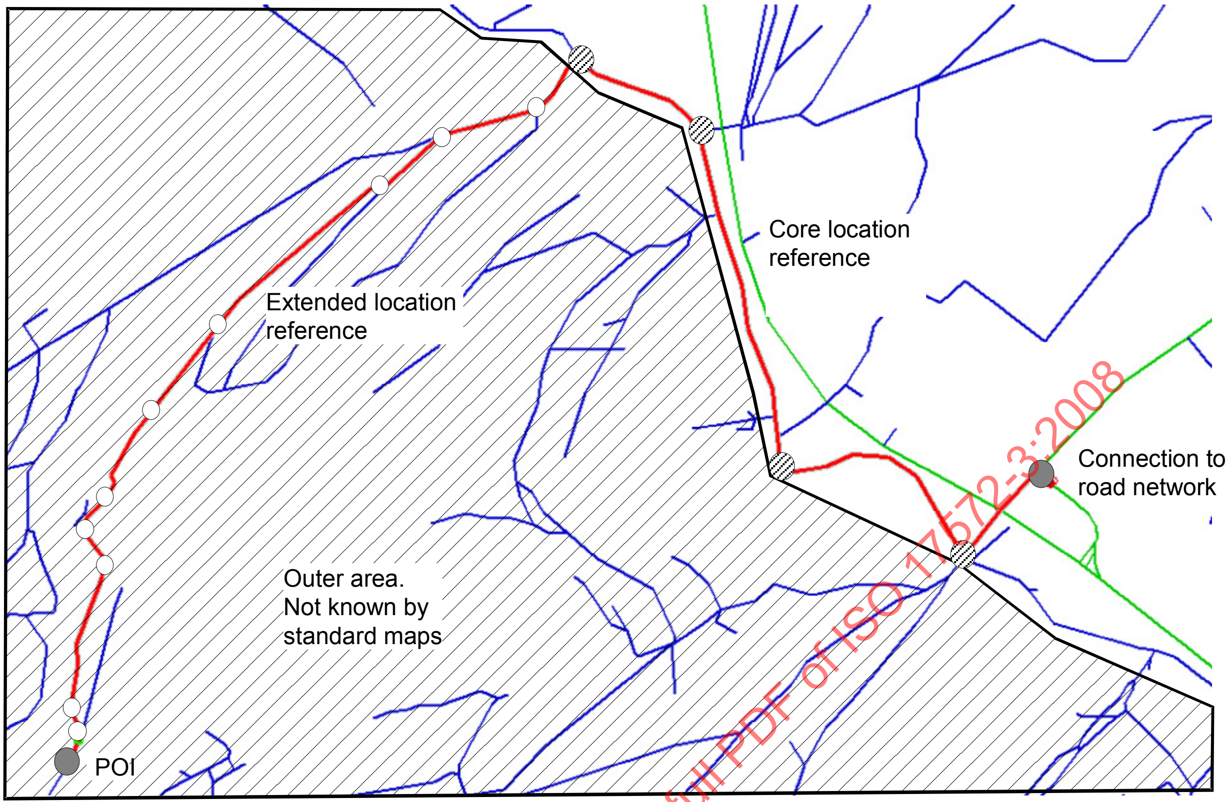

Figure 12 — Example for a destination location including extended location reference

The next sub-clauses provide rules specifying when a location reference extension is necessary and further rules specifying the selection of points for the *location reference extension*.

8.4.2 Location reference extension necessity rules

RULE-28 If either the location reference core does not include at least one road section with functional road class of at least (parameter) FC_{min}, or, in case of a destination location, the road segment of the intended destination is not at least functional road class of (parameter) FC_{min} a location reference extension shall be added for that part having a functional class less important than FC_{min}. In case that the database lacks the attribute FC, Table B.1 should be used to specify the value of FC_{min}.

8.4.3 Location reference extension point selection rules

A location reference extension provides a number of topologically connected points expressing a *precise geometry description*. Put differently it lists a number of (extension) points, which lie on a simple path in the most detailed road network graph of the encoder's database starting at the first point listed and ending at the last point listed. This path is also called the location reference extension. Figure 13 shows the different possible positions for connecting the *location reference extension* to the location reference core.

The *location reference extension* is understood as a unique geometric pattern to recognize the position of the location in the decoders map (see case A-C). In the case that the destination flag is set at a point of the location reference extension, the part from the *location reference core* until that point of the *location reference extension* is treated as a part of the location to be described (see case E-F).

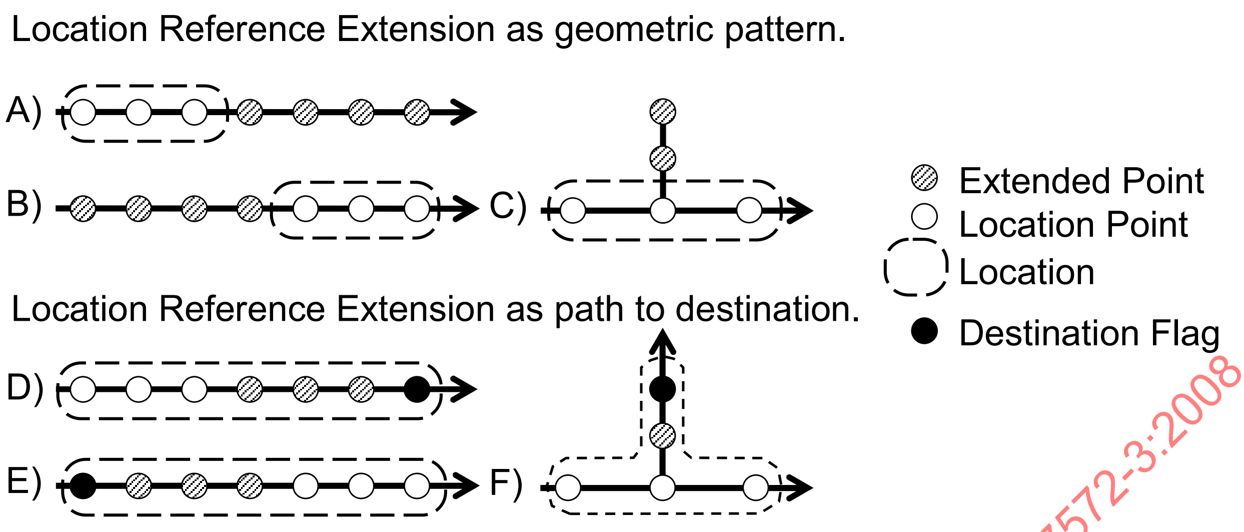

Figure 13 — Possible positions of location reference extension

NOTE Even though it is possible to code case F) it is not expected to be used because it would cause not to be linear anymore.

RULE-29 The original path of road segments captured in the *location reference extension* shall be coded in front of (respectively after) the *location reference core* or start at a location point captured in the location reference core (the connection point index (attribute CPI) is coded then, see Figure 13, case C and F), and provide a single topologically connected path including a road segment with functional road class of at least (parameter) FC_{min}. In case the default (parameter) FC_{min} can not be used, an extension specific value can be stored as attribute to the location reference extension. In case that the database lacks of the attribute FC, Table B.1 should be used to specify the value of FC_{min}. In case that the given destination to be coded is located at one point of the location reference (attribute DSF) shall be added to this point.

NOTE The attribute CPI signals the connection point in a location reference core that a location reference extension connects to. The connection point is identified using the index of the point in the location reference core.

RULE-30 In case that more than one possible path exists that satisfies RULE-29, the path corresponding to the recommended driving route to the location shall be chosen. In case that more than one recommended driving route exists, more than one extended geometry can be stored to the location reference.

RULE-31 Any part of the actual segment of the original path in between successive extension points shall have a perpendicular distance D_{perp} to the straight line connection of the successive extension points of the simple path in the most detailed road network graph of at most $D_{perp-max}$. The value of $D_{perp-max}$ along the path shall be provided with the first extension point of that path and shall be valid throughout that path. The value shall be stored in between when the D_{perp_max} value is changed due to the change of the functional class of the given path.

8.4.4 Location reference extension encoding parameters

The dynamic profile encoding rules for the location reference extension as specified in previous subclauses contain two parameters. Both are provided with default values to be taken if possible. For efficiency reasons the encoder may decide to take other values than defined here. In case of changing the value FC_{min} to a lower classified street the path to this street is getting shorter, but the probability of correct decoding will decrease. By using different intervals for correctness of the road shape, specifically making it less precise, the reconstruction of that path may be not possible for the decoder. In both cases a decoding success rate of 95 % can not be guaranteed any more.

Table 4 — Parameter value specification for the location reference extension

Parameter	Specification value	Used in rule
FC_{min}	Second road class	RULE-28, RULE-29
$D_{perp\text{-}max}$	16 m if FC has the value “main road” 8 m if FC has the value “first or second class road” 4 m if FC has a value less important than “second class road”	RULE-10, RULE-31

8.5 Coding of point locations

The rules described in previous sections apply all to point locations. However, some of the rules would require two points coded with the same coordinates to store the requested attributes. This section describes the collection of exceptional coding in case that one coordinate sufficiently supports all rules specified.

RULE-32 First, it shall be assured that the rules of 8.3.3, 8.3.4, 8.3.5 and 8.4.2 without the rules of routing distance, do not require additional coordinate pairs. In other words RULE-01, …, RULE-14, RULE-16 and RULE-28 have to be supported (RULE-17and RULE-18 are excluded and instead of RULE-15 requiring two routing points only one routing point is sufficient). A point location not fulfilling aforementioned rules with one coordinate pair shall have more points, even if the location is a point location.

RULE-33 In case that one coordinate pair sufficiently serves all rules stated above, the singular point shall be of type location point, intersection point and routing point in one, and attributes defined by 8.3.6 and 8.3.7 shall be stored to this point.

8.6 Coding of area locations

In case of area locations two different concepts described in ISO 17572-1 are specified. Both make use of the rules specified in 8.2, 8.3, 8.4 and 0.

8.6.1 Coding of explicit area

8.6.1.1 General

RULE-34 If the area location is coded as a geometric figure described by one of the following formulas, the given point of origin O shall be coded either as a point location according to 8.5, which is used to match the geometric figure to the decoders map, or as a coordinate pair if accuracy requirements are not that high.

8.6.1.2 Coding of a simple geometric area as rectangle

RULE-35 The area is defined as a rectangle by adding the south-western coordinates (V_2) and the north-eastern coordinates (V_1) as difference vectors from a point O, where O is defined by a point location or an absolute coordinate pair, to the given location reference. In optimal case O builds the south western coordinate of the rectangle, so that only one offset vector (V_1) needs to be added to O.

RULE-36 The angle α_1, counted clockwise between geographic northern direction and upright direction of the rectangle, can optionally be added to the rectangle; the vector(s) is/are build(ing) up the dimensions of the rectangle regardless of angle α_1.

NOTE Figure 14 shows the measurement of all values.

8.6.1.3 Coding of a simple geometric area as circle or ellipse

NOTE The Geometric coding with Circle or Ellipse is not recommended, because it requires a lot more resources by decoding the area designated network with not being very different from a rectangle.

RULE-37 In case that a circle is required, at least the radius V_1 shall be added in metres to the location point.

RULE-38 In case of an ellipse the semi-major axis V_1, the angle α_1 between geographic north counted clockwise and the semi-minor axis V_2 shall be added to the location point.

NOTE Figure 14 shows the measurement of all the values.

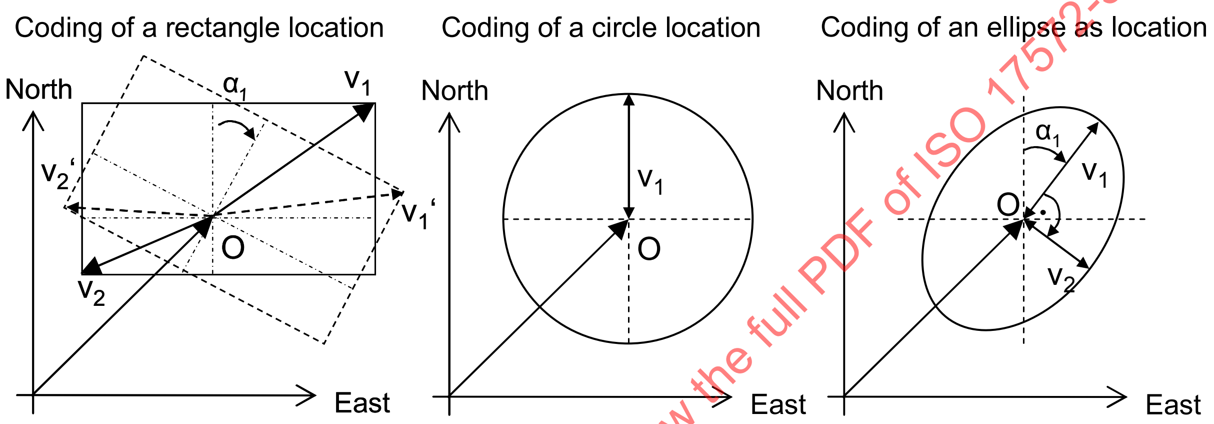

Figure 14 — Coding of simple geometric area locations

8.6.1.4 Coding of an outlined area

RULE-39 In case the former geometric figures do not fit the needs of coding a geographical area it can be shaped with its outline. The outline can be a concatenated list of segments. These segments can either be location references, matched to the street network, or polyline connections. All segments are connected at their first and last point, except that the first segments begin and the last segments end if the outline is encoded openly.

NOTE Figure 15 shows an example where two concatenated location references are closed with a polyline segment consisting of one additional coordinate in between.

RULE-40 In case that the area shall be specified as being open, the bearing of the last point is extended as a great-circle straight-line through out the part of the map in question [8].

8) A typical end of this straight-line will be the end of the given digital map or the boundary of a country.

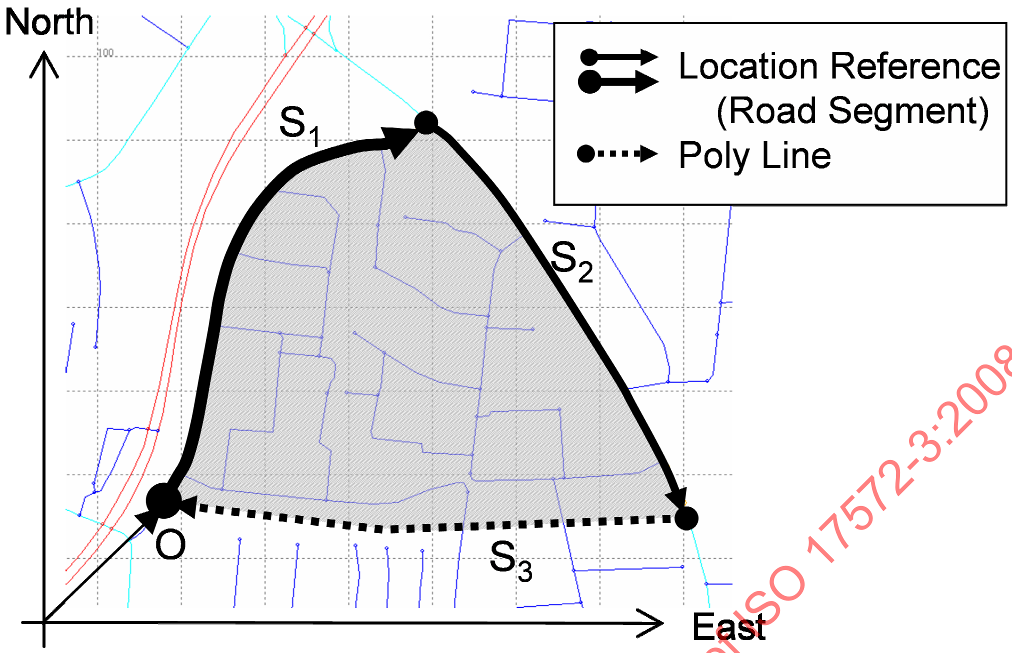

Figure 15 — Coding of an outlined area location

8.6.2 Coding of implicit area

RULE-41 Implicit Area also known as *Network Location References* (NLR) specifies a complete or partial road network. In opposition to the explicit (geometric) area reference all streets constituting a certain area shall be coded into one connected or unconnected road network. All road segments of the road network are defined as subnetwork locations in terms of linear location references. Additional connection point indexes (attribute CPI) and subnetwork indexes (attribute SNI) optionally define connectivity between the different subnetwork location references.

NOTE The attribute CPI signals the connection point in a location reference core that a subnetwork location connects to. The connection point is identified using the index of the point in the location reference core. The attribute SNI signals the subnetwork that a subnetwork location reference connects to. The subnetwork index is implicitly defined by the order of the subnetwork locations making up the implicit area and is counted from the first subnetwork location in the list, starting at value zero.

RULE-42 A network location reference can be connected to another network location reference at a connection point. The connection point is identified by the index of the connection point, which helps the decoder to close the connection between two location references without having a gap. The network location reference can be composed of three levels of network location references.

One condition should apply to simplify the network structure:

1st level-network LR: Start point of location reference may connect to another 1st level NLR, but no other than the 1st level.
The 1st level NLR is associated with road sections which are more important than the road sections associated with the 2nd and 3rd level. The decoder will start the decoding with the 1st NLRs package.

2nd level -network LR: Start point of location reference may only connect to a 1st level network location reference.
The 2nd level NLR is associated with road sections which are more important than the road sections associated with the 3rd level and less important than the road sections associated with the 1st level NLR. The decoder will make use of connection points to already decoded 1st level NLRs if provided.

3rd level -network LR: The start point of 3rd level NLR may only connect to 2nd level NLR. The 3rd level NLR is associated with road sections which are less important than the road sections associated with the 1st and 2nd level.

NOTE 1 Although the network could be expressed more flexibly by also allowing connections to same or lower level networks, this would increase the risk of mismatching. For example, if a 2nd level NLR is mismatched and additionally connected at another point to a 1st level NLR the decoder would have to decide which NLR is taken to be correct. If the 2nd level is only connected one time to the 1st level this cannot happen. The decoding quality therefore relies at maximum on location referencing and not on the connection between different NLRs.

NOTE 2 Figure 16 shows an example of coding the different levels. Which NLR belongs to the different levels is up to the encoder. Figure 17 shows an example of the Tokyo downtown area as NLR.

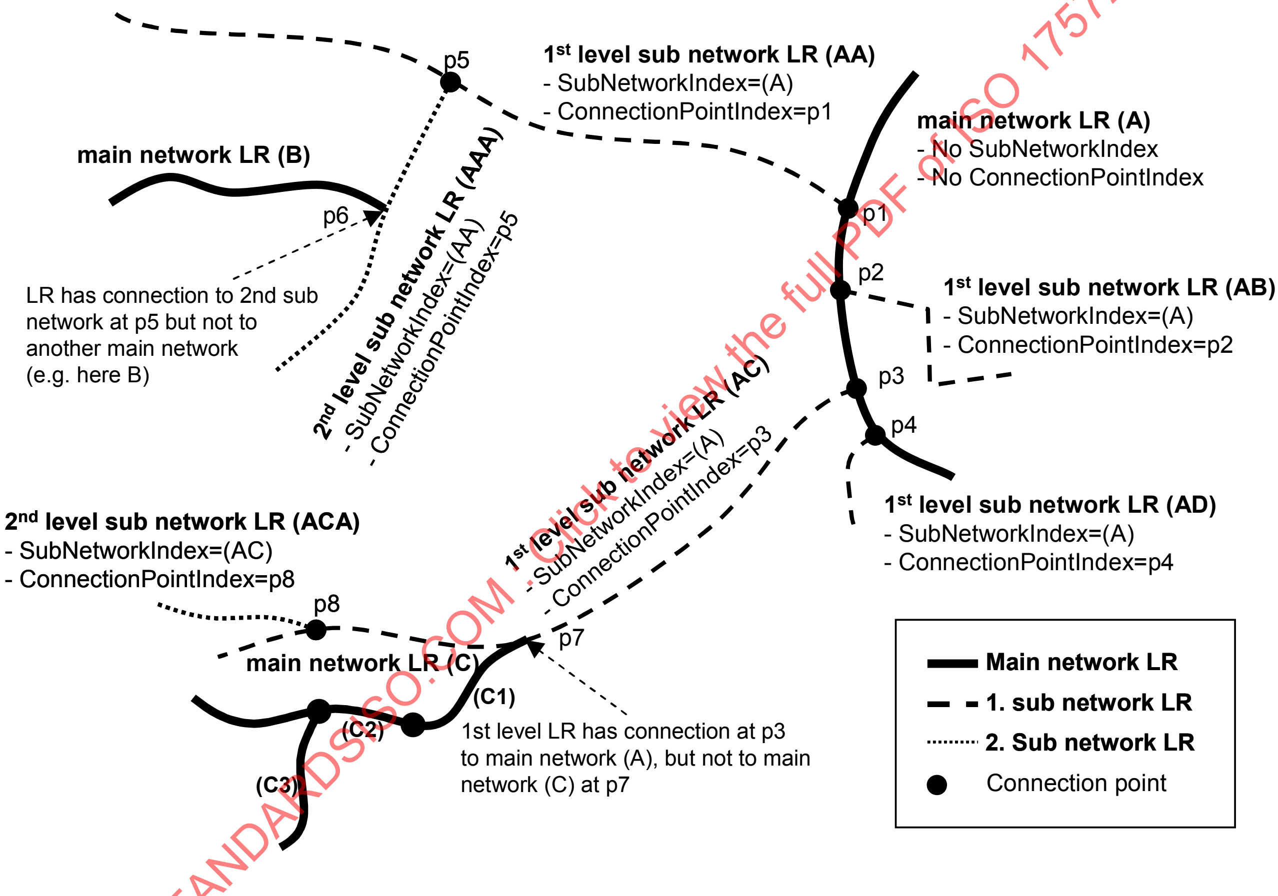

Figure 16 — Coding of different levels of network location references

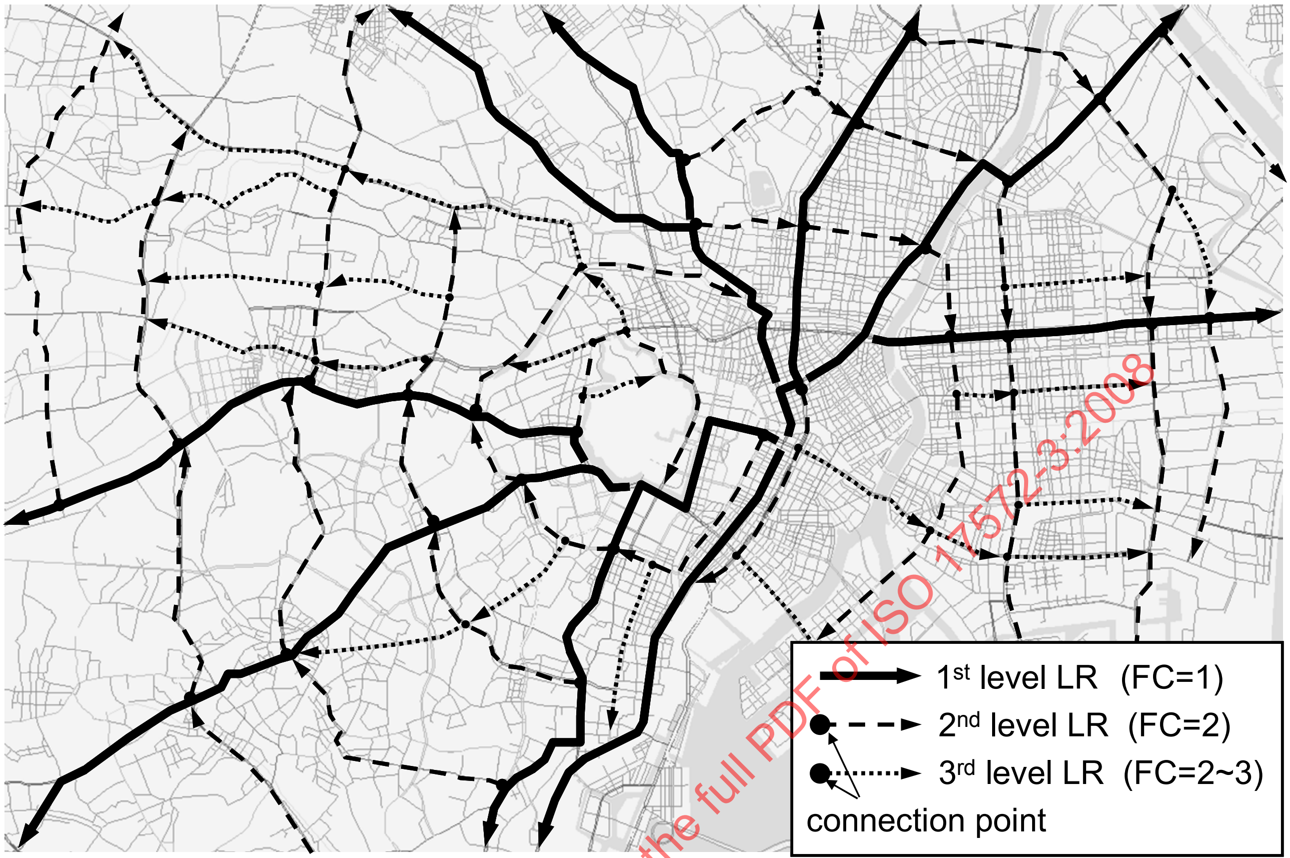

Figure 17 — Example of coding of road network in Tokyo area

9 Logical data format specification

9.1 General

This clause describes the minimum requirements in the case that a physical format is used and specified that is not defined in Annexes A to C. It gives a comprehensive overview for all data required by the different rules of Clause 8.

The data is subdivided into three different general types of data, points, attributes and there relations to each other. For the points it shall be possible to store not only the coordinates but also the corresponding order of those points regarding their topological position in relation to the real world situation.

Points and road sections carry attributes discriminating them from other road sections in the neighbourhood. Therefore it shall be possible to store attributes from different types defined here.

It is expected that a dynamic location reference has more than one point or road section. Whereby, it shall be possible to bring attributes and their corresponding parts of the location reference in relation to each other.

9.2 Data model definition

9.2.1 Introduction

This clause summarises the data structure and the minimum and maximum values used for dynamic location referencing method of the dynamic profile, while the attributes are defined in 7.2.5. The following diagrams define all elements of the dynamic location references and if they are optional or mandatory. It also specifies the data types of each simple attribute.

9.2.2 General data model

A location reference can either be an explicit area or an implicit area or a linear location reference and indicates an interpretation type information signalling the used data format description. The following main UML data model explains this fact.

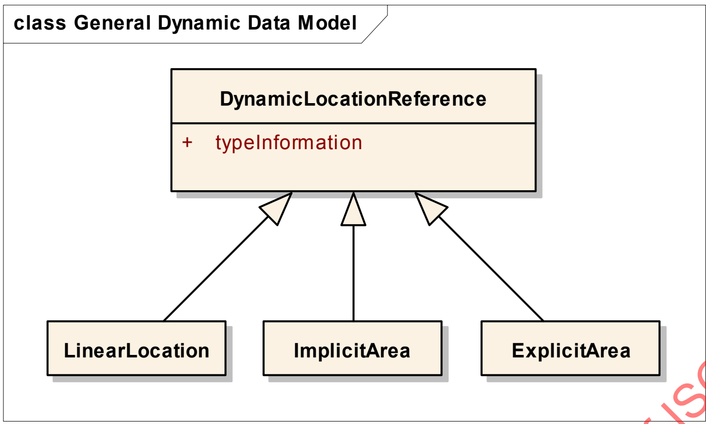

Figure 18 — UML definition of the general dynamic location reference data model

9.2.3 Linear location data model

The dynamic profile logical data model for linear location references is outlined in Figure 19, and described by the following set of rules:

1. The (linear) location reference consists of a location reference core, and optionally location reference extensions.

2. The location reference core has overall attributes, the location type and the location direction.

3. The location reference core contains an ordered list of 1...n core points.

4. A core point has one to three point types.

5. A core point has a coordinate pair and attributes, depending on its point type(s) and the local situation.

6. Each core point except the last one has a next-point relationship to one other point in the set or list.

7. Each attribute has a type, a length and a value.

8. The location reference extension has two overall attributes, the connection point (the location point in the location reference core to which the extension path connects), and the extension path encoding parameter FC_{min}.

9. The location reference extension contains an ordered list of 1...n extension points.

10. An extension point has a coordinate pair and the geometric encoding parameter distance $D_{perp\text{-}max}$ indicating the geometric approximation of the original path from the previous extension point (or connection point for the first one) to this extension point.

11. For forward compatibility with future versions, an extension point is allowed to have attributes.

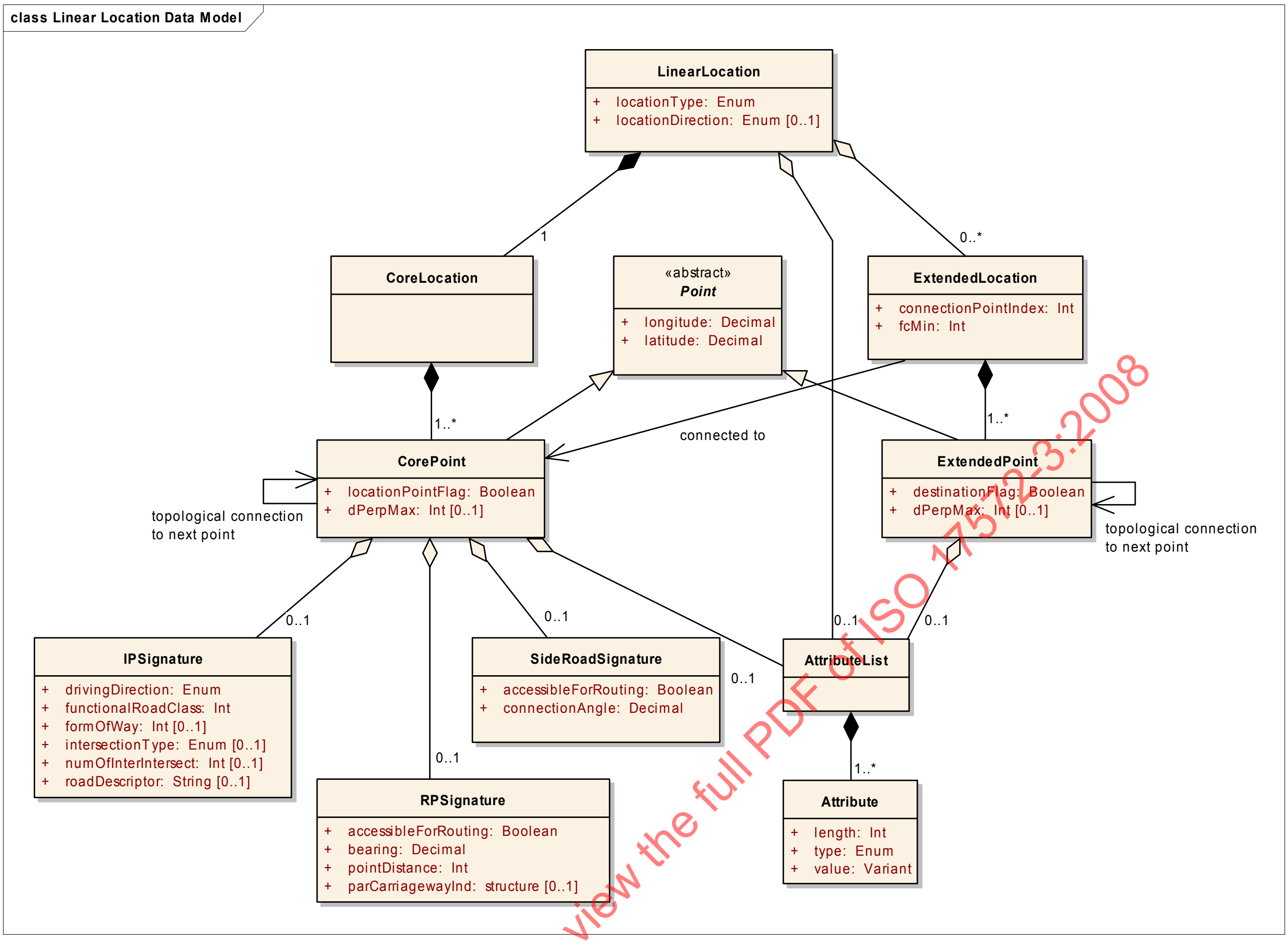

Figure 19 — UML definition of the linear location data model[9)]

9.2.4 Implicit area data model

The implicit area consists of (linear) location references which can be connected to other predecessor location references.

9) Both the RPSignature and the IPSignature comprise a set of RP and IP relevant attributes.

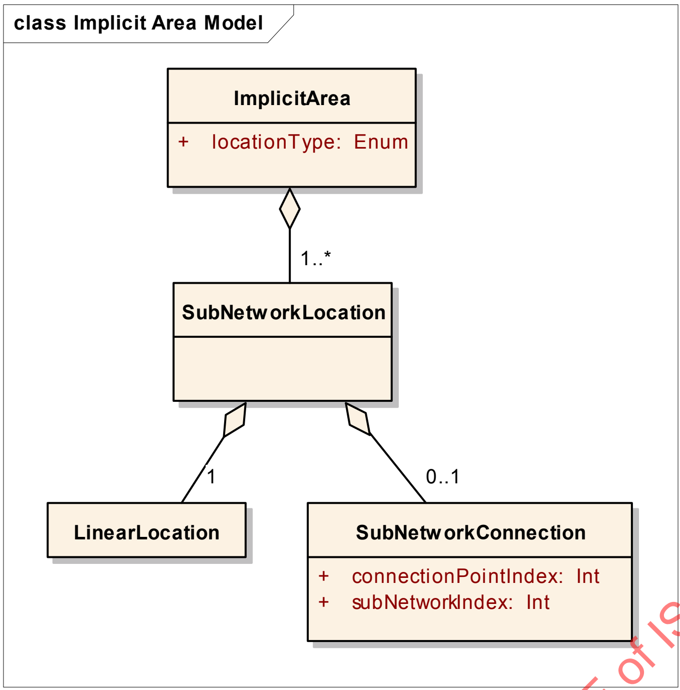

Figure 20 — UML definition of the implicit area data model

9.2.5 Explicit area data model

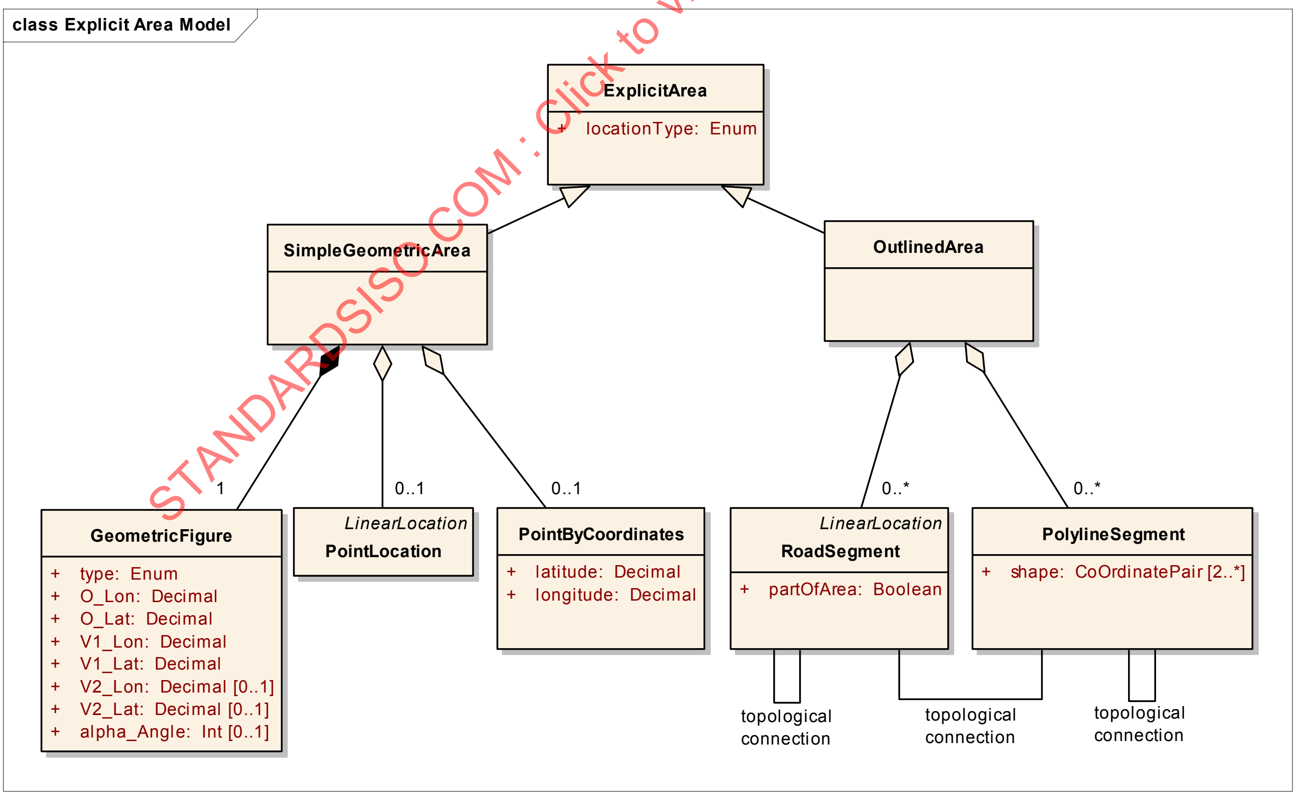

Figure 21 — UML definition of the explicit area data model

Annex A
(informative)

TPEG physical format specification for dynamic location references

A.1 Introduction

ISO 17572-1:2008, Annex E, describes the general concept of TPEG Location Reference Container. For introductory measure please refer to that section.

TPEG defines data structures in form of UML-Data Diagrams and defines rules, which allow to automatically derive the resulting formats in XSD and binary. This ensures that XML-messages on their way through the service chain can be converted to binary messages and vice versa. The following clause will define the container for dynamic location referencing named *DLR1LocationReference*.

A.2 Version

The version is stored into the message to make sure decoder can decode the messages even after doing changes in the format. The version number divides into two parts:

— *minor version* in low nibble and

— *major version* in high nibble.

All format changes being backward compatible shall be reflected in incrementing the minor version. All format changes breaking the backward compatibility shall be reflected in incrementing the major version.

This physical format is called Version 3.0 (0x30).

A.3 Data model of the dynamic location reference container

A.3.1 Introduction

The following sections define the data structure common for both formats of TPEG. For use and definition of UML-Elements used in the diagram refer to UML-Transitions defined by the TPEG Forum it is more or les self explanatory from natural UML definition. The resulting binary and xml formats are derived from these diagrams to ensure that xml messages can easily be transcoded into binary and backwards.

A.3.2 Dynamic location reference

A.3.2.1 General

The data stored in a message for the method of AGORA-C defined in Clause 8 is specified here. The following figure depicts the data structure at first level of detail, which let choose a linear location coding (including point locations), implicit area coding or explicit area coding.

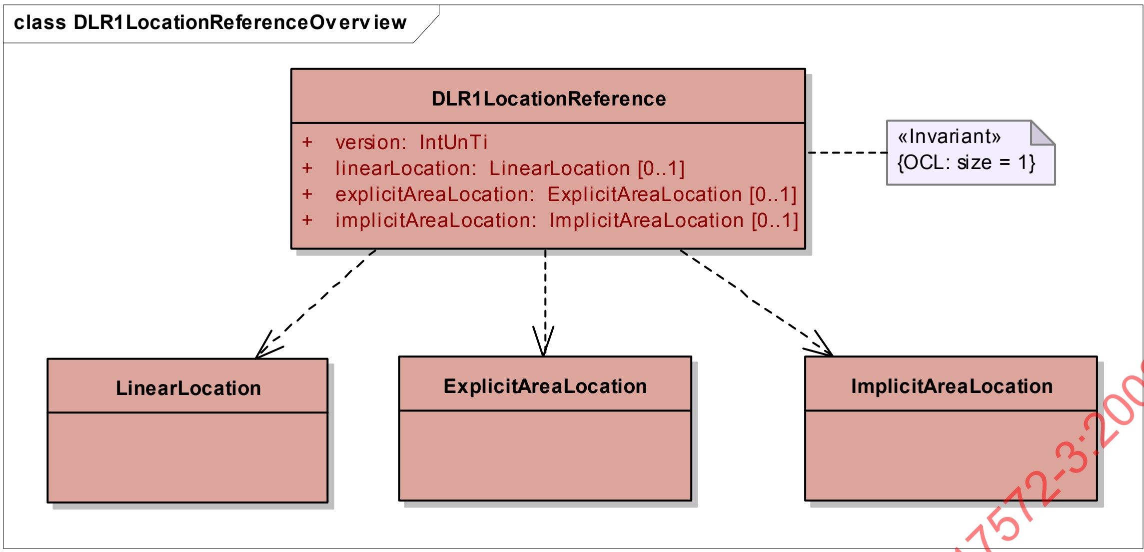

Figure A.1 — Dynamic location reference container (AGORA-C)

A.3.2.2 Agora linear- and point-location reference

The LinearLocationReference stores a point- or a linear-location as a list of points. The method to create the location reference is specified in Clause 8.

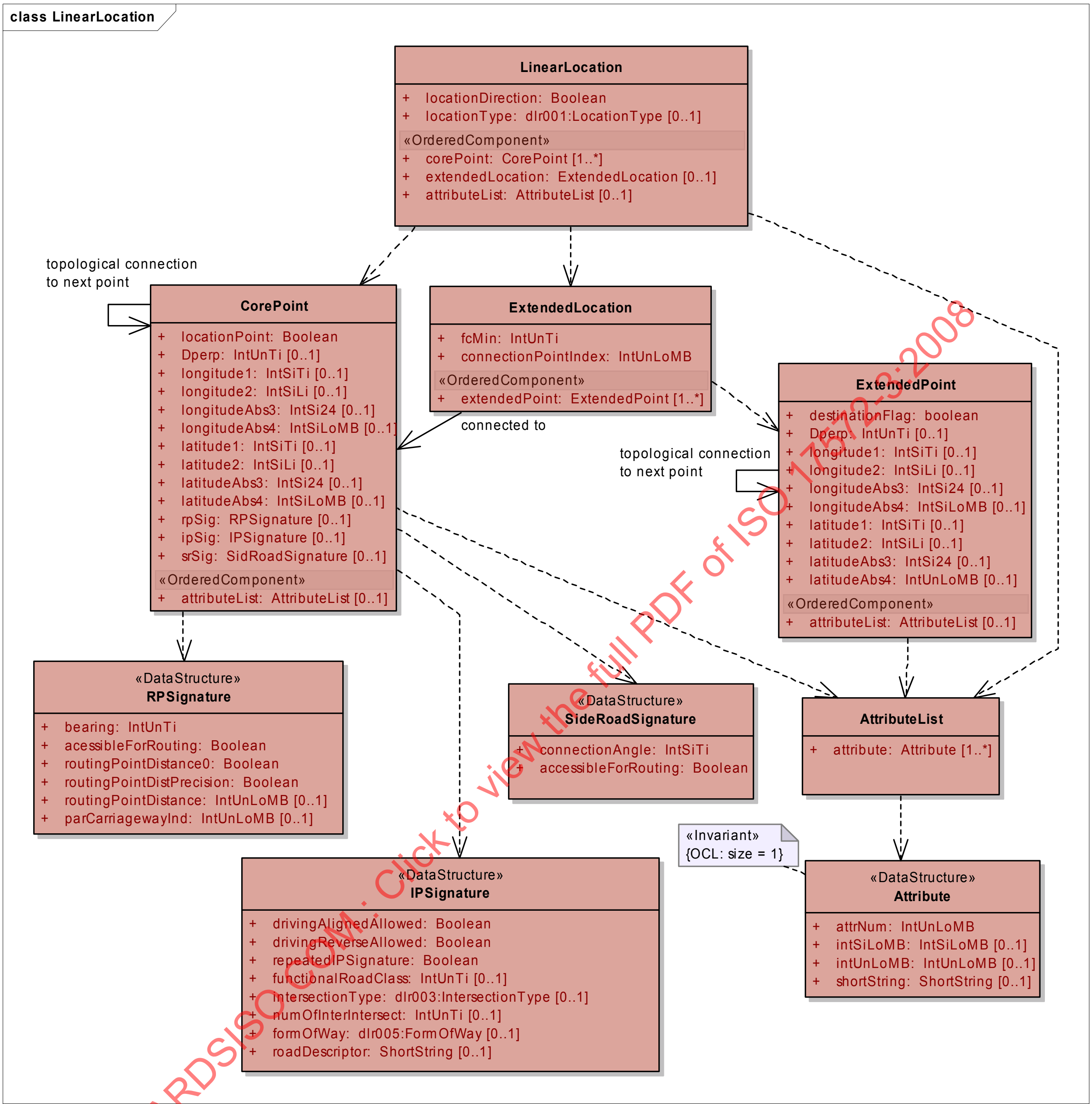

Figure A.2 — Linear dynamic location reference (AGORA-C)

A.3.2.3 Explicit area location reference container

The explicit area encapsulates different possibilities to code geometrically areas as defined in 8.6.1. The expAreaType depicts one of the possible types of an area. In case of an outlined area by default the segments span an open polygon. The attribute isClosed is set to true in case that the chain of segments shall be closed by connecting the last segment's end with the first segment's begin. In case of an open chain both ends are linear expanded with the bearing at that point up to a useful end of the map. In order of the coding direction the left side is the area to be considered. For example, a clockwise drawn circle expresses the whole area outside of this circle. A segment is one absolute coordinate followed by multiple relative coordinates.

Table A.1 — Table ExpAreaType

Id	Type of Area	Comment
001	geometric rectangle	see [8.6.1.2]
002	geometric circle	see [8.6.1.3]
003	geometric ellipse	see [8.6.1.3]
004	outlined area	see [8.6.1.4]

The data structure of the explicit area allows coding of all types defined in Table A.1. The different data elements in the ExplicitAreaLocation are used in the following ways:

1. geometric rectangle

   The first ExpAreaSegment shall be a point location designating the origin O.
   The second ExpAreaSegment shall be a point location designating the north-eastern coordinate V_1.
   The third ExpAreaSegment shall be a point location designating the south-western coordinate V_2. If this element is not given, V_2 is expected to be 0 and the origin is the south-western coordinate.
   The rotationAngle may be used to rotate the rectangle in clockwise direction.

2. geometric circle

   The first ExpAreaSegment shall be a point location designating the origin O.
   The attribute radiusA is used to define the radius of the circle.

3. geometric ellipse

   The first ExpAreaSegment shall be a point location designating the origin O.
   The attribute radiusA is used to define the radius V_1 of the ellipse.
   The attribute radiusB is used to define the radius V_2 of the ellipse.
   The rotationAngle may be used to rotate the ellipse in clockwise direction.

4. outlined area

   The first ExpAreaSegment shall be beginning of the outline around the area.
   The last ExpAreaSegment shall be the end of the outline.
   The attribute isClosed specifies if the end and the beginning shall be treated as connected with a straight line.
   The rotationAngle is not in use for outlined areas.

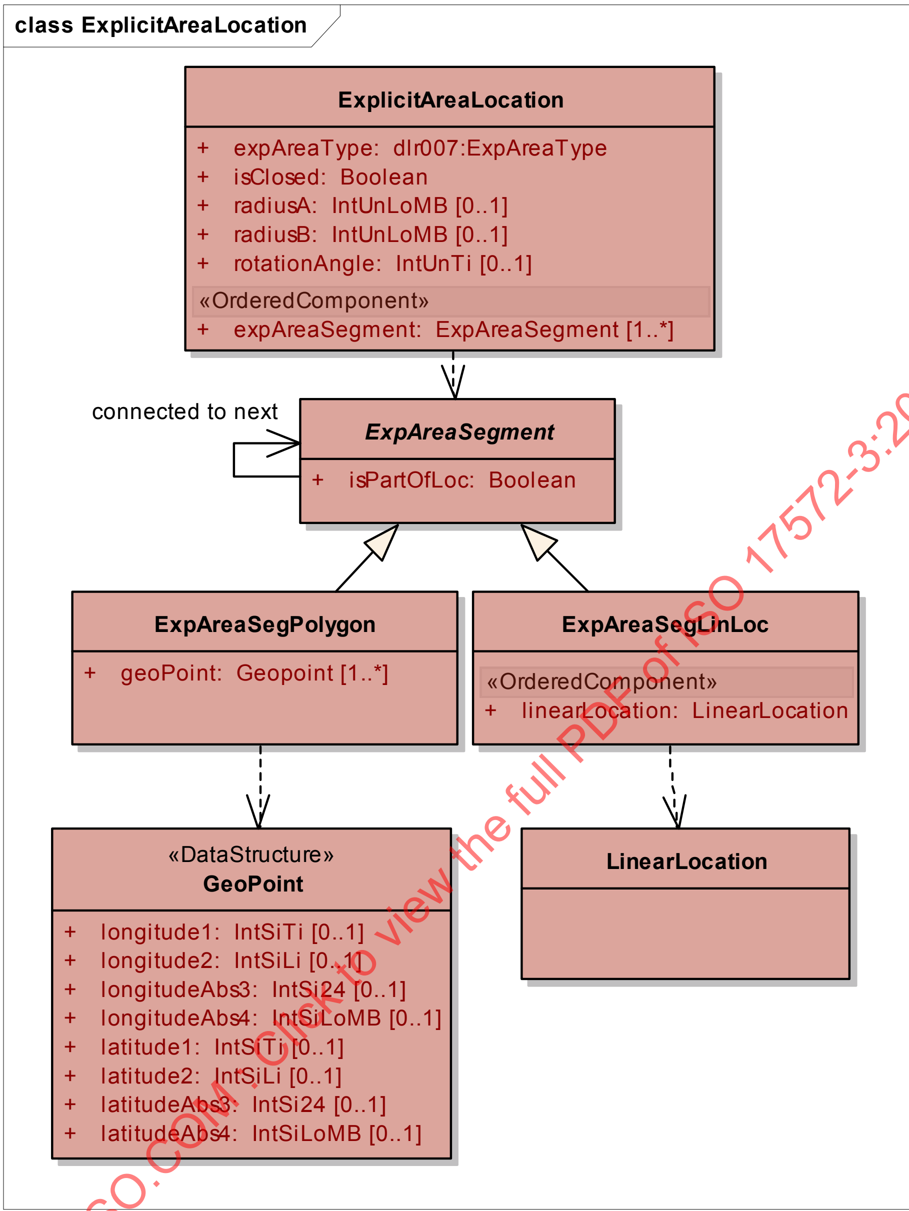

Figure A.3 — Structure of explicit area reference

A.3.2.4 Implicit area reference

This implicit area as described in 8.6.2 can be stored in the containers specified in Figure A.4.

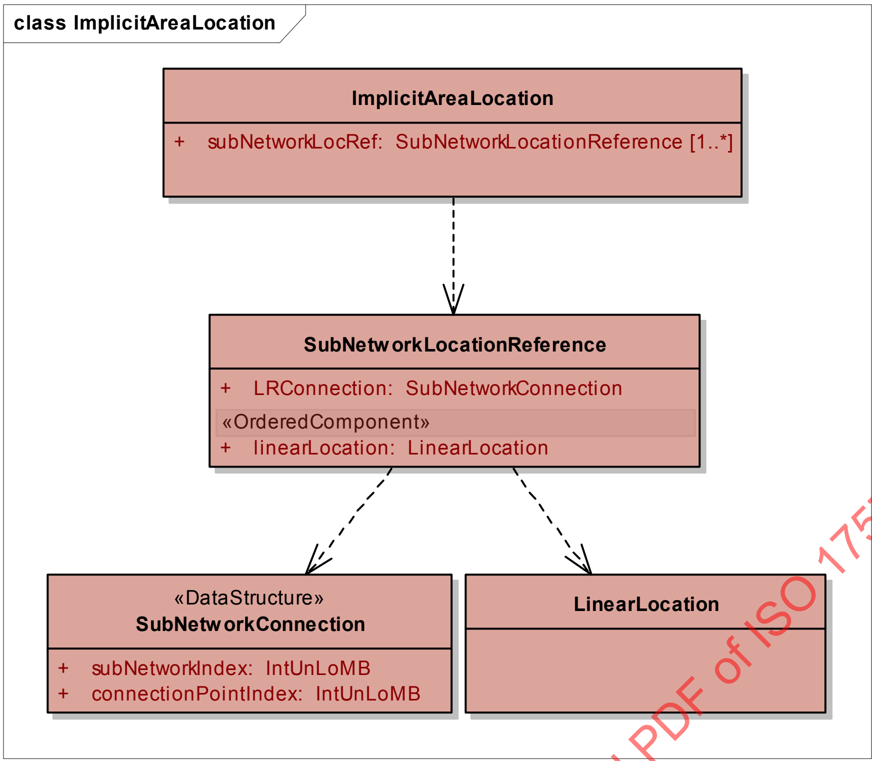

Figure A.4 — Structure of implicit area location reference

A.4 Basic field formats for encoding formats

A.4.1 General

This clause describes relevant parts for the definition of the encoding format for this part. It refers to the TPEG specifications and ISO 17572-1:2008, Annex E, in terms of the names of definitions. Specifically the naming of basic types and syntax description is intended to be changed in ISO/TS 18234-2 [8], which makes it necessary to define the attributes being used here to be internally consistent.

A.4.2 Primitive data types

For primitive data types like IntUnLo refer to ISO 17572-1:2008, Annex E.

A.4.3 Coordinate format

A.4.3.1 General

According to different needs of location referencing coordinates can be stored in two different resolutions. One standard resolution with precision accommodating needs of today's location referencing applications and one with higher precision foreseen for future applications with higher requirements to the precision of coordinates. In standard resolution the main focus was to use a precision good enough to ensure correct decoding in today's digital maps and also to save bandwidth as much as possible. The format fits in 3 byte per ordinate in opposition to that the high resolution format uses 4 bytes in variable length coding which makes a good compromise between precision and the possibility to save bytes if the distance between relative coordinates is very small. The following clauses describe how this format are specified

A.4.3.2 Absolute coordinate format

The coordinates format uses ITRS for longitude (lon) and latitude (lat) in degrees (deg). The standard accuracy for ITS application stores coordinates in order of magnitude of 10 micro degrees resolution (five decimals). The directions east and north are represented as positive values. The order is (lon, lat). In case of future use (e.g. pedestrian navigation) the resolution may be changed to a higher value. In that case the resolution has to be changed to a higher (bit-size) value. The next foreseeable useful resolution is 28 bit which is specified as high resolution in A.4.3.5. The function shifts the values by 0,5 to get round errors as small as possible.

Equation A.1 gives the transformation from decimal coordinates into integer values:

$$\mathrm{int} = \left(\mathrm{sign}\,(\mathrm{deg}) * 0{,}5 + \frac{\mathrm{deg} * 2^{\mathrm{Resolution}}}{360°} \right) \qquad \text{(A.1)}$$

Equation A.2 gives the transformation from integer values into decimal coordinates:

$$\mathrm{deg} = \left(\frac{\left(\mathrm{int} - \mathrm{sign}\,(\mathrm{int}) * 0{,}5\right) * 360°}{2^{\mathrm{Resolution}}} \right) \qquad \text{(A.2)}$$

A.4.3.3 Standard resolution 24 bit

The equations defined in A.4.3.2 are used to specify coordinates in the physical format. The standard resolution defines a quantity of 24 bit signed integer. This leads to an error of about 2,4 metres at most, which is acceptable for location referencing.

For example, 5,11233° is `0x03A2AC` and −1,98984 is `0xFE95C3`. Thus the total length of the calculated coordinate format is 6 bytes.

A.4.3.4 Relative coordinate format

In case of subsequent points it would be inefficient to store all coordinates absolute. In order to save bandwidth the relative coordinate format specification is given. With the help of corresponding flags the encoder expresses which coordinate format is in use per point. Relative coordinates are calculated as subsequent difference from one point to the next point in the linear set of points. The resolution of the relative coordinates is implicitly specified by the resolution of the preceding absolute coordinate pair.

Table A.2 presents the value ranges for each relative offset size possibility for longitude and latitude (valid for longitude at around 45° latitude).

Table A.2 — Offset ranges per number of bytes in binary physical format

Resolution	Bytes	Range longitude	Range latitude
24 bits	1 (8 bit)	319 m	606 m
28 bits	2 (16 bit)	83 km	156 km

A.4.3.5 High resolution 28 bit

The equations defined in A.4.3.2 are used to specify coordinates in the physical format. The high resolution defines a quantity of 28 bit signed integer. This leads to an error of about 0,19 m at most, which is expected to be precise enough also for pedestrian navigation. 28 bit allows storing the values in multi bytes with 4 byte per ordinate at most.

A.4.4 Attribute list value definition

Table 1 defines attributes to be stored in a location reference. Table A.3 defines for each attribute possible values and corresponding data types. When any of these attributes is used outside the standard coding layout, it may be referenced by its type number, which is represented as a multi-byte. Currently numbers specified in Table A.3 are used all other numbers are reserved for future use.

Table A.3 — Attribute list value definition

Num	Attribute type	Abbreviation	Data Type	Value	Description of values and range
01 = 0x01	location type	LT	IntUnTi	1 2 3 4 5 6	- *intersection* - *restricted access road* - *ferry* - *settlement* - *point-of-interest* - *road*
02 = 0x02	number of intermediate intersections	NIT	IntUnTi	[0-255]	
03 = 0x03	functional road class	FC	IntUnTi	0 1 2 3 4 5 6 7 8 9	- FC 0 - Main road - FC 1 - First class road - FC 2 - Second lass road - FC 3 - Third class road - FC 4 - Fourth class road - FC 5 - Fifth class road - FC 6 - Sixth class road - FC 7 - Seventh class road - FC 8 - Eight class road - FC 9 - Ninth class road
04 = 0x04	form of way	FW	IntUnTi	0 or absent 1 2 3 4 5 6 7 8 9 10 11 12	- *undefined*[10] - *motorway* - *multiple carriageway (which is not a motorway)* - *single carriageway* - *roundabout circle* - *traffic square* - *enclosed traffic area* - *slip road* - *service road* - *car park entrance or exit* - *service entrance to or exit* - *pedestrian zone* - *walkway*
05 = 0x05	road descriptor	RD	ShortString		See part 1 annex E for definition of primitive data type ShortString.
06 = 0x06	bearing	BR	IntUnTi	[0-255]	(in units of 360/256 degree),
07 = 0x07	connection angle	CA	IntSiTi	[(-128)-127]	(in units of 360/128 degree)

10) The physical format specification defines "undefined" as absent.

Table A.3 (*continued*)

Num	Attribute type	Abbreviation	Data Type	Value	Description of values and range
08 = 0x08	parallel carriageway indicator	PCI	IntUnLoMB		structure see sub sections:
		PCISF	1st LSBit Boolean	0 1	- *default search area* - *search area enhancement defined*
		PCIT	2nd Bit Boolean	0 1	- *horizontal* - *vertical*
		PCII	3.-5. Bit unsigned		[0 – 7]
		PCIN	6.-8. Bit unsigned		[0 – 7]
		PCIS	up to 3 byte of the IntUnLoMB		[1-(2^{24}-1)] (in metres)
09 = 0x09	intersection type	IT	IntUnTi	0 or absent 1 2 3 4 5 6	- *undefined*[11] - *interchange involving motorways or other limited access roads (complex)* - *roundabout* - *complex road crossing not of type "interchange involving motorways or other limited access roads" or "roundabout"* - *simple road crossing* - *traffic square* - *bivalent intersection*
10 = 0x0A	point distance	PD	IntUnLoMB	[0-(2^{32}-1)]	(in metres)
11 = 0x0B	accessible for routing flag	AFR	Boolean	0 1	- not accessible - accessible
12 = 0x0C	road descriptor of intersection	RDI	ShortString		
13 = 0x0D	driving direction	DD	2 Bit flag	0 or absent 1 2 3	- *none, undefined* - *aligned* - *reverse* - *both*
14 = 0x0E	perpendicular distance maximum ($D_{perp-max}$)	PDM	IntUnLoMB	[0-(2^{32}-1)]	(in metres)
15 = 0x0F	minimal functional road class (FC_{min})	FCM	IntUnTi	[0-9]	(values as defined for attribute FC)
16 = 0x10	destination flag	DSF	<none>		Existence at the corresponding point indicates this point as being a destination point. The attribute has no additional value to its type.
17 = 0x11	distance measure BR ($D_{m-bearing}$)	DMB	IntUnLoMB	[0-(2^{32}-1)]	Optional value in case that not parameter $D_{m-bearing}$ was used. (in metres)

11) The physical format specification defines “undefined” as absent.

Table A.3 (*continued*)

Num	Attribute type	Abbreviation	Data Type	Value	Description of values and range
18 = 0x12	distance measure CA ($D_{m\text{-}co\text{-}angle}$)	DCA	IntUnLoMB	$[0\text{-}(2^{32}\text{-}1)]$	Optional value in case that not parameter $D_{m\text{-}co\text{-}angle}$ was used. (in metres)
19 = 0x13	location point flag	LPF	<none>		Flagging that point is a location point.
20 = 0x14	location direction	LD	Boolean	0 1	- *aligned*, - *both*
21 = 0x15	connection point index	CPI	IntUnLoMB	$[0\text{-}(2^{32}\text{-}1)]$	Index of connection point
22 = 0x16	subnetwork index	SNI	IntUnLoMB	$[0\text{-}(2^{32}\text{-}1)]$	Index of subnetwork

A.5 Binary format description

A.5.1 General

For definition of the model of components or data structures refer to ISO 17572-1:2008, Annex E.

A.5.2 Binary dynamic location reference, DLR

A.5.2.1 Generic component ids

Generic component	**id**
DLR1LocationReference	defined by LRC
LinearLocation	0
ExplicitAreaLocation	1
ImplicitAreaLocation	2
AttributeList	3
CorePoint	4
ExtendedLocation	5
ExtendedPoint	6
ExpAreaSegment	defined by derived component
ExpAreaSegPolygon	7
ExpAreaSegLinLoc	8
SubNetworkLocationReference	9

A.5.2.2 DLR1LocationReference component

The DLR1LocationReference is the main container. It collects data according to the rules defined in this standard. The decoder, by knowledge of this standard can interpret the data elements in such, that it decodes the intended location into its digital map regardless of pre-coded location information.

<DLR1LocationReference(x)<Component(x)>>:=	:	
<IntUnTi>(x),	:	Identifier defined by LRC
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<IntUnTi>(version),	:	High nibble is for major changes breaking backward compatibility low nibble is for minor changes
<LinearLocation>,	:	
<ExplicitAreaLocation>,	:	
<ImplicitAreaLocation>;	:	

A.5.2.2.1 LinearLocation component

The Agora Location container is for the dynamic location references of AGORA-C. It includes all necessary information used for a receiver to decode the location reference.

<LinearLocation<Component(0)>>:=	:	
<IntUnTi>(0),	:	Identifier = 0
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<Boolean>(locationDirection),	:	If set, the locationDirection has value *both*, else, locationDirection has value *aligned*.
if (bit 1 of selector is set)	:	
<dlr001:LocationType>(locationType),	:	
n * <CorePoint>[1..*],	:	n > 0
n * <ExtendedLocation>[0..1],	:	optional
n * <AttributeList>[0..1];	:	optional

A.5.2.2.1.1 AttributeList component

The Attribute List allows adding any additional attribute to a point, Core- or Extended-location or the whole location reference.

<AttributeList<Component(3)>>:=	:	
<IntUnTi>(3),	:	Identifier = 3
<IntUnLoMB>(lengthComp);	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes, always 0 since this component has no attributes
n*<Attribute>[1..*];	:	n > 0

A.5.2.2.1.2 CorePoint component

The CorePoint is part of the location reference core it is defined by the rules in 8.3 which elements have to be filled.

<CorePoint<Component(4)>>:=	:
<IntUnTi>(4),	: Identifier = 4
<IntUnLoMB>(lengthComp),	: Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	: Length of attributes of this component in bytes
<BitArray>(selector),	:
if (bit 0 of selector is set)	:
<Boolean>(locationPoint),	: If set, core point is a location point (attribute LPF)
if (bit 1 of selector is set)	:
<IntUnTi>(Dperp),	: If set, the precise geometry description is defined
if (bit 2 of selector is set)	:
<IntSiTi>(longitude1),	: Relative coordinate 1 byte
if (bit 3 of selector is set)	:
<IntSiLi>(longitude2),	: Relative coordinate 2 bytes
if (bit 4 of selector is set)	
<IntSi24>(longitudeAbs3)	: Absolute Coordinate calculated with standard resolution (see A.4.3.3)
if (bit 5 of selector is set)	
<IntSiLoMB>(longitudeAbs4)	: Absolute Coordinate with high resolution in multibyte coding
if (bit 6 of selector is set)	:
<IntSiTi>(latitude1),	: Relative coordinate 1 byte (see A.4.3.4)
if (bit 7 of selector is set)	:
<IntSiLi>(latitude2),	: Relative coordinate 2 bytes (see A.4.3.4)
if (bit 8 of selector is set)	:
<IntSi24>(latitudeAbs3),	: Absolute Coordinate calculated with standard resolution (see A.4.3.3)
if (bit 9 of selector is set)	:
<IntSiLoMB>(latitudeAbs4),	: Absolute Coordinate with high resolution in multibyte coding (see A.4.3.5)
if (bit 10 of selector is set)	:
<RPSignature>,	:
if (bit 11 of selector is set)	:
<IPSignature>,	:
if (bit 12 of selector is set)	:
<SideRoadSignature>,	:
n * <AttributeList>[0..1];	: optional

A.5.2.2.1.3 ExtendedLocation component

Extended part of the reference, showing a street a way from the location to a part of the major road network, this is used in case that it is expected that the receiver may not know the last, smallest street of the location. It is an application decision to use the extended location or not.

<ExtendedLocation<Component(5)>>:=	:	
<IntUnTi>(5),	:	Identifier = 5
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<IntUnTi>(fcMin),	:	
<IntUnLoMB>(connectionPointIndex),	:	
n*<ExtendedPoint>[1..*];	:	n > 0

A.5.2.2.1.4 ExtendedPoint component

The extended point belongs to a list of coordinate points heading towards the destination in such, that as long as road network is available and afterwards the resulting shape can be used as solution.

<ExtendedPoint<Component(6)>>:=	:	
<IntUnTi>(6),	:	Identifier = 6
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<Boolean>(destinationFlag),	:	If set, extended point is a destination point (attribute DSF)
If (bit 1 of selector is set)		
<IntUnTi>(Dperp),	:	If set, the precise geometry description is defined
if (bit 2 of selector is set)	:	
<IntSiTi>(longitude1),	:	Relative coordinate 1 byte
if (bit 3 of selector is set)	:	
<IntSiLi>(longitude2),	:	Relative coordinate 2 bytes
if (bit 4 of selector is set)		
<IntSi24>(longitudeAbs3)	:	Absolute Coordinate calculated with standard resolution (see A.4.3.3)
if (bit 5 of selector is set)		
<IntSiLoMB>(longitudeAbs4)	:	Absolute Coordinate with high resolution in multibyte coding (see A.4.3.5)
if (bit 6 of selector is set)	:	
<IntSiTi>(latitude1),	:	Relative coordinate 1 byte (see A.4.3.4)
if (bit 7 of selector is set)	:	
<IntSiLi>(latitude2),	:	Relative coordinate 2 bytes (see A.4.3.4)
if (bit 8 of selector is set)	:	
<IntSi24>(latitudeAbs3),	:	Absolute Coordinate calculated with standard resolution (see A.4.3.3)
if (bit 9 of selector is set)	:	
<IntSiLoMB>(latitudeAbs4),	:	Absolute Coordinate with high resolution in multibyte coding (see A.4.3.5)
n * <AttributeList>[0..1];	:	optional

A.5.2.2.2 ExplicitAreaLocation component

The explicit area location encapsulates coding of a geographical area with the help of coordinates and/or dynamic location references. The segments span an area of an open polygon. The attribute isClosed is set to true in case that the chain of segments shall be closed by connecting the last segment end with the first segment begin. In case of an open chain both ends are linear expanded with the bearing at that point up to a useful end of the map. In coding order used as direction the left side is the area to be considered. For example, a clockwise drawn circle expresses the whole area outside of this circle. It is the issuer’s responsibility to ensure that useful areas are described only.

<ExplicitAreaLocation<Component(1)>>:=	:
<IntUnTi>(1),	: Identifier = 1
<IntUnLoMB>(lengthComp),	: Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	: Length of attributes of this component in bytes
<dlr007:ExpAreaType>(expAreaType),	: Specifies the interpretation rule for the geometric area: see Table A.1 for details.
<BitArray>(selector),	:
if (bit 0 of selector is set)	:
<Boolean>(isClosed),	: Defines if the geometric area shall be treated as closed, The different types specify if this attribute is in use or not. The attribute is valid only for polygon area type.
if (bit 1 of selector is set)	:
<IntUnLoMB>(radiusA),	: Defines the radius in units of 100 metre directed to north if rotationAngle is 0°.
if (bit 2 of selector is set)	:
<IntUnLoMB>(radiusB),	: Defines the radius in units of 100 metre directed to east if rotationAngle is 0°. In case of circle the attribute shall be suppressed.
if (bit 3 of selector is set)	:
<IntUnTi>(rotationAngle);	: Defines the rotation of the geometric figure after its construction in 360/256 degree
n * <ExpAreaSegment>[1..*]	: n > 0, ordered by next point relation ship.

A.5.2.2.2.1 ExpAreaSegment component

The explicit area segment allows distinction between a part of the outline being a linear location or a geographic polygon. The isPartOf flag by that allows distinguishing if the Segment shall be understood as being part of the area or not.

<ExpAreaSegment(x)<Component(X)>>:=	:
<IntUnTi>(x),	: Identifier defined by inheriting components
<IntUnLoMB>(lengthComp),	: Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	: Length of attributes of this component in bytes
<BitArray>(selector),	:
if (Bit 0 of selector is set)	:
<Boolean>(isPartOfLoc);	: If the selected segment shall be treated as being part of the location or not. In case of coordinates, all streets partly lying under the coordinates shall be treated to be part of the location if the attribute is true.

A.5.2.2.2.2 ExpAreaSegPolygon component

The explicit area segment allows ExpAreaSegment being a geographic polygon.

<ExpAreaSegPolygon< ExpAreaSegment(7)>>:=	:	
<IntUnTi>(7),	:	Identifier = 7
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<BitArray>(selector),	:	
if (Bit 0 of selector is set)	:	
<Boolean>(isPartOfLoc),	:	If the selected segment shall be treated as being part of the location or not. In case of coordinates, all streets partly lying under the coordinates shall be treated to be part of the location if the attribute is true.
<IntUnLoMB>(multi)	:	
multi * <GeoPoint>;	:	multi * coordinate pairs

A.5.2.2.2.3 ExpAreaSegLinLoc Component

Defines the explicit area segment as a linear location.

<ExpAreaSegLinLoc< ExpAreaSegment(8)>>:=	:	
<IntUnTi>(8),	:	Identifier = 8
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes of this component in bytes
<BitArray>(selector),	:	
if (Bit 0 of selector is set)	:	
<Boolean>(isPartOfLoc),	:	If the selected segment shall be treated as being part of the location or not. In case of coordinates, all streets partly lying under the coordinates shall be treated to be part of the location if the attribute is true.
<LinearLocation>;	:	The linear location designating the segment

A.5.2.2.3 ImplicitAreaLocation component

The implicit area collects a plurality of linear locations and optionally connects them to a network. The encoder encodes any street belonging to the referenced location. This enables the service provider to select precisely which part of the road network shall be treated and which not.

<ImplicitAreaLocation<Component(2)>>:=	:	
<IntUnTi>(2),	:	Identifier = 2
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes, always 0 since this component has no attributes
n * <SubNetworkLocationReference>[2..*];	:	n > 1

A.5.2.2.3.1 SubNetworkLocationReference component

The SubNetworkLocationReference connects different linear locations to a bigger subnetwork according to the definitions in 8.6.2.

<SubNetworkLocationReference<Component(9)>>:=	:	
<IntUnTi>(9),	:	Identifier = 9
<IntUnLoMB>(lengthComp),	:	Length of component in bytes, excluding the id and length indicator
<IntUnLoMB>(lengthAttr),	:	Length of attributes, always 0 since this component has no attributes
<LocationReferenceConnection>;	:	m = 1
<LinearLocation>,	:	n = 1

A.5.2.3 Datatypes of DLR

A.5.2.3.1 RPSignature datastructure

Collection of information elements used for routing point rules.

<RPSignature>:=		
<IntUnTi>(bearing),	:	360/256 in degree
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<Boolean>(accessibleForRouting),	:	see definition of accessible for routing flag AFR in Table A.3
if (bit 1 of selector is set)	:	
<Boolean>(routingPointDistance0),	:	RoutingPointDistance defined to = 0m (although no routingPointDistance attribute follows)
if (bit 2 of selector is set)	:	
<Boolean>(routingPointDistancePrecision),	:	routingPointDistancePrecision defines the precision of routingPointDistance: 0 -> precision of 10 metres 1 -> precision of 100 metres
if (bit 3 of selector is set)	:	
<IntUnLoMB>(routingPointDistance);	:	RoutingPointDistance in either parts of 10 or 100m
If (bit 4 of selector is set)	:	
<IntUnLoMB>(parCarriagewayInd)	:	see definition of PCI in Table A.3

A.5.2.3.2 IPSignature datastructure

Collection of information elements used for intersection point rules.

<IPSignature>:=		
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<Boolean>(drivingAlignedAllowed),	:	see definition of DD in Table A.3 bit for aligned direction.
if (bit 1 of selector is set)	:	
<Boolean>(drivingReverseAllowed),	:	see definition of DD in Table A.3 bit for reverse direction.
if (bit 2 of selector is set)	:	
<Boolean>(repeatedIPSignature),	:	If this flag is set IP has same signature as previous one. Absent values are repeated unless one of the following values is set. In this case it overrides the value of previous IP.
if (bit 3 of selector is set)	:	
<IntUnTi>(functionalRoadClass),	:	see definition of FC in Table A.3
if (bit 4 of selector is set)	:	
<dlr003:IntersectionType>(intersectionType),	:	see definition of IT in Table A.3
if (bit 5 of selector is set)	:	
<IntUnTi>(numOfInterIntersect),	:	see definition of NIT in Table A.3 Absent is defined as 0. Last point shall not define NIT.
if (bit 6 of selector is set)	:	
<dlr005:FormOfWay>(formOfWay),	:	see definition of FW in Table A.3
if (bit 7 of selector is set)	:	
<ShortString>(roadDescriptor);	:	see definition of RD in Table A.3

A.5.2.3.3 SideRoadSignature datastructure

Collection of information elements used for routing points having need to code a side road.

<SideRoadSignature>:=		
<IntSiTi>(connectionAngle),	:	360/128in degree. See definition of connection angle CA in Table A.3
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<Boolean>(accessibleForRouting);	:	see definition of accessible for routing flag AFR in Table A.3

A.5.2.3.4 Attribute datastructure

The attribute is a generic concept of name-value pairs with different types it is designed to be open in adding new types in further versions of LRM.

<Attribute>:=		
<IntUnTi>(attrNum),	:	Id-Number as defined in Table A.3.
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<IntSiLoMB>(intSiLoMB),	:	
if (bit 1 of selector is set)	:	
<IntUnLoMB>(intUnLoMB),	:	
if (bit 2 of selector is set)	:	
<ShortString>(shortString),	:	

A.5.2.3.5 GeoPoint datastructure

The GeoPoint defines a geographical position on the surface of the earth preferably at the centre line of a road.

<GeoPoint>:=	:	
<BitArray>(selector),	:	
if (bit 0 of selector is set)	:	
<IntSiTi>(longitude1),	:	Relative coordinate 1 byte
if (bit 1 of selector is set)	:	
<IntSiLi>(longitude2),	:	Relative coordinate 2 bytes
if (bit 2 of selector is set)		
<IntSi24>(longitudeAbs3)	:	Absolute Coordinate calculated with standard resolution
if (bit 3 of selector is set)		
<IntSiLoMB>(longitudeAbs4)	:	Absolute Coordinate with high resolution in multibyte coding
if (bit 4 of selector is set)	:	
<IntSiTi>(latitude1),	:	Relative coordinate 1 byte (see A.4.3.4)
if (bit 5 of selector is set)	:	
<IntSiLi>(latitude2),	:	Relative coordinate 2 bytes (see A.4.3.4)
if (bit 6 of selector is set)	:	
<IntSi24>(latitudeAbs3),	:	Absolute Coordinate calculated with standard resolution (see A.4.3.3)
if (bit 7 of selector is set)	:	
<IntSiLoMB>(latitudeAbs4);	:	Absolute Coordinate with high resolution in multibyte coding (see A.4.3.5)

A.5.2.3.6 LocationReferenceConnection datastructure

The LocationReferenceConnection defines where the SubNetworkLocationReference connects to the next higher level of SubNetworkLocationReference.

<LocationReferenceConnection>:=		
<IntUnLoMB>(subNetworkIndex),	:	Index of the higher level location reference
<IntUnLoMB>(connectionPointIndex);	:	Index of the point connecting to this SubNetworkLocationReference

A.5.2.3.7 dlr001:LocationType Table

Is defined in Table A.3 as enumeration of attribute LT.

A.5.2.3.8 dlr003:IntersectionType Table

Is defined in Table A.3 as enumeration of attribute IT.

A.5.2.3.9 dlr005:FormOfWay Table

Is defined in Table A.3 as enumeration of attribute FW.

A.5.2.3.10 dlr007:ExpAreaType Table

Is defined in Table A.1.

A.6 XML schema definition for dynamic location reference container

A.6.1 Introduction

The location references sent e.g. between traffic data provider and traffic control centre may be part of a larger database exchange done a for the internet typical xml-format. To serve this demand this clause defines a XML schema carrying the same content as the binary format.

A.6.2 Schema definition tables for DLR1LocationReference

The tables are defining unique tokens to any allowed entry. These tokens can be used by a decoder to translate it to other processing format.


```xml
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
	<xs:simpleType name="dlr001">
		<xs:annotation>
			<xs:documentation>LocationType</xs:documentation>
		</xs:annotation>
		<xs:restriction base="xs:string">
			<xs:enumeration value="dlr001_001"/>
			<xs:enumeration value="dlr001_002"/>
			<xs:enumeration value="dlr001_003"/>
			<xs:enumeration value="dlr001_004"/>
			<xs:enumeration value="dlr001_005"/>
			<xs:enumeration value="dlr001_006"/>
		</xs:restriction>
	</xs:simpleType>
	<xs:simpleType name="dlr003">
		<xs:annotation>
			<xs:documentation>IntersectionType</xs:documentation>
		</xs:annotation>
		<xs:restriction base="xs:string">
			<xs:enumeration value="dlr003_001"/>
			<xs:enumeration value="dlr003_002"/>
			<xs:enumeration value="dlr003_003"/>
			<xs:enumeration value="dlr003_004"/>
			<xs:enumeration value="dlr003_005"/>
			<xs:enumeration value="dlr003_006"/>
		</xs:restriction>
	</xs:simpleType>
	<xs:simpleType name="dlr005">
		<xs:annotation>
			<xs:documentation>FormOfWay</xs:documentation>
		</xs:annotation>
		<xs:restriction base="xs:string">
			<xs:enumeration value="dlr005_001"/>
			<xs:enumeration value="dlr005_002"/>
			<xs:enumeration value="dlr005_003"/>
			<xs:enumeration value="dlr005_004"/>
			<xs:enumeration value="dlr005_005"/>
			<xs:enumeration value="dlr005_006"/>
			<xs:enumeration value="dlr005_007"/>
			<xs:enumeration value="dlr005_008"/>
			<xs:enumeration value="dlr005_009"/>
			<xs:enumeration value="dlr005_010"/>
			<xs:enumeration value="dlr005_011"/>
			<xs:enumeration value="dlr005_012"/>
```



```
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="dlr007">
        <xs:annotation>
            <xs:documentation>ExpAreaType</xs:documentation>
        </xs:annotation>
        <xs:restriction base="xs:string">
            <xs:enumeration value="dlr007_001"/>
            <xs:enumeration value="dlr007_002"/>
            <xs:enumeration value="dlr007_003"/>
            <xs:enumeration value="dlr007_004"/>
            <xs:enumeration value="dlr007_005"/>
        </xs:restriction>
    </xs:simpleType>
</xs:schema>
```

A.6.3 Schema definition DLR1LocationReference

Note: tpegTYP is specified in ISO 17572-1.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="TPEG" xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="TPEG"
elementFormDefault="qualified">
    <xs:include schemaLocation="tpegTYP.xsd"/>
    <xs:include schemaLocation="tpegDLR-Tables.xsd"/>
    <xs:element name="DLR1LocationReference" type="DLR1LocationReference"/>
    <xs:complexType name="LinearLocation">
        <xs:sequence>
            <xs:element name="CorePoint" type="CorePoint" maxOccurs="unbounded"/>
            <xs:element name="AttributeList" type="AttributeList" minOccurs="0"/>
            <xs:element name="ExtendedLocation" type="ExtendedLocation" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="locationDirection" type="Boolean" use="required"/>
        <xs:attribute name="locationType" type="dlr001" use="optional"/>
    </xs:complexType>
    <xs:complexType name="Attribute">
        <xs:sequence minOccurs="1" maxOccurs="1">
            <xs:element name="shortString" type="ShortString" minOccurs="0"/>
            <xs:element name="intSiLoMB" type="IntSiLoMB" minOccurs="0"/>
            <xs:element name="intUnLoMB" type="IntUnLoMB" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="attrNum" type="intUnTi" use="required"/>
    </xs:complexType>
    <xs:complexType name="AttributeList">
        <xs:sequence>
            <xs:element name="Attribute" type="Attribute" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RPSignature">
        <xs:attribute name="bearing" type="IntUnTi" use="required"/>
        <xs:attribute name="accessibleForRoutingFlag" type="Boolean" use="required"/>
        <xs:attribute name="routingPointDistance0" type="Boolean" use="required"/>
        <xs:attribute name="routingPointDistPrecision" type="Boolean" use="required"/>
        <xs:attribute name="routingPointDistance" type="IntUnLoMB" use="optional"/>
        <xs:attribute name="parCarriagewayInd" type="IntUnLoMB" use="optional"/>
    </xs:complexType>
    <xs:complexType name="CorePoint">
        <xs:sequence>
            <xs:element name="RPSignature" type="RPSignature" minOccurs="0"/>
```



```xml
            <xs:element name="IPSignature" type="IPSignature" minOccurs="0"/>
            <xs:element name="SideRoadSignature" type="SideRoadSignature" minOccurs="0"/>
            <xs:element name="AttributeList" type="AttributeList" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="locationPoint" type="Boolean" use="required"/>
        <xs:attribute name="Dperp" type="IntUnTi" use="optional"/>
        <xs:attribute name="longitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="longitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="longitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="longitudeAbs4" type="IntSiLoMB" use="optional"/>
        <xs:attribute name="latitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="latitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="latitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="latitudeAbs4" type="IntSiLoMB" use="optional"/>
    </xs:complexType>
    <xs:complexType name="SideRoadSignature">
        <xs:attribute name="connectionAngle" type="IntSiTi" use="required"/>
        <xs:attribute name="accessibleForRoutingFlag" type="Boolean" use="required"/>
    </xs:complexType>
    <xs:complexType name="GeoPoint">
        <xs:attribute name="longitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="longitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="longitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="longitudeAbs4" type="IntSiLoMB" use="optional"/>
        <xs:attribute name="latitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="latitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="latitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="latitudeAbs4" type="IntSiLoMB" use="optional"/>
    </xs:complexType>
    <xs:complexType name="LocationReferenceConnection">
        <xs:attribute name="subNetworkIndex" type="IntUnLoMB" use="required"/>
        <xs:attribute name="connectionPointIndex" type="IntUnLoMB" use="required"/>
    </xs:complexType>
    <xs:complexType name="ExtendedLocation">
        <xs:sequence>
            <xs:element name="ExtendedPoint" type="ExtendedPoint" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="fcMin" type="IntUnTi" use="required"/>
        <xs:attribute name="connectionPointIndex" type="IntUnLoMB" use="required"/>
    </xs:complexType>
    <xs:complexType name="IPSignature">
        <xs:attribute name="functionalRoadClass" type="IntUnTi" use="required"/>
        <xs:attribute name="drivingAlignedAllowed" type="Boolean" use="required"/>
        <xs:attribute name="drivingReverseAllowed" type="Boolean" use="required"/>
        <xs:attribute name="intersectionType" type="dlr003" use="optional"/>
        <xs:attribute name="numOfInterIntersect" type="IntUnTi" use="optional"/>
        <xs:attribute name="formOfWay" type="dlr005" use="optional"/>
        <xs:attribute name="roadDescriptor" type="ShortString" use="optional"/>
    </xs:complexType>
    <xs:complexType name="CoreLocation">
        <xs:sequence>
            <xs:element name="CorePoint" type="CorePoint" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ExplicitAreaLocation">
        <xs:sequence>
            <xs:element name="ExpAreaSegment" type="ExpAreaSegment" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="expAreaType" type="dlr007" use="required"/>
        <xs:attribute name="isClosed" type="Boolean" use="required"/>
```

```
        <xs:attribute name="radiusA" type="IntUnLoMB" use="optional"/>
        <xs:attribute name="radiusB" type="IntUnLoMB" use="optional"/>
        <xs:attribute name="rotationAngle" type="IntUnTi" use="optional"/>
    </xs:complexType>
    <xs:complexType name="ExpAreaSegment">
        <xs:sequence>
            <xs:element name="LinearLocation" type="LinearLocation" minOccurs="0"/>
            <xs:element name="GeoPoint" type="GeoPoint" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="isPartOfLoc" type="Boolean" use="required"/>
    </xs:complexType>
    <xs:complexType name="ImplicitAreaLocation">
        <xs:sequence>
            <xs:element name="SubNetworkLocationReference" type="SubNetworkLocationReference"
minOccurs="1" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="SubNetworkLocationReference">
        <xs:sequence>
            <xs:element name="LinearLocation" type="LinearLocation"/>
            <xs:element name="LocationReferenceConnection" type="LocationReferenceConnection"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ExtendedPoint">
        <xs:sequence>
            <xs:element name="AttributeList" type="AttributeList" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="destinationPoint" type="Boolean" use="required"/>
        <xs:attribute name="Dperp" type="IntUnTi" use="optional"/>
        <xs:attribute name="longitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="longitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="longitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="longitudeAbs4" type="IntSiLoMB" use="optional"/>
        <xs:attribute name="latitude1" type="IntSiTi" use="optional"/>
        <xs:attribute name="latitude2" type="IntSiLi" use="optional"/>
        <xs:attribute name="latitudeAbs3" type="IntSi24" use="optional"/>
        <xs:attribute name="latitudeAbs4" type="IntSiLoMB" use="optional"/>
    </xs:complexType>
    <xs:complexType name="DLR1LocationReference">
        <xs:sequence>
            <xs:element name="LinearLocation" type="LinearLocation" minOccurs="0"/>
            <xs:element name="ExplicitAreaLocation" type="ExplicitAreaLocation" minOccurs="0"/>
            <xs:element name="ImplicitAreaLocation" type="ImplicitAreaLocation" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="version" type="IntUnTi" use="required"/>
    </xs:complexType>
</xs:schema>
```


Annex B
(informative)

Coding guidelines for dynamic location references

B.1 Introduction

This clause presents help to implement the method of location referencing in a seamless way. For this purpose some helping information and a coding procedure, based on the rules described in Clause 8 is provided. The procedures may be used as the basis to develop a coding algorithm. Since the number of decoding methods is manifold and dependent on the implementation of the used digital map database, no guidelines for decoding dynamic location references are given.

B.2 Use of optional attributes in the location reference

B.2.1 General

In some cases, optional attributes or points may be set to the location reference in addition to the minimal set of attributes required by the rules in Clause 8. This section explains some of the attributes put to the location reference to make it even more useful to the decoder for specific applications.

B.2.2 Additional point distance attribute

In case that the precise geometry description was applied as defined in RULE-10 the attribute PD should be added to the last point belonging to the precise geometry description segment, indicating the driving length to the next segment being a precise geometry description or the following routing point.

In case of explicitly known distance between an intersection and one of the extremities of the location attribute PD may be used to explicitly define this length.

B.2.3 Additional road descriptor attribute

In case that RULE-11 applies to an intersection having a name different from any street connected to it (e.g. the name of the roundabout), the attribute RDI should be added.

B.2.4 Additional side road information

In case of extended location reference coding on intersections RULE-24 may be considered to inform about intended turn.

B.2.5 Additional points

In case that a special need require a point (e.g. connecting different location references at a given intersection) at a position where no other rule require a point, the point can be added and labelled as location point without breaking the rules.

B.3 Interpretation of functional road class

Dynamic location referencing uses this classification for distinguishing the parts of the road network, having a certain higher probability of existence in different databases. A standard map database will deliver this attribute as defined in GDF; however, the attribute is quite differently used between countries and providers. The location referencing method considers this in the rules by leaning only on categorisations with high congruence between different map databases. The attribute FC may be not stored in some databases, but the encoder and decoder need to be able to derive it from other information available (speed, lanes, routing tables, etc.). For this purpose, the Table B.1 provides an interpretation of the most used functional road classes in the rules.

Table B.1 — Functional road class interpretation description

FUNCTIONAL ROAD CLASS	main roads	first class roads	second class roads	⩾ third class roads
INTERPRETATION DESCRIPTION	Connection of countries or metropolitan areas, mainly with separated carriageways.	National main roads connecting different centres, mainly with separate carriageways.	National or arterial roads.	other roads.

B.4 Coding procedures

B.4.1 General

First the coding procedure for the location reference core is described in B.4.2, followed by the coding procedure for the optional location reference extension. A location reference extension may be added in case the location to be referenced is a destination location, i.e. to be used as destination to determine a route from the current position of the user to the location in question.

B.4.2 Coding procedure for the linear location reference core

In defining the code by following the procedure below, a table (1) is constructed with the following fields:

— point number (from 1 onwards)
— location point (1 = yes, 0 = no)
— intersection point (1 = yes, 0 = no)
— routing point (1 = yes, 0 = no)
— coordinates
— one field for every mandatory attribute (value if present)

For additional (i.e. not mandatory) attributes a second table (2) is constructed, with the following fields:

— point number
— attribute type code
— attribute value

A third table (3) stores the attributes used for the overall information of the location reference with the following fields.

— attribute type code
— attribute value

All points that make up the reference are determined in several steps. The points constitute a logically ordered list, from the defined start point of the location to the defined end point, where each point has topologically (i.e. in the network) a direct connection (i.e. not via another point in the list) with the next point in the list. In this way the list implicitly defines the next-point-relationship. New points are added to the list in such a way as to maintain this logical next-point relationship order in the direction of the location (further to be called in logical order).

1. Define the location (stretch of the road network) to be coded and store (attribute LT) for overall information in Table 3.

2. If the defined location or part of it is separately digitised for either direction, or contains very complex intersections (which are not only passed through), either direction shall be considered a separate location, and be separately coded if both directions need to be referenced. Define its (coding) location direction (and thereby its start and end points), if possible correspondent with the direction to be referenced, if this is only one direction.

3. Determine the referenced direction(s) of the referenced location in relation to the coding direction of the location (attribute LD) for overall information in Table 3.

4. Determine if the start point of the location is at an intersection or not.

   — If the start point is at an intersection that is not a complex intersection, it defines both the first IP and the first RP.

   — If the start point is at a complex intersection it only defines the first IP, and the first RP is chosen away from the intersection.

   — If the start point is not at an intersection, it represents both the first IP and the first RP.

   — In any case the start point defines the first LP.

   — Add the points, their IP/RP attributes and their coordinates to Table 1.

5. Determine if the end point of the location is at an intersection or not.

   — If the end point is at an intersection that is not a complex intersection, it defines both the last IP and the last RP.

   — If the end point is at a complex intersection it only defines the last IP, and the last RP is chosen away from the intersection.

   — If the end point is not at an intersection, it only represents the last RP, and not an IP. In that case find the last intersection before the end point regardless of signature change happened, and create the last IP at that intersection.

   — In any case the end point defines the last LP.

   — Add the points, their IP/RP attributes and their coordinates to Table 1, in logical order.

6. Determine all intermediate intersections between the first and the last IP that mark a change of road section signature, and thereby define all attributes for the intersection point. Add these points, their IP attributes and their coordinates in the order of the location direction to the existing points in Table 1.

7. While following the stretch of the location, determine if the difference between driving length and airline between successive points marked gets longer than 10 m or 5 % of the length of the original path and mark a point at the place of largest perpendicular distance. Add these points as Location Points in logical order to the existing points in Table 1.

8. Determine if intermediate routing points are needed to calculate a route from first RP to last RP, taking into account the functional class weight factor of RULE-17, and the criteria of RULE-18. By successive iteration determine the required intermediate routing points. Where possible these should coincide with already defined intersection points. Add new points in logical order to the list, including their IP/RP attributes and their coordinates.

9. Each IP and/or RP that is located on the location to be referenced also defines an LP.

10. For each intersection point in the list determine, and add to Table 1:

    — number of intermediate intersections between this and the next intersection point (attribute NIT), and add to the table if 1 or more

    — the functional class of the stretch until the next intersection point (attribute FC)

    — the form of way of the stretch until the next intersection point (attribute FW)

    — the road descriptor of the stretch until the next intersection point (attribute RD), and add to the table if different from previous stretch, or if it concerns the first intersection point

    — the intersection type of the intersection point (attribute IT)

    — the driving direction of the stretch (attribute DD)

11. For each routing point in the list determine, and add to Table 1:

    — the road's bearing (attribute BR)

    — the parallel carriageway indicator if applicable (attribute PCI)

    — the distance to the next routing point (attribute PD) — not applicable to the last routing point

    — the accessible for routing flag (attribute AFR),

12. For each point determine if additional attributes are needed, determine these and add them with point number to Table 2. In particular, if a routing point is located on an at least trivalent junction, then determine:

    — the connection angle (attribute CA)

    — the side road's accessible for routing flag (attribute AFR)

13. For each pair of successive location points determine whether the segment fulfils all conditions specified in RULE-10. If not so, find that point having the largest perpendicular distance towards the airline and add it as intermediate location point in Table 1. If the point requires precise geometry description according to RULE-10, add the precise geometry points marked as location points in logical order in Table 1. Add the maximum perpendicular distance used to the first point of each stretch where the value changes.

14. Repeat step 14 recursively until all segments between successive location points fulfil RULE-10.

15. From the information under item 3, and in Tables 1 to 3 construct the location reference.

B.4.3 Coding procedure for the location reference extension for location referencing

The Location Reference Extension makes use of the same set of tables defined in B.4.2. The determination of the location reference for the location reference extension proceeds as follows:

1. In case of locations expected to being not stored in any map for reasons like being in a rural area, being a newly created part of the road network or being on parts already known as having differences to digital maps, define the location (i.e. destination) as being at the end of the extended location by adding the destination flag (attribute DSF) to the end point of the extended location reference.

2. Determine whether the location reference core explained in B.4.2 contains at least one road section of at least functional road class FC_{min}. If so, stop. If not, proceed to item 3.

3. For all location points which are captured in the location reference core, plan a path with least costs as defined in RULE-17 outward until reaching a road section with road class of at least (parameter) FC_{min}. Preferably take the proposed access route for the given location in use.

4. Determine the location reference core satisfying the relevant rules as specified in section 8 in a fashion equivalent as described in the coding procedure in section B.4.2 either for the location in question if step 1 didn't apply or for the entry point determined by step 3.

5. For the path of the location reference extension, remove any intermediate shape point, as long as the perpendicular distance of the straight-line approximation between the remaining points to the original path is less than the distance (parameter) $D_{perp\text{-}max}$.

6. Store the location reference extension with the start location point (as connection point), the encoding parameter FC_{min}, and the remaining shape points as extension points in direction of the location and the attributes PDM used to generalise the segments.

B.4.4 Coding procedure for encoding implicit areas

The implicit area is a package of linear locations more or less connected to each other. It is application dependent if connection of location references is required or not. However, the connection between location references allows the decoder to close possible small gaps between successive location references. For this reason this procedure describes the connected subnetwork.

1. Determine all road sections being in question and belonging to the major road network (having the lowest value of attribute FC) and build a network of sections.

2. Determine the location reference core satisfying the relevant rules as specified in Clause 8 in a fashion equivalent as described in the coding procedure in B.4.2. Make sure that any intersection with a side road being also part of the implicit area is coded with at least a location point (LP) in the right location reference.

3. Store all location references as first level network location references.

4. For each side road also being in question start with the corresponding point of the location references (already stored in step 3) as connection point and follow the paths until the secondary road network is left.

5. Again determine the location reference core satisfying the relevant rules as specified in section 8 in a fashion equivalent as described in the coding procedure in B.4.2. Make sure that any intersection with a side road being also part of the implicit area is coded with at least a location point (LP) in the right location reference. And store the location references as second level network location references.

6. In case that also less classified parts of the road network shall be connected to the implicit area. Repeat steps 4 and 5 by considering if location reference extensions are applicable especially for this parts of the road network.

7. After coding all road sections as location references, apply some optimization to reduce redundancies in the data records stored to reduce the size of the implicit area location reference.

B.4.5 Coding procedure for encoding of explicit (outlined) area

In case the service requires a explicit area to be coded a choice have to be made if a simple geometric area serves the needs or if the area shall be described by its outline. The following procedure assumes the need of an outlined area because this seems to be the more sophisticated procedure.

1. From the given outline determine parts having the need to be coded as exact location references according to a normal linear location and code for each of this segments the by determining the location reference core satisfying the relevant rules as specified in Clause 8 in a fashion equivalent as described in the coding procedure in B.4.2. Define for each location reference if the given part of the road network shall be considered as being a part of the area or not.

2. Determine polylines for each intermediate part to close the outline of the explicit area. In case that the area shall be specified as being open, the bearing of the last point is extended as a great-circle straight-line through out the part of the map in question 12).

3. Make sure that each of the segments (either location reference segment or shape segment) is connected to its predecessor in direction of the outline. The area in question is defined as being on the right hand side of the outline, which means a clockwise connected outline determines the inner part of a closed outline and a counter clockwise connected outline determines the outer part as area in question.

4. The start of the first segment is being called the origin (O) of the outline which may be used as coordinate for relative coordinate specification.

12) A typical end of this straight-line will be the end of the given digital map or the boundary of a country.

Annex C
(informative)

Compressed data format specification

C.1 Introduction

In some cases additional requirements come into play where a great amount of data shall be transported in one location reference package. Doing so, it causes the desire to shrink the overall amount of data size to a valuable amount. The experiments with location references turned out, that one good approach is to code all roads of a network in one. By having a much more homogenous data set a very effective compression helps to shrink the data size to a quite good size factor.

The explains this data format of compressed storing of location references with some topological points and some geometric road shapes.

This annex specifies only a binary data format because this specification applies to the communication between centres and terminals such as car equipments when small capacity media like broadcast media are adapted.

C.2 Compressed-format specific terms

C.2.1 deflection angle

The deflection angle shows the direction difference between the internal link (see ISO 17572-1:2008, Figure A.1) from targeted point to next point and the internal link from upstream point to targeted point. If the deflection angle between two internal links is 0, these internal links line up on a straight line. If the deflection angle is not 0, the deflection has a positive and negative sign, with the clockwise direction being positive.

C.2.2 predicted angle value

The predicted value is the targeted deflection angle predicted by prediction formula using deflection angles on the upstream side. Its prediction formula shall change according to the features of the aimed road segment.

C.2.3 prediction formula

The prediction formula presumes targeted deflection angle in upstream direction.

C.2.4 predicted angle difference value

This value expresses the difference between the deflection angle predicted by the formula and the actual deflection angle of the targeted point.

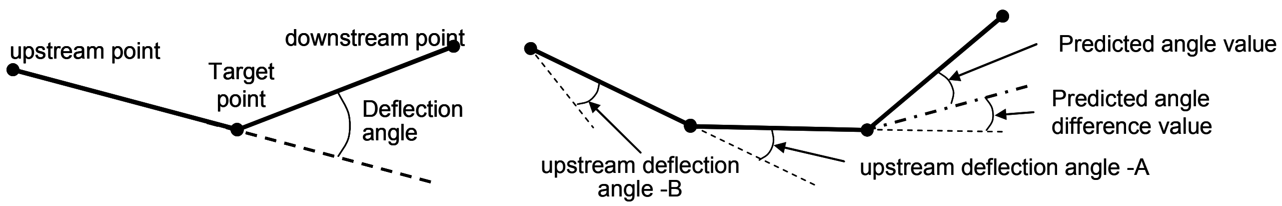

Figure C.1 — Deflection angle and predicted angle difference value

C.2.5 quantum predicted angle difference value

Predefined values of predicted angle difference value expressed per some degrees (resolution).

C.2.6 curvature, radius of curvature

The curvature is defined originally as follows: in a two dimensional curve, points are exhibited on the curve of length L measured from a fixed point, the angle ω is defined as the angle between the tangent lines of two points P1 (L) and P2 (L + ΔL) separated by distance ΔL, and the curvature is defined as the differential value dω/dL at P1 (ω/ L as ΔL→0). In case of a circle with the radius r, its curvature is 1/r.

In this method, however, the shape data expressed by a string of x-y coordinates assumes straight lines between two adjoining points; straight lines are only used to set ranges of values. Thus, the curvature calculated from the straight lines between points approaches the uniform curvature. ω or θ is calculated by the simple calculation methods in Figure C.2 and the curvature is approximated by ω/D or θ/D. The radius of curvature is a reciprocal of the curvature.

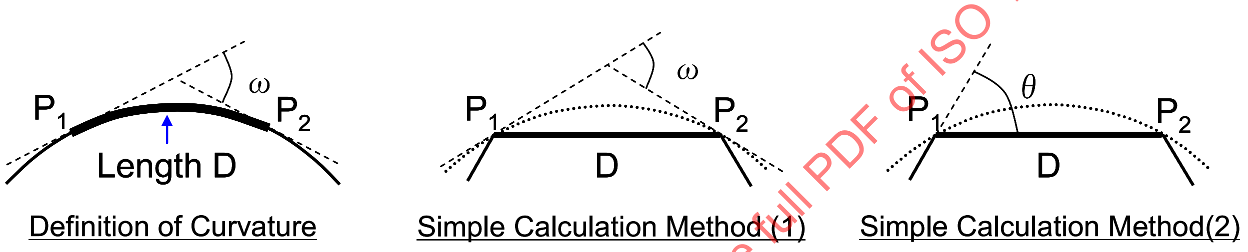

Figure C.2 — Curvature

C.3 Overall procedure of data compression

C.3.1 General

As shown in Figure C.3, two steps describe the process of encoding and decoding process. Both initial input data to the encoder side and final output data from the decoder side become the shape (of coordinates) describing the line shape by a string of coordinate shape data.

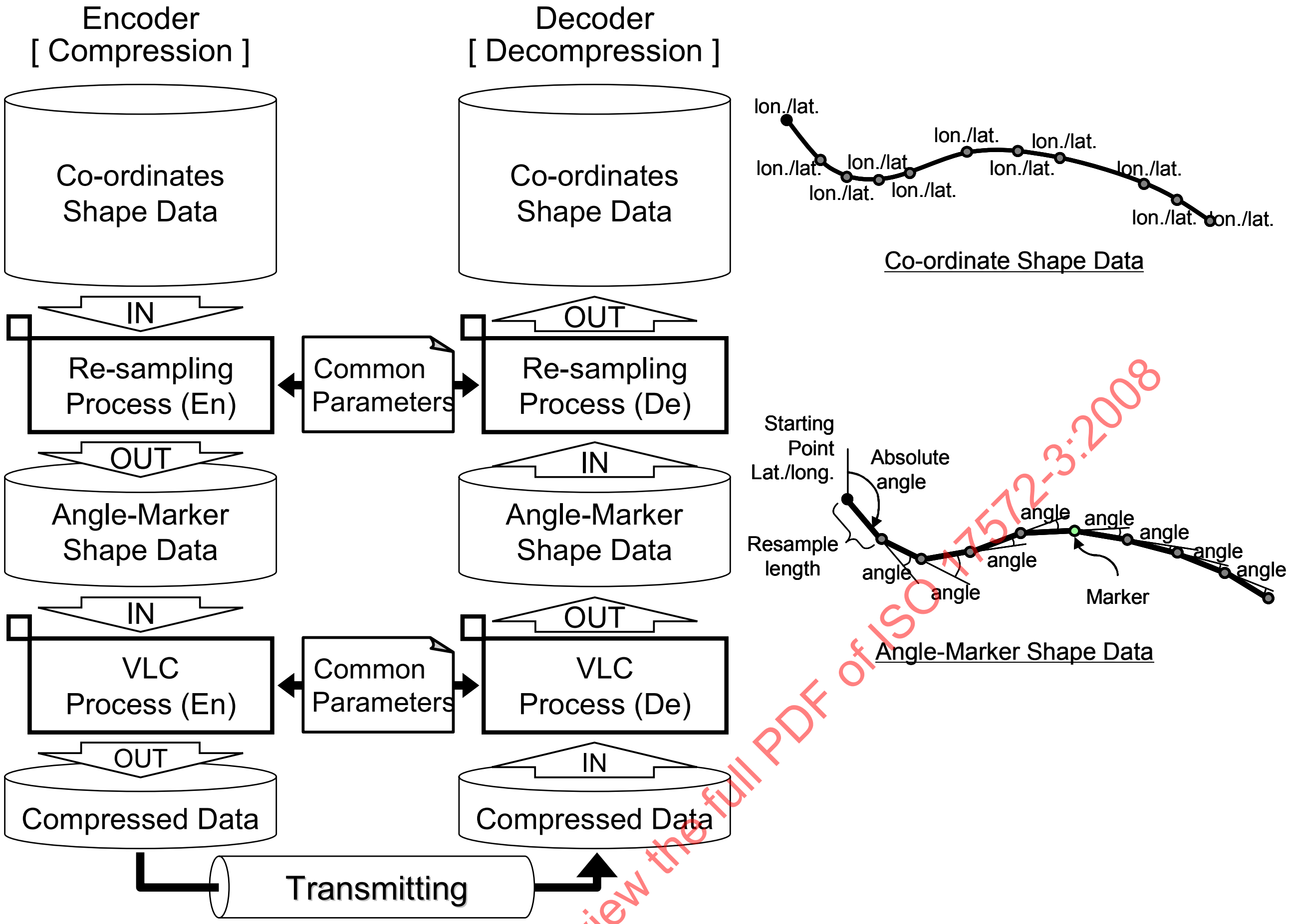

Figure C.3 — Procedure of compression and decompression

In the first step (see C.3.2), the input shape is modelled and the too precise information is removed. This process is called the “Resampling process”. As a result of the resampling process, the coordinate shape data is replaced with *angle marker shape data* which consists of *quantum angles and various markers*. This is a format allowing easy compression.

In the second step (see C.3.3) redundancy is eliminated in the data structure through so-called entropy coding compression algorithm. This method is also called “Variable-Length-Coding” (VLC). Among various entropy coding methods this compression method uses the “Adaptive-type Range-coder” format . As a result of the VLC, the data to be exchanged is output as compressed data. On the decoder side, the resampling process and the VLC become reversed conversion processes. The encoding and decoding processes have same parameters for their respective resampling process and VLC.

C.3.2 Resampling process

C.3.2.1 General

The resampling length L considering the radius of curvature is selected and it is ensured that lengths are equal in the resampling period. The variables used are reduced to one angle when the road shape information is converted from (X, Y) to coordinates to (L, θ) coordinates because resampling is done by equal lengths L during a given period. If the radius of curvature changes on the course of one shape segment, the resampling length is changed. Next, the above angle data θ string is converted to the deflection angle Φ string and finally to the predicted difference value ΔΦ string. See Figure C.4 for this step.

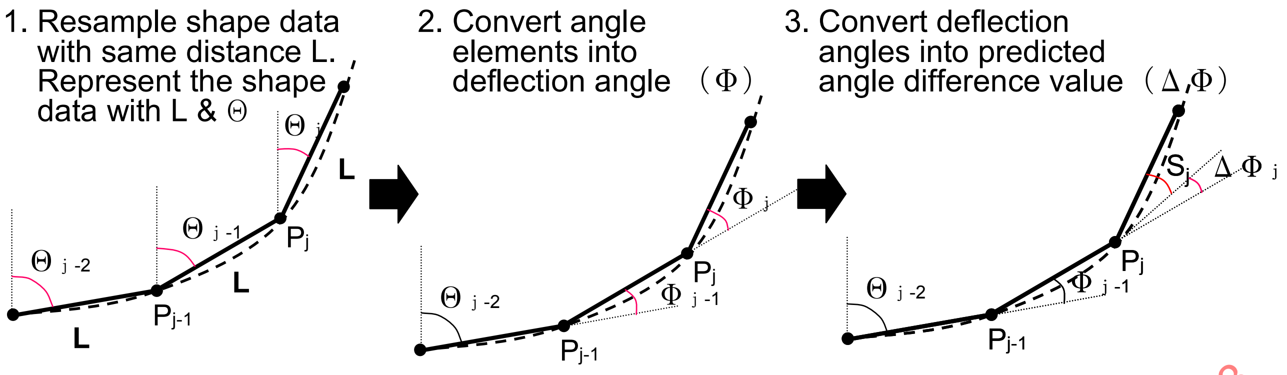

Figure C.4 — Procedure of resampling process

After the resampling process the road shape data consists of various markers (resampling length marker, prediction formula marker, attribute marker, etc.) and quantum angle values (quantum predicted difference values). These quantum angle values and markers are indicated by one byte (0-255) to manage these data uniformly. Directly stated, 0-180 is allocated to the quantum angle values and 181-255 is allocated to markers. If these are expressed as above, quantum angle values and markers can be managed by the following structure uniformly.

```
[Quantum angle value or Marker symbol] + [Incidental code1] + [Incidental code2]
    + … + [Incidental code7]
```

The number of incidental codes and the content of each code can be identified uniquely by this symbol. The outline configuration of the deflection Angle-Marker shape data and the data structure are shown in Figure C.5 and Table C.5. Table C.5 shows then a list of all currently defined Deflection angle marker symbols.

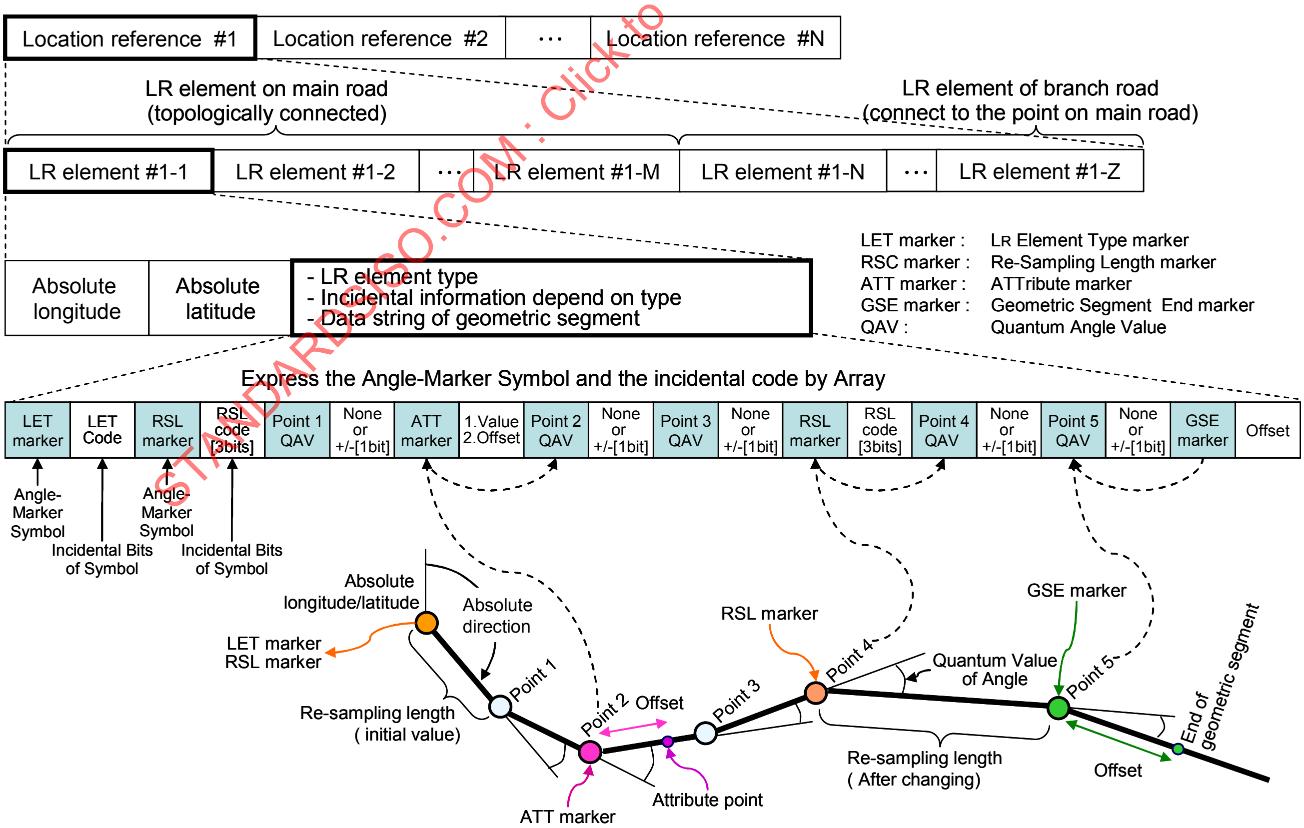

Figure C.5 — Example of a bit stream of angle-marker shape data

C.3.2.2 Description of starting point

The shape segment starts with an absolute value of coordinates and bearing.

```
[Starting point longitude]
 + [Starting point latitude]
 + [Bearing]
```

C.3.2.3 Description of quantum angle value

The predicted difference value is given a quantum angle value according to the resampling length and an incidental code for positive/negative sign (+,–), and is expressed by the following format. If its quantum angle value is 0, it needs no incidental code.

— If the angle value is 0:
`[The quantum angle value]`

— If the angle value is not 0:
`[The quantum angle value] + [Positive and negative sign]`

The positive and negative sign uses one bit (the positive sign is 0, the negative sign is 1). The angle is expressed in degrees (°) and the most detailed resolution is 1°. This method fixes the quantum angle parameter δ of each re-producing mode.

C.3.2.4 Description of shape control marker

C.3.2.4.1 General

The *shape control marker* allows changing the actual quantum predicted angle difference value differs according to the compression parameters, resampling length and the prediction formula. Into the compression parameters, resampling length and the prediction formula are inserted to the shape control markers at a change point as follows:

At the resampling shape segment furthest upstream following the starting point (down to where the first quantum angle value appears), the resampling length marker with the initial value are optionally attached in order to allow the following set of angle values.

The prediction formula of the starting point is always prediction formula [deflection angle], because the starting point has no deflection angle with the previous segment.

C.3.2.4.2 Geometric segment end marker

The *geometric segment end marker* is set on the end of geometric segment to describe the end point. The end of geometric segment usually doesn't coincide with resampling points. So "Geometric segment end marker" is set to the end resampling point and given "offset" described on C.3.2.5.2. The end point is represented by "Geometric segment end marker" and "offset". The structure of "Geometric segment end marker" shows as follows;

```
[Geometric segment end marker] + (Offset)
```


C.3.2.4.3 Prediction formula marker

The prediction formula change point: The predicting formula code is 2 bits and indicates three prediction formulas.

Three prediction formulas are defined as following:

— Prediction formula1 predicted value = 0
— Prediction formula2 predicted value = nearest upstream deflection angle
— Prediction formula3 predicted value = Roundup [(nearest upstream deflection angle + second nearest one) / 2)]

```
[Prediction formula change marker] + [Prediction formula code]
```

C.3.2.4.4 Resampling length marker

The resampling length change point: The resampling code is 3 bits and indicates eight resampling lengths.

```
[Sampling length change Marker] + [Sampling length code]
```

C.3.2.5 Encoding rules for attribute marker

C.3.2.5.1 General

The attributes are designated to the location i.e. overall shape segment or to a point. All attributes are expressed by attachment of an “Attribute Marker”. If an attribute is given to points, the resampling points may be not at the same position as the attributes were before the resampling process. In order to reproduce points with attributes, the resampling point prior to the intended attribute position is carrying the “attribute marker” and the attribute (refer to Figure C.6).

C.3.2.5.2 Offset

Offset expresses the distance from a resampling point given various attribute markers to the original position of the attribute (refer to Figure C.6).

To the `[Geometric segment end marker]` an offset is attached which is the distance from the last resampling point given a Marker, to the coordinates of the end point of the shape segment and then this section becomes a valid shape segment.

An `[Attribute Marker]` is given an offset which is the distance D from the resampling point given an attribute marker to the point with that attribute in the coordinate shape data, and the location shown by the offset is the point with the attribute. The maximum error with regard to the original position is made D/2[m], because what ever is nearer to the points to front or back is chosen. The forward and backward reproducing resolution D[m] of the offset is also the reproducing (compression rate) parameter. So if D becomes smaller, the reproducing preciseness of the position of the point with the attribute increases, but the number of bits needed to express the offset value becomes larger. Conversely, if D becomes larger, the reproducing preciseness decreases, but the bit number is small.