

# ADASIS v2 Protocol

Version 2.0.3.0

December 2013

**Proprietary and Confidential**

**Document control sheet**

ERTICO project title: Advanced Driver Assistance Systems Interface Specifications

Document title: ADASIS v2 Protocol

Deliverable number: D2.2

Electronic reference: 200v2.0.3-D2.2-ADASIS\_v2\_Specification.0

Main author(s) or editor(s): Alexander Bracht (Daimler)

Current author(s): Bernd Raichle (Daimler), Manuel Rauch (Continental), Axel Varchmin (Bosch), Johannes Stille (Nokia HERE)

Contributing author(s): Stephen T'Siobbel (TomTom), Sabine Marwitz (TomTom), Christian Ress (Ford), Nic Smith (Nokia HERE), Jan Löwenau (BMW)

Previous author(s): Sinisa Durekovic (NAVTEQ), Julian Requejo (Ford), Dmitri Toropov (Ford), Dirk Balzer (OPEL/GM), Michael Wagner (OPEL/GM), Martin Griesbeck (Continental), Michel Mittaz (Bosch/Blaupunkt)

Document history:

Version	Status	Date	Main author	Organization	Summary of changes
2.0.0.0	Release	April 13 <sup>th</sup> , 2010			<ul style="list-style-type: none"> <li>• META-DATA message revised</li> <li>• CAN layouts revised</li> <li>• Document structure modified</li> <li>• Number of smaller changes in text</li> </ul>
2.0.0.1	Release	August 12 <sup>th</sup> , 2010			<ul style="list-style-type: none"> <li>• Changed ERTICO logo</li> </ul>
2.0.1.0	Release	Feb 23 <sup>rd</sup> , 2011	S.Durekovic A. Bracht	Navteq Daimler	<ul style="list-style-type: none"> <li>• Added proposal for ALTITUDE profile.</li> <li>• Added proposal for Bezier PROFILE (section 10.6).</li> <li>• Added proposal for Permanent Link Id PROFILE (section 10.7).</li> <li>• Updated color coding in table 30, "Curvature Coding" in section 10.1. Color coding in line 831 does now follow the regular scheme.</li> <li>• Reformatted inequations of curvature decoding in section 10.1.2. Mathematical meaning did not change, but reformatting shows better the regular scheme with powers of two.</li> <li>• Corrected typo in Table 14 "Stub Message". Turn Angle value 126 was defined twice.</li> <li>• Limited allowed number range of Altitude profile (Table 22) to -1000m to +10.000m</li> <li>• Corrected typo in right boundary value "value 1" in Table 25</li> <li>• <b>The version 2.0.1.0 is completely compatible to 2.0.0.0.</b> <b>All changes were additions of profile types or typo corrections.</b></li> </ul>

2.0.2.0	Release	July 10 <sup>th</sup> , 2012	A. Bracht S. Durekovic M. Rauch	Daimler Nokia L&C Continental	<ul style="list-style-type: none"> <li>Table 12: Effective Speed Limit (5-bit), value 31: interpretation changed from "N/A" to Invalid.</li> <li>Section 6.8 Change of Speed Limit units in run-time: explanation reworked and recommendation added to re-start horizon</li> <li>Section 5.8 ADASIS v2 Mini: Added table of default values</li> <li>Section 4.8 PROFILE LONG Message: added that altitude shall be above WG84 reference ellipsoid</li> <li>Section 4.9 META-DATA Message: Added IPC and NavInfo to map provider list. Changed name of TeleAtlas to TomTom.</li> <li>Introduced new section 4.10 SYSTEM SPECIFIC Message</li> <li>Added specification for truck specific speed limits</li> <li>Added section 8.3 Long-Range Horizon with decreased Offset Resolution</li> <li>Added section 9 Current Usage of ADASIS by Different Partners</li> <li>Moved section "Clothoid Profiles", "Cubic Bezier Spline Profile", "Link Identifier Profile" and "Truck Speed Limit profile" from section 8 to section 9</li> <li>Added Copyright notice</li> <li><b>The version 2.0.2.0 is completely compatible to 2.0.1.0. All changes were additions or editorial improvements.</b></li> </ul>
2.0.3.0	Release	Dec 16th, 2013	A. Bracht	Daimler	<ul style="list-style-type: none"> <li>Deleted section 3.4.4 "Unlimited Horizon Length" and sect. 5.7 "Unlimited Horizons". It turned out, that the ADASIS options for unlimited length are too complex and that no partner uses them. Consequently, they have been removed from the standard.</li> <li>Added LONG PROFILE "Extended Lane" and "Traffic Sign"</li> <li>Reworked Section 3.4</li> <li>Editorial Improvements</li> <li><b>The version 2.0.3.0 is compatible to 2.0.2.0 for syntax and semantics of the CAN signals</b></li> </ul>

Review / Circulation:

Recipient	Date
ADASIS forum members v0.60	July 15, 2009
ADASIS forum members v2.0.0.0	April 27, 2010
ADASIS forum members v2.0.1.0	Feb 23, 2011
ADASIS forum members v2.0.2.0	July 10, 2012
ADASIS forum members v2.0.3.0	Dec 16, 2013

Abstract:

The ADASIS forum seeks to standardize the interface to predict the road geometry with its related attributes ahead of a vehicle based on the vehicle's position and a digital map, so called *ADAS Horizon*.

Copyright:

This specification is protected by copyright laws and contains material proprietary to the ADASIS Forum, a Forum coordinated by ERTICO - ITS Europe. It or any components may not be reproduced, republished, distributed, transmitted, displayed, broadcast, or otherwise exploited in any manner without the express prior written permission of the ADASIS Forum. You may use this specification for implementing the functionality therein, without altering or removing any trademark, copyright or other notice from the specification, but the receipt or possession of this specification does not convey any rights to reproduce, disclose, or distribute its contents, or to manufacture, use, or sell anything that it may describe, in whole or in part.

The usage, implementation or distribution of this specification is only allowed when following the rules as defined in the ADASIS Terms of Reference (as available on <http://www.ertico.com/adasisforum/>).

## **Table of Contents**

<b>1</b>	<b>INTRODUCTION .....</b>	<b>21</b>
<b>2</b>	<b>VERSIONING .....</b>	<b>24</b>
2.1	PROTOCOL VERSIONING.....	24
2.2	API VERSIONING.....	24
2.3	VERSIONING FOR THIS DOCUMENT .....	24
<b>3</b>	<b>ADASIS V2 CONCEPT .....</b>	<b>26</b>
3.1	ADASIS V2 OR OPTIMIZED PATH REPRESENTATION OF THE ADAS HORIZON.....	30
3.2	ADASIS V2 BUILDING BLOCKS .....	32
3.3	POSSIBLE CONFIGURATIONS OF THE ADASIS V2 HORIZON PROVIDER .....	32
3.4	PATHS AND OFFSETS.....	35
3.4.1	<i>Definition of Offsets.....</i>	<i>35</i>
3.4.2	<i>Cyclic Offset Value.....</i>	<i>36</i>
3.4.3	<i>Path Length Limits .....</i>	<i>37</i>
3.4.4	<i>Horizon Trailing Length .....</i>	<i>40</i>
3.5	PATH PROFILES.....	42
<b>4</b>	<b>MESSAGES.....</b>	<b>45</b>
4.1	MESSAGE TYPES AND CAN FRAME MULTIPLEXING .....	46

4.2	INTEL CAN MESSAGE LAYOUTS .....	50
4.3	ALTERNATIVE BUS CONFIGURATIONS.....	50
4.4	POSITION MESSAGE.....	51
4.5	SEGMENT MESSAGE .....	56
4.6	STUB MESSAGE.....	66
4.7	PROFILE SHORT MESSAGE.....	72
4.8	PROFILE LONG MESSAGE .....	79
4.9	META-DATA MESSAGE.....	84
4.10	SYSTEM SPECIFIC MESSAGE .....	89
<b>5</b>	<b>USE OF ADASIS V2 PROTOCOL .....</b>	<b>90</b>
5.1	REQUIRED MESSAGES.....	90
5.2	STUB TYPES .....	91
5.3	IMPLICIT STUBS.....	92
5.4	IMPLICIT ENTITY REMOVAL ON CLIENT .....	93
5.5	ADAS HORIZON PATH CREATION IN CASE OF VEHICLE POSITION JUMPS AND U-TURNS.....	98
5.5.1	<i>Backward Jump of Position Offset.....</i>	<i>98</i>
5.5.2	<i>Vehicle makes a U-Turn Maneuver.....</i>	<i>98</i>
5.5.3	<i>Vehicle drives backward.....</i>	<i>98</i>
5.6	UPDATING OF CONTENT AFTER TRANSMISSION .....	98



5.7	PROFILE INTERPOLATION USING ANALYTIC CURVES .....	99
5.7.1	<i>Basic approach</i> .....	100
5.7.2	<i>Complex interpolation functions</i> .....	103
5.8	ADASIS v2 MINI.....	107
<b>6</b>	<b>GUIDELINES AND RECOMMENDATIONS .....</b>	<b>109</b>
6.1	ADAS v2 HORIZON PROVIDER (AV2HP).....	109
6.2	MESSAGE TIMINGS.....	111
6.3	POSITION MESSAGE PRIORITY .....	111
6.4	USE OF ALTERNATIVE POSITIONS IN PREVENTION OF ADAS HORIZON RE-INITIALIZATION .....	113
6.5	PROFILE INTERPOLATION: PRE-CALCULATED DATA OR REAL-TIME APPROXIMATION .....	116
6.6	ABSOLUTE ROAD GEOMETRY .....	118
6.7	SAMPLE SYSTEM-SPECIFIC PROFILES.....	119
6.8	CHANGE OF SPEED LIMIT UNITS IN RUN-TIME.....	120
<b>7</b>	<b>ERROR DETECTION AND RECOVERY .....</b>	<b>121</b>
7.1	MISSING POSITION MESSAGE .....	121
7.2	MISSING SEGMENT MESSAGE.....	121
7.3	MISSING STUB MESSAGE.....	122
7.4	MISSING PROFILE MESSAGE .....	122
7.5	MISSING META-DATA A MESSAGE .....	122

<b>8</b>	<b>EXAMPLES OF USE .....</b>	<b>123</b>
8.1	DIFFERENT ADAS HORIZON REQUIREMENTS IN THE VEHICLE NETWORK.....	123
8.1.1	<i>Example Set-up 1: Cyclic offsets with different path lengths.....</i>	<i>124</i>
8.1.2	<i>Example set-up 2: Multi-path and main path only .....</i>	<i>125</i>
8.2	EXAMPLE APPLICATIONS AND POSSIBLE CONFIGURATIONS .....	126
8.2.1	<i>Speed Limit Warning .....</i>	<i>126</i>
8.2.2	<i>Curve Warning.....</i>	<i>127</i>
8.2.3	<i>Intersection Warning.....</i>	<i>127</i>
8.2.4	<i>Lane Change Assist.....</i>	<i>127</i>
8.2.5	<i>Vehicle Dynamics Adjustment.....</i>	<i>127</i>
8.2.6	<i>Fuel Saving Application.....</i>	<i>127</i>
8.3	LONG-RANGE HORIZON WITH DECREASED OFFSET RESOLUTION .....	128
<b>9</b>	<b>CURRENT USAGE OF ADASIS BY DIFFERENT PARTNERS.....</b>	<b>129</b>
<b>10</b>	<b>APPENDIX – DETAILS FOR CERTAIN PROFILES .....</b>	<b>132</b>
10.1	CURVATURE INTERPRETATION.....	132
10.1.1	<i>Coding method.....</i>	<i>135</i>
10.1.2	<i>Decoding method .....</i>	<i>136</i>
10.2	ROUTE NUMBER TYPES .....	138
10.3	HEADING CHANGE DEFINITION.....	145

10.4	CODING SCHEME FOR “REGION CODE” .....	146
10.5	CLOTHOID PROFILES .....	149
10.5.1	<i>Clothoid for Relative Geometry</i> .....	150
10.5.2	<i>Clothoid for Absolute Geometry</i> .....	151
10.6	CUBIC BÉZIER SPLINE PROFILE .....	152
10.7	LINK IDENTIFIER PROFILE .....	154
10.8	TRAFFIC SIGN AND SITUATION PROFILE .....	156
10.9	TRUCK SPEED LIMIT PROFILE .....	165
10.10	EXTENDED LANE PROFILE .....	169
10.10.1	<i>Lane Connectivity</i> .....	170
10.10.1.1	Example: Lane ends .....	171
10.10.1.2	Example: Exit Lane .....	172
10.10.1.3	Example: Merging Highways .....	172
10.10.1.4	Example: Merging Lanes without Priority .....	173
10.10.1.5	Example: Single Lane widens on both sides into Two Lanes .....	173
10.10.1.6	Example: New Lane on the Right Side .....	174
10.10.1.7	Example: Toll Plaza .....	175
10.10.1.8	Example: Turning Lane .....	176
10.10.1.9	Example: Roundabout .....	177

---

10.10.2	<i>Lane Attributes</i> .....	178
11	<b>REFERENCES</b> .....	<b>181</b>

## **List of Figures**

Figure 1: ADASIS v2 System Architecture .....	22
Figure 2: Road network .....	26
Figure 3: Digital Map Database .....	27
Figure 4: ADAS Horizon.....	28
Figure 5: ADAS Horizon – (Pure) Path representation .....	29
Figure 6: ADASIS v2 Horizon - Optimized Path Representation .....	31
Figure 7: ADASIS v2 Horizon - Application View.....	32
Figure 8: Main Path only.....	33
Figure 9: Main Path with Stubs .....	33
Figure 10: Main Path with first level Sub-paths .....	34
Figure 11: Full Horizon.....	34
Figure 12: Positioning of entities on ADASIS v2 Horizon.....	35
Figure 13: Short ADAS Horizon.....	36
Figure 14: Limited Cyclic Offset .....	37
Figure 15: Definition of Path Lengths .....	39
Figure 16: Visualisation of the ADASISv2 Horizon Length, Trailing Length and Forbidden Offset Range .....	41
Figure 17: Profiles.....	42

Figure 18: Three types of STUB message .....	92
Figure 19: PROFILES in ADASIS v2 – optimized for bandwidth.....	101
Figure 20: PROFILES in ADASIS v2 – optimized for robustness.....	101
Figure 21: Step Interpolation.....	102
Figure 22: Linear Interpolation.....	103
Figure 23: Second-degree polynomial interpolations .....	104
Figure 24: Re-Use of Path Indices.....	110
Figure 25: Alternative positions .....	115
Figure 26: Shape-point based profile example (here: curvature radius r) .....	116
Figure 27: Optimized shape-point based profile .....	117
Figure 28: Possible Vehicle Network with different ADAS Applications .....	123
Figure 29: Different Horizon lengths after start-up .....	124
Figure 30: Extension of short Horizon with Message Retransmission .....	125
Figure 31: Message Retransmission for Sub Paths.....	125
Figure 32: Calculation of Heading Change angle.....	145
Figure 33: Heading Changes along a Path .....	146
Figure 34: Unit clothoid.....	149
Figure 35 Straight (blue); transition curve (clothoid, red); circular arc (green) .....	150

Figure 36: Clothoid Representation of Roads.....	151
Figure 37: Cubic Bézier Spline (red) and its control points .....	153
Figure 38: Permanent Link Id Profile .....	155

## **List of Tables**

Table 1: Message Type field .....	46
Table 2: ADASIS v2 as fully multiplexed CAN protocol .....	47
Table 3: ADASIS v2 as non-multiplexed CAN protocol .....	48
Table 4: ADASIS v2 as partially multiplexed CAN protocol .....	49
Table 5: POSITION message .....	53
Table 6: CAN layout of POSITION message .....	54
Table 7: Path Indices with Special Meaning .....	55
Table 8: Current Lane values .....	56
Table 9: SEGMENT message .....	60
Table 10: CAN layout of SEGMENT message .....	61
Table 11: Form of Way (4-bit) .....	63
Table 12: Effective Speed Limit (5-bit) .....	64
Table 13: Speed Limit Type (3-bit) .....	65
Table 14: STUB message .....	70
Table 15: CAN layout of STUB message .....	71
Table 16: PROFILE SHORT message .....	73
Table 17: Standard PROFILE SHORT Types .....	77



Table 18: Variable Speed Sign Positions.....	77
Table 19: CAN layout of PROFILE SHORT .....	78
Table 20: PROFILE LONG message .....	80
Table 21: CAN layout of PROFILE LONG message.....	81
Table 22: Standard PROFILE LONG Types.....	83
Table 23: META-DATA message.....	87
Table 24: CAN layout of META-DATA message .....	88
Table 25: 5 <sup>th</sup> -degree polynomial PROFILE interpolation – Message 1 .....	105
Table 26: 5 <sup>th</sup> -degree polynomial PROFILE interpolation – Message 2.....	105
Table 27: 5 <sup>th</sup> -degree polynomial PROFILE interpolation – Message 3.....	106
Table 28: Default Values in ADASIS v2 Mini Message .....	108
Table 29: System Specific PROFILE SHORT Types.....	120
Table 30: Typical usage of ADASIS v2 Protocol.....	129
Table 31: Curvature Interpretation (10-bit) .....	133
Table 32: Route numbers types for selected countries .....	138
Table 33: Coding Scheme for field “Region Code” to map each ISO 3166-2 alphanumeric character to a 5 bit code .....	148
Table 34: Traffic Sign and Situation: Value (32 bit).....	159
Table 35: Traffic signs and situations: Type .....	164

Table 36: Truck Speed Limit .....	167
Table 37: CAN layout of LONG_PROFILE Message “Truck Speed Limit” .....	168
Table 38: Lane Type .....	179
Table 39: Line Marking .....	180

## Glossary of Terms:

ADAS	Advanced Driver Assistance Systems
ADASIS	ADAS Interface Specification
Av2	Version 2 of the ADASIS protocol definition (to distinguish between ADASIS version 1 that has been developed and tested in the Prevent Maps&ADAS project)
CAN (Controller Area Network)	A message based protocol used in automotive applications to allow devices to communicate
ADAS Horizon	Extract of the Map Database in front of the vehicle's current position
Horizon Provider (Av2HP)	The software component that generates the Electronic Horizon for transmission to other applications
Horizon Reconstructor (Av2HR)	Implementation of the Av2 Programming Interface that rebuilds the Horizon produced by the Av2HP
Network of Links	A collection of links, connected at their end points, forming the road network
Path	A possible route through the road network that a vehicle could follow
Stub	The attributes defining the start of a new path
Profile	The characteristics of a path at a specific distance along the associated path
Profile Type	Type of profile information, e.g. curvature or slope
Segment	Part of a Path where the attributes, most significant for ADAS applications, remain the same
Link	An underlying map database component representing the roadway between two intersections
Main Path (Most Probable Path, MPP)	The most likely route of the vehicle on the road, represented as a single Path
Sub Path	A path (other than the MPP) that the vehicle may potentially follow
First level sub path	Sub path that is directly attached to the main path
Second level sub path	Sub path that is directly attached to a first level sub path
Parent Path	Path to which a sub path is attached
Transmitted Path	Path that has been actually transmitted from the Provider to the Client via the CAN network
Path Index	Path identification number as used in the ADASISv2 CAN messages

Path Identifier	Internal path identification number of the horizon provider or horizon reconstructor that may have more values than the Path Index
Offset	Position along a path
Unknown Value	The value assigned to an attribute that is supported by the Horizon Provider but is not available at the current location or data item
Signal not Available Value (N/A)	The value assigned to an attribute that is not supported or not implemented by the current Horizon Provider
Spline	A representation of a curve using a few control points applied to a polynomial function
Clothoid	A curve whose curvature increases or decreases linearly with distance and describes the optimal form of a curve in the road network
Message Type	Type of a CAN message in the scope of the ADASISv2 protocol
Multiplexing	Usage of a multiplexor ID (e.g., message or profile type) to define different CAN message layouts under one CAN ID

## 1 Introduction

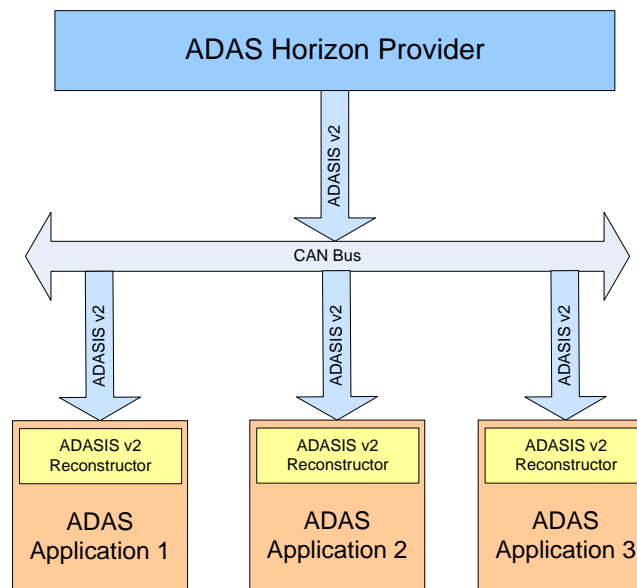
The ADASIS forum [1] and the Prevent/Maps&ADAS project [2][5][6][7] have developed a specification to describe the road geometry with its related attributes ahead of a vehicle based on the vehicle's position and a digital map (known as the *ADAS Horizon*). This specification ("ADASIS v1") has been prototypically implemented on a CAN bus by several partners and the performance and interoperability of the implementations have been successfully tested. The ADASIS concept has proven to be technically feasible.

However, the ADASISv1 interface has turned out to be rather complex to implement in the OEM production environment. In particular, the client ADAS applications needed to implement sophisticated horizon reconstruction software, which threatened the success of the standardization approach.

This document specifies a simplified ADASIS interface ("ADASIS v2") that makes the interpretation and reconstruction of ADAS Horizon information much easier.

In the general architecture of ADAS applications, the main entities are:

- **ADAS Horizon Provider**, which maintains the ADAS Horizon;
- **ADAS Protocol** that defines how the ADAS Horizon will be sent from the ADAS Horizon Provider to the ADAS Applications;
- **ADAS Application** is a client application that receives the ADAS Protocol messages then reconstructs and uses the ADAS Horizon;
  - **ADAS Reconstructor** is a common component of ADAS Applications that is built in accordance with this general architecture. The task of the ADAS Reconstructor is to receive, parse and interpret ADAS Protocol messages, and, in effect, reconstruct a copy of the ADAS Horizon on the client side.



**Figure 1: ADASIS v2 System Architecture**

In this document we will refer to above entities as **ADASIS v2 Horizon Provider (Av2HP)**, **ADASIS v2 Protocol (Av2)** and **ADASIS v2 Horizon Reconstructor (Av2HR)**. In addition, the ADASIS v2 Protocol assumes a CAN bus communication channel between the ADASIS v2 Horizon Provider and the ADAS Applications.

## 2 Versioning

### 2.1 Protocol versioning

The ADASIS v2 Protocol version is described with three numbers separated by points: **X.Y.Z**.

- **X** is the major version number. It is number 0, 2 or 3. Pre-releases of ADASIS v2 protocol will contain a major version number of 0; the first official release will have the major version number 2; the next version of protocol will be designated by the major version number 3.
- **Y** is the minor version number in the range 0-15. The minor version number will be increased each time the syntax of the protocol is changed, such as: new or removed messages, new or removed message fields, changed number of bits of a signal etc.
- **Z** is the minor sub-version number in the range 0-7. The minor sub-version number is increased each time the semantics of the protocol are modified – for instance, interpretation of field values or similar.

The number ranges are defined according to the version signals in the META-DATA message (see section 4.9).

### 2.2 API versioning

API version numbers follow the same convention as the protocol. In general, versioning of the API is independent of the versioning of the protocol.

### 2.3 Versioning for this document

Versions of this document, *ADASIS v2 Protocol*, are designated by **X.Y.Z.W**. The first three fields are the same as the corresponding protocol version. The last field, **W**, indicates changes in the document that are not related to changes in the protocol – revision of the text, changed or extended descriptions (but when syntax and semantics are kept intact), etc.



Working versions are labeled A, B, C. When **W** is a number 0 to 999, the specification document is approved for release.

The current version of this document is Version 2.0.3.0.

Because of naming conventions of the ERTICO documents, the name of this file will be in the form:

`200vX.Y.Z-D2.2-ADASIS v2 Specification.W.doc.`

For instance,

`200v0.7.3-D2.2-ADASIS v2 Specification.C.doc`

contains the third working version (**C**) of the **7<sup>th</sup>** revision of the protocol syntax and the **3<sup>rd</sup>** modifications of the semantics of the draft protocol specification.

Similarly,

`200v0.7.3-D2.2-ADASIS v2 Specification.20.doc`

refers to the same protocol syntax and semantics, but there were a number (20) of changes in the document such as better descriptions, new pictures etc.

The 200 at the beginning of the filename indicates that this document is the 200<sup>th</sup> of the ADASIS forum. The 200 is not version information. The "D2.2" means that this document is the **2<sup>nd</sup>** delivery document of the working group **2** in the ADASIS forum.

Please note that as auxiliary document to this one can find the file Vector CANdb++ that follows similar convention

`200vX.Y.Z-D2.2-ADASIS v2 CAN Database.W.dbc.`

### 3 ADASIS v2 Concept

In the digital map database, the road network (Figure 2) is represented as a collection of *links* (or *segments*) and *nodes* that define connectivity between links (Figure 3).

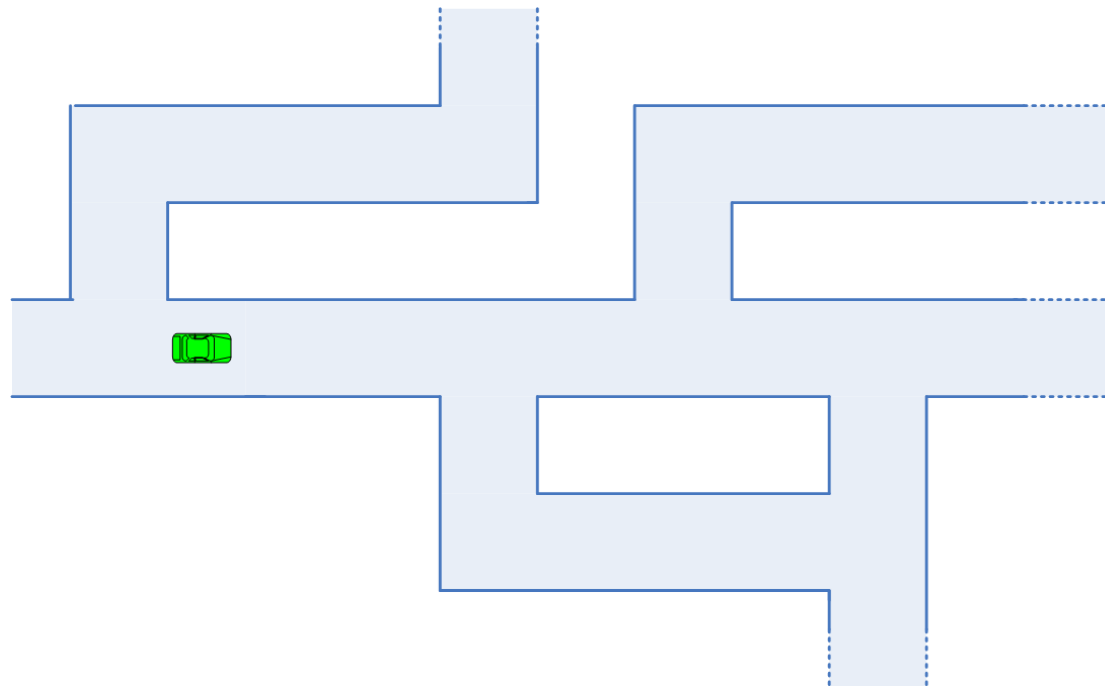
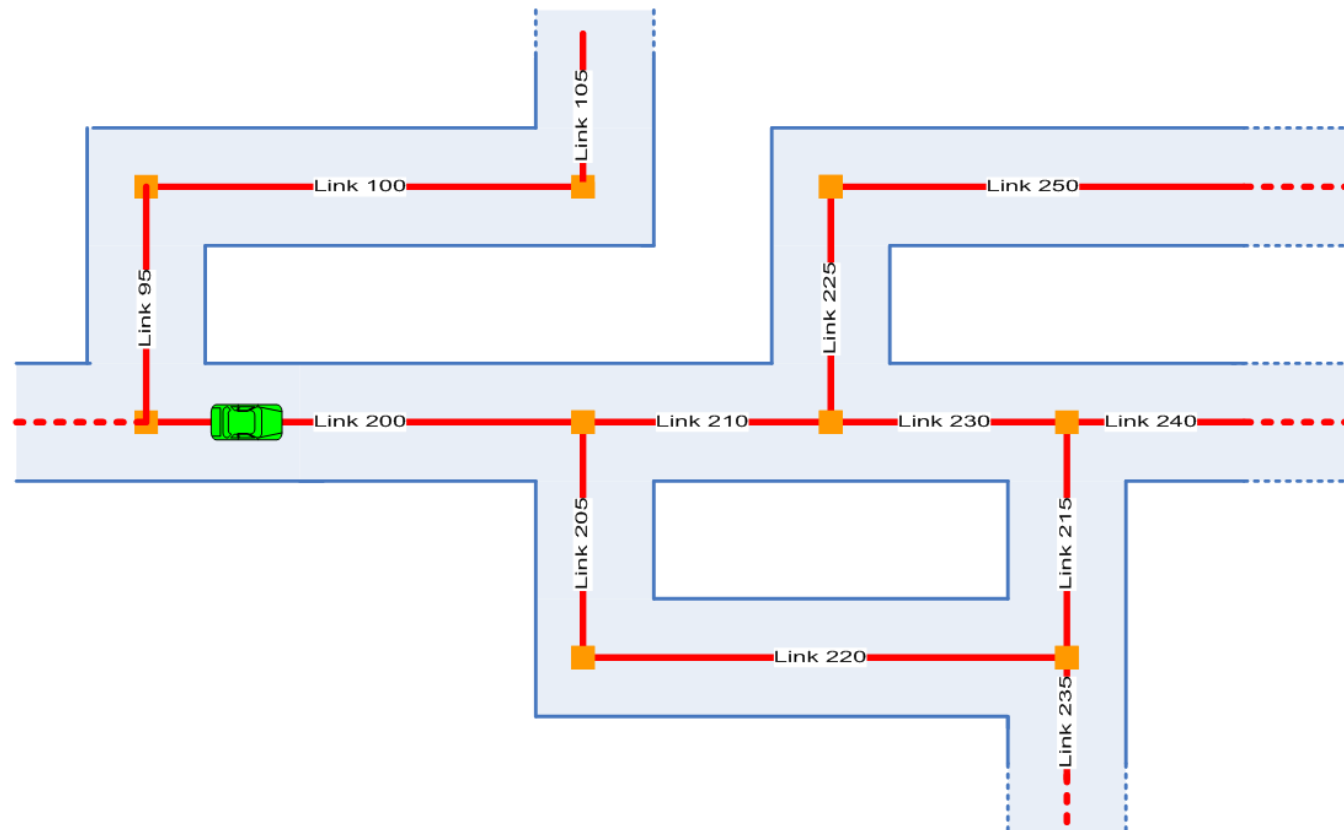


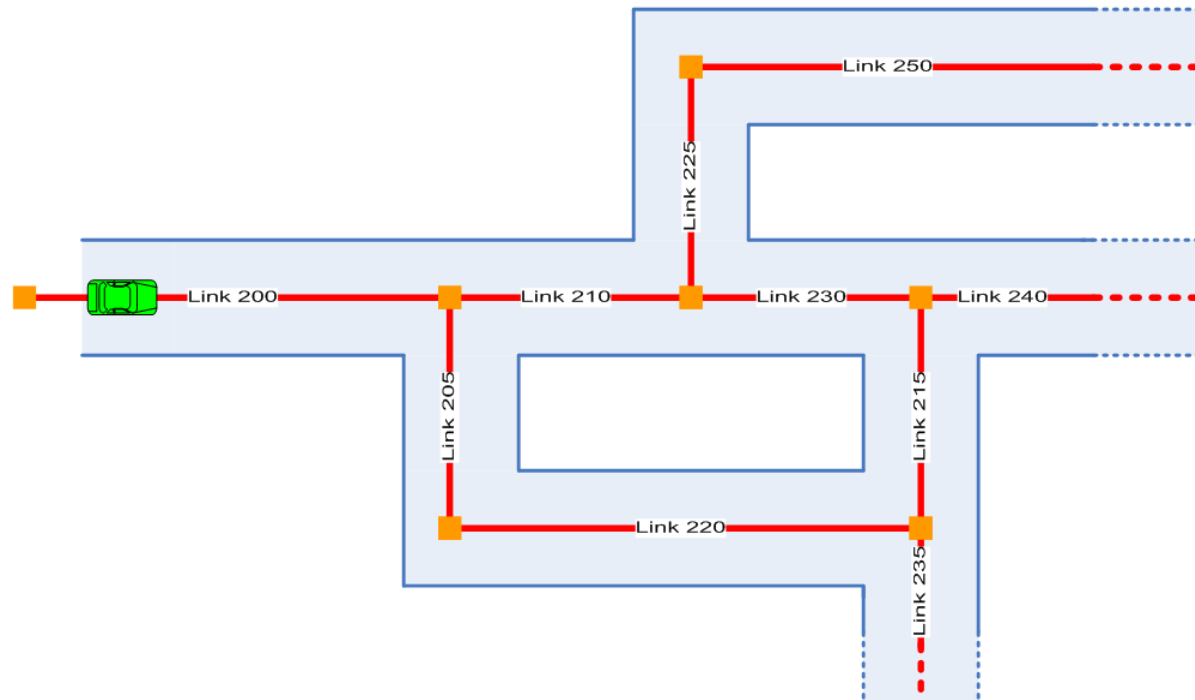
Figure 2: Road network



**Figure 3: Digital Map Database**

With regard to the map-enabled and map-enhanced ADAS application, the only roads of interest are those that are ahead of the vehicle and accessible in a reasonable time.

*The ADAS Horizon (Electronic Horizon, eHorizon...)* is the part of the digital map that contains only those roads in front of the vehicle (Figure 4).



**Figure 4: ADAS Horizon**

Comparing a simple extract of roads around the vehicle in Figure 3 and the ADAS Horizon in Figure 4, one can see that links 95, 100 and 105, being not important for majority of ADAS applications, are not in the ADAS Horizon. In addition, not all link attributes available in the digital map need to be present on links of the ADAS Horizon. For instance, street names or house number ranges will rarely be needed by ADAS applications.

In other words, the ADAS Horizon provides to ADAS application an optimized view of the environment, allowing for more efficient processing.

There are different algorithms for the construction of the ADAS Horizon. However descriptions of those algorithms are not within the scope of this document. One such implementation is described in [13].

The ADAS Horizon can be represented as a *Network of (Digital Map) Links*, as shown in Figure 4. For the ADAS application, this view is rather complex because it must analyze the network in order to get information about entities in front of the vehicle. For instance, if there is a traffic sign of interest on link 235, the application must realize by itself that this link can be reached by following links 200→210→230→215→235 or 200→205→220→235.

It is more convenient for the ADAS application to deal with *paths* – trajectories the vehicle may follow in the near future. Internally, paths are built from database links and their connectivity, but each path is seen by the application as a single entity.

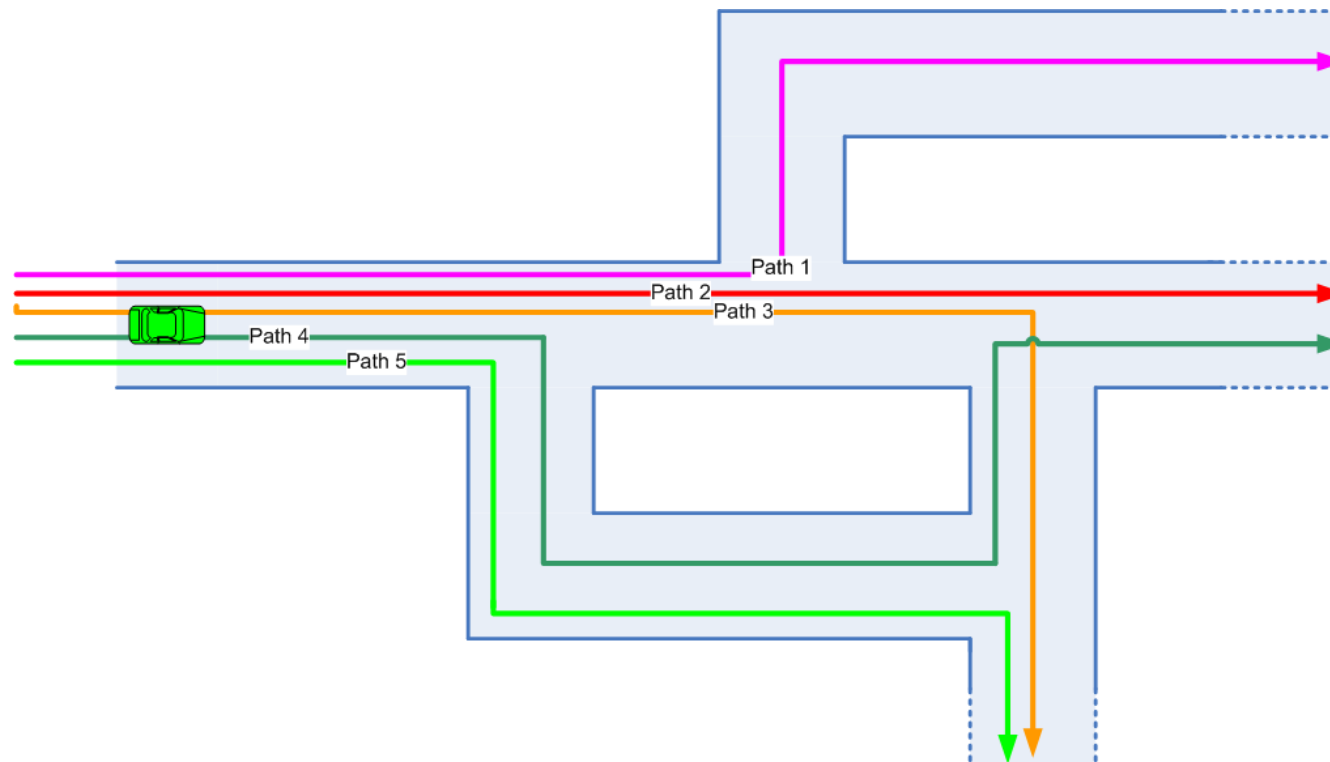


Figure 5: ADAS Horizon – (Pure) Path representation

Coming back to the original example, by querying path characteristics, the application can now easily recognize that there is a specific traffic sign on paths 3 and 5 (Figure 5). It may or may not be known that traffic signs on those two paths are the same physical traffic sign. For most applications, however, this information is not fundamentally important – the only significant information is that there is a traffic sign ahead and the distance to it.

*Pure Path Representation*, as shown in Figure 5, contains a number of redundancies. For instance, all five paths have a common start point. This characteristic is not very desirable if the ADAS Horizon is to be transferred over a slow communication channel.

*Optimized Path Representation* in ADASIS v2 reduces, but does not fully remove, such redundancies.

### 3.1 ADASIS v2 or Optimized Path Representation of the ADAS Horizon

The ADASIS v2 concept describes an ADAS Horizon using *Optimized Path Representation*. This approach reduces the amount of duplicated data, but still provides most of the advantages of the path approach over a network representation of the ADAS Horizon (Figure 6).

As a minimum, in Optimized Path Representation just one path needs to be present, let us call this the *Main Path*. The defining characteristic of the Main Path is that the current vehicle position is located on it. In Figure 6, Path 2 is the Main Path.

The vehicle may turn from the Main Path to *First-Level Sub-Paths*. In the above example Paths 1, 3 and 4 are first-level sub-paths.

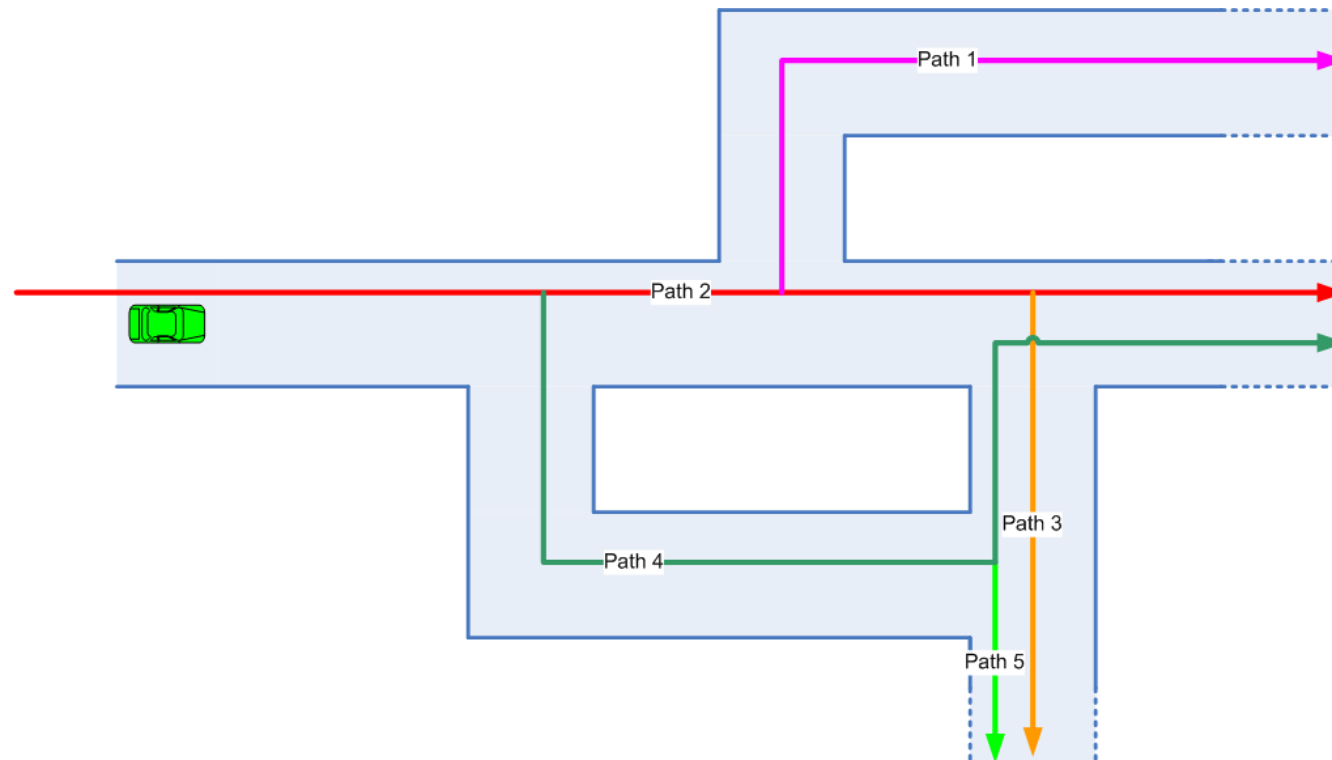
From first-level sub-paths there are possible turns to *Second-Level Sub-Paths* (Path 5, for instance, from Path 4), etc.

The ADAS Horizon's construction algorithm should choose the Main Path so that it appears to be the most likely alternative for the vehicle to continue driving. First-level sub-paths are less likely to be driven on and so on.

Most ADAS applications operate on the Most Probable (Main) path.

In distributed environments, with one Av2HP and one or more Av2 applications, question how many level of sub-paths shall be sent depends of the application tolerance to lack of horizon. If few seconds of non-functioning are acceptable, only main path can be sent over communication channel. After vehicle has left most probable path, some time will be necessary to send new one.

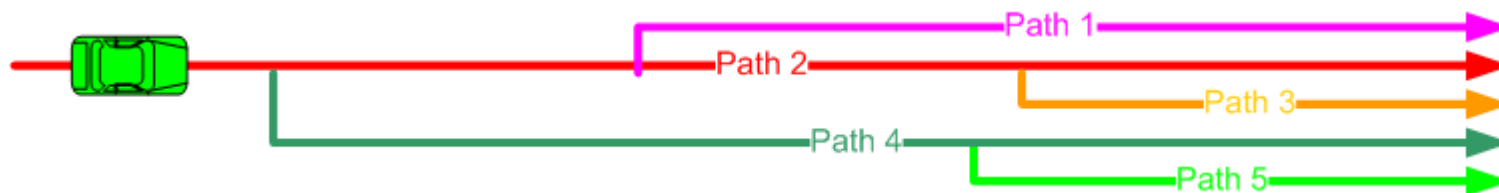
If application must have horizon data all the time, full horizon shall be transferred. This ensures that, even if vehicle leaves one path, it will automatically be positioned on another path that is already available at the client side.



**Figure 6: ADASIS v2 Horizon - Optimized Path Representation**

Since ADASIS v2 views the ADAS Horizon using Optimized Path Representation, let us call this representation the *ADASIS v2 Horizon*.

On the ADAS Horizon, crossings, road attributes and even geometry may be seen only as characteristics of (one of) the paths. Therefore, *Path* is the main entity that needs to be accessed by the application in order to retrieve the desired information.



**Figure 7: ADASIS v2 Horizon - Application View**

From the application viewpoint (Figure 7), the vehicle is located on one (main) path; the data of interest are either on the same path or on one of sub-paths ahead of the vehicle.

### 3.2 ADASIS v2 Building Blocks

As we have seen, the ADASIS v2 Horizon consists of *paths*. Each path is uniquely identified by a *path identifier*.

It will be shown that the relationships between paths are defined by entities called *stubs*. Each stub marks the start of a (sub-) path and it is located on another (parent) path. We will also introduce the concept of *profiles*, which are the characteristics of paths.

The position of Stubs and Profiles on the ADASIS v2 Horizon are defined by a path identifier and an *offset* along that path. All offsets are defined as the distance between an entity and the start of a path.

The current *Vehicle status* is described in terms of its position in respect to the Av2 Horizon (again path identifier and offset) and other characteristics such as speed and heading.

### 3.3 Possible Configurations of the ADASIS v2 Horizon Provider

In general, ADASIS v2 Horizon Providers should initially consider and calculate all possible paths.

However, the client ADAS application may need just a subset of those paths. Most applications will actually only need the main path. Therefore, in many installations it will be enough if the ADASIS v2 Horizon Provider (Av2HP) sends only the main path to the client (see Figure 8).



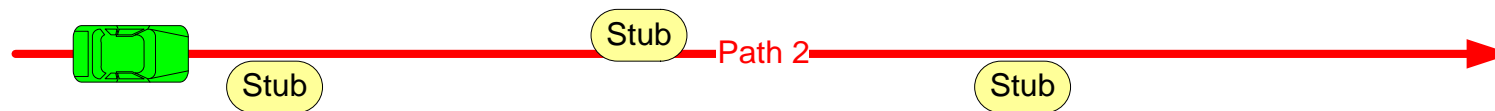


**Figure 8: Main Path only**

If only the main path has been provided, but the vehicle leaves it, the application will be “blind” for a moment until new information for the new path is available.

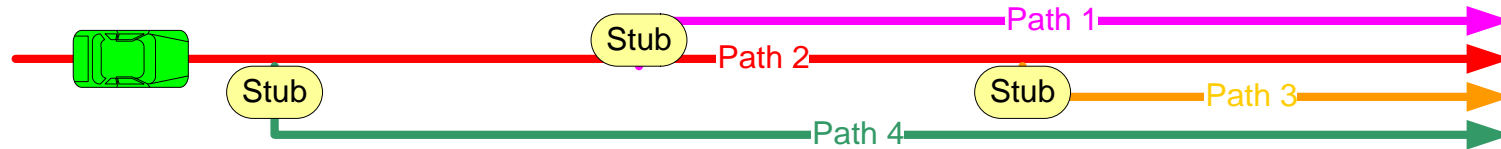
If the ADAS application also needs information for sub-paths, the ADASIS v2 Horizon Provider may offer extended configuration levels, also transmitting preview information reaching into sub path structures. This is done by using "stubs" which indicate the start of a new path attached to a parent path, and which contain basic information about the attached road (e.g. turn angle, road class). The detailed definition of "stubs" is described in section 4.6.

Starting from Figure 8, a next configuration level for the Horizon Provider would be the Main Path with attached stubs, which contain basic intersection information (see Figure 9).



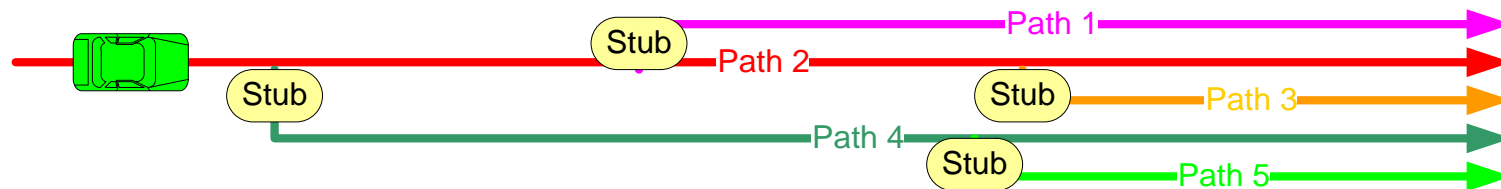
**Figure 9: Main Path with Stubs**

If the application is sensitive to a short "blindness" when the vehicle leaves the main path then the Horizon Provider must also preventively transmit information about upcoming sub paths. With the availability of sub paths, the client will receive the information when the vehicle position has, for example changed from Path 2 to Path 1, no gap will occur in the available track preview data on the client side.



**Figure 10: Main Path with first level Sub-paths**

If the client needs even more information and has enough memory to store it, the ADASIS v2 Horizon Provider can be configured to transmit the full available map information with higher-level sub paths as shown in Figure 11. In that case, the client will never be in a "blind" state and will always have immediate map information available when the vehicle leaves a path.



**Figure 11: Full Horizon**

The configuration of the ADASIS v2 Horizon Provider and the completeness of the transmitted horizon is installation specific and not defined in this document. Each vehicle installation may have a different configuration, depending on the requirements of the applications, the available bus bandwidth, and the storage capabilities of the application control units.

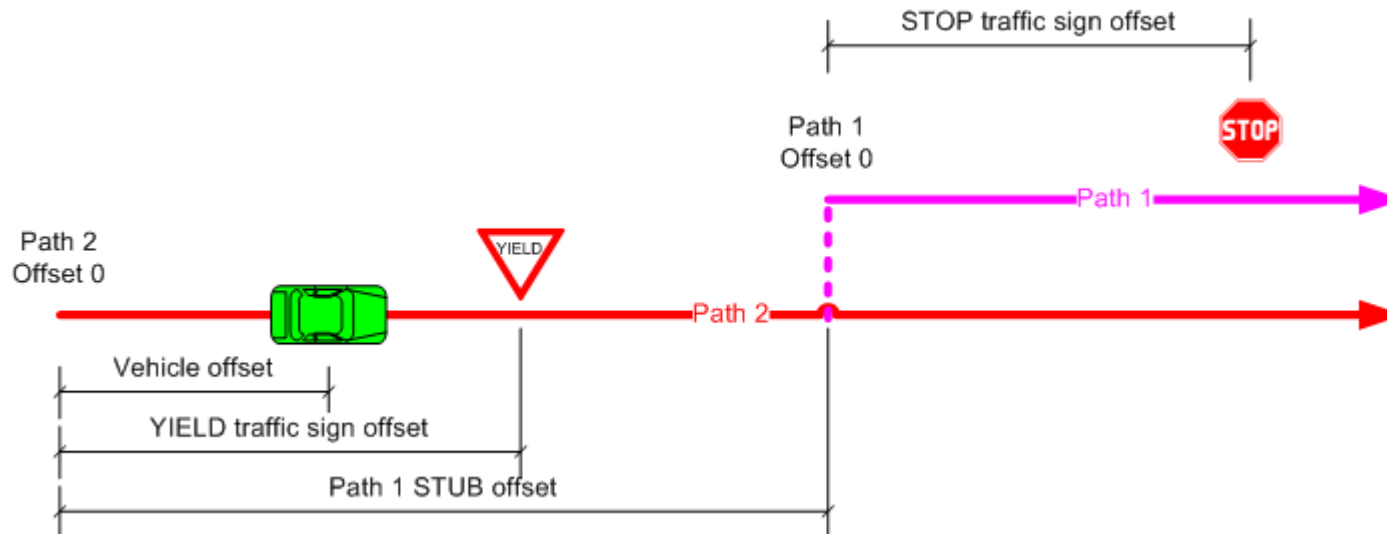
It can be assumed that there are simple applications needing only the main path transmission, as well as more complex applications needing the full horizon in one vehicle set-up. In this case, the ADASIS v2 Horizon Provider would provide the full horizon, but the simple application would be allowed to delete at the CAN receiver interface all information it cannot store or does not need. The ADASIS v2 protocol is designed in a way that simple applications are not forced to implement more ADASIS v2 functionality than they absolutely need and that more complex applications can use additional information layers (see also section 8.1).

## 3.4 Paths and Offsets

### 3.4.1 Definition of Offsets

As already described, each path is uniquely identified by some number (*Path Identifier*), the position of map entities and the vehicle position is defined by the path identifier and the distance from the start of the path. This along-path distance is called *Offset*.

The start of each path is defined to be at offset zero.



**Figure 12: Positioning of entities on ADASIS v2 Horizon**

At intersections encountered on the main path, sub-paths may branch off. These entries into branches are called *stubs*. Like other map entities, the location of *stubs* along the ADASIS v2 Horizon is described by specifying the (parent) path identifier and offset. Physical crossings on the path can be described as locations with one or more stubs – one for each alternate road at the crossing.

As we will see later, the ADASIS v2 Protocol defines only 6 bits for the identification of the path that is referenced by each CAN message. For the offset, just 13 bits are available. After removing special values, an ADASIS v2 message accommodates 56 numbers that can be used for path identification and offsets between 0 and 8190 meters inclusive.

For many ADAS applications, the limitation to maximum 56 paths and a maximum length of approximately 8 kilometres will not raise any problems. Many ADAS applications such as map-based Adaptive Cruise Control (ACC) or adaptive curve lighting will require only a short horizon length and very few paths. A possibility to transmit longer paths is described in section 8.3.

### 3.4.2 Cyclic Offset Value

The ADASISv2 CAN message definition allows only a limited number range to encode the offset value (13 bits). The following figures show an example for the "main path only" configuration with a maximum length of transmitted path of 1000 m and an offset wrap (= bit overflow) from 8190 to 0 at the end of the maximum offset range.

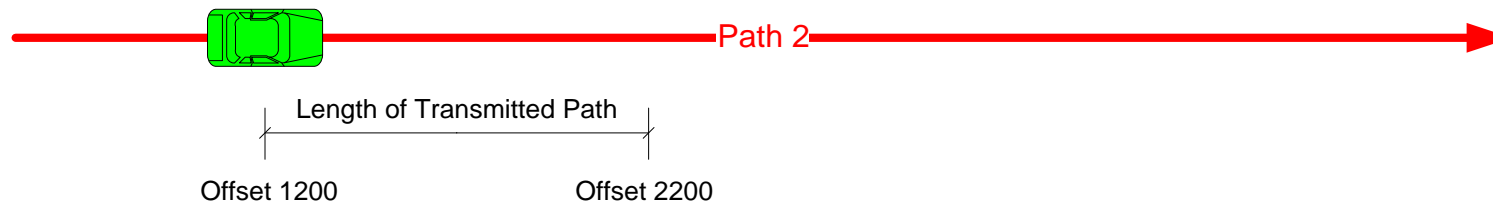
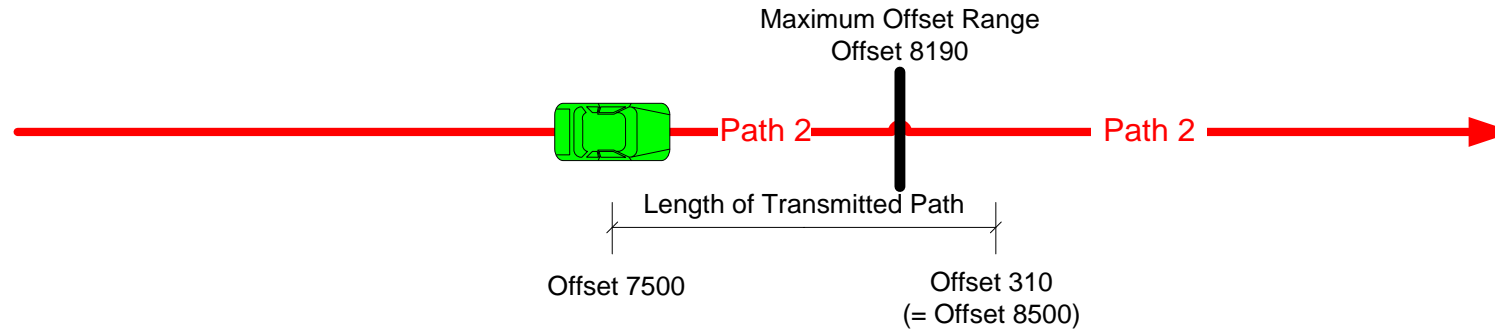


Figure 13: Short ADAS Horizon



**Figure 14: Limited Cyclic Offset**

The whole ADASIS v2 concept is event-based, which means that the Av2HP will not send out new horizon messages until there is a change of attribute or a new event to be announced. As the current length of the transmitted path is marked by the outmost event location, this length can be less than the maximum length of transmitted path.

### 3.4.3 Path Length Limits

The ADASIS v2 concept uses different range values to define the length of the ADASIS horizon. These are defined in this section.

#### Maximum Offset Range:

Allowed number range to define an offset value in each CAN message: 0 – 8190 (13 bit).

#### Maximum Length of Transmitted Path:

This value configures the ADASIS horizon provider and defines, that the provider shall never transmit information farther away from the current vehicle position than "maximum length of transmitted path".

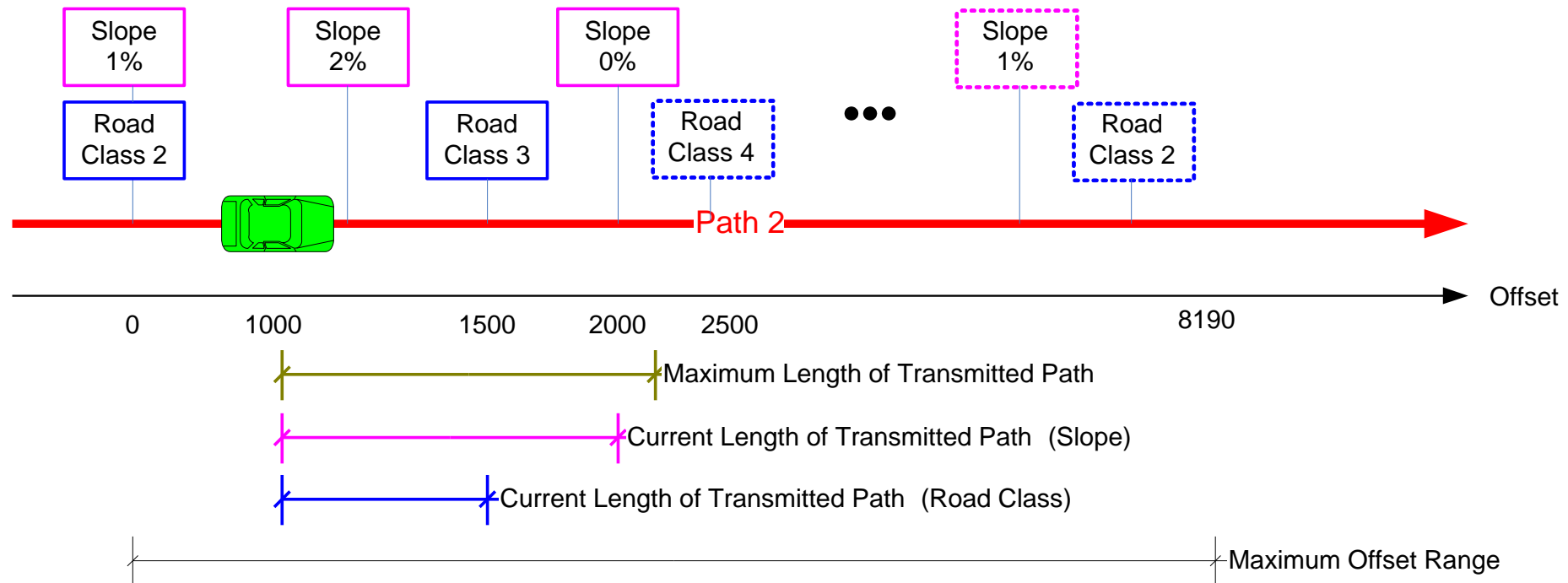
### Current Length of Transmitted Path:

This value is dynamic and changes during driving. The whole ADASIS v2 concept is event-based, which means that if the map attributes do not change, the Av2HP will not send out new horizon messages. Consequently, a minimal "current length of the transmitted path" is not guaranteed and this length can vary.

The "current length of transmitted path" depends on the message type and on the profile type. It is defined by the distance of the current vehicle position to the farthest horizon message offset.

### Example:

- Figure 15 shows an example of the different length values.
- The current vehicle position is at offset value 1000m
- The "maximum offset range" is limited to 8190
- The "maximum length of transmitted path" is configured to 1200m
- The "current length of transmitted path" for the short profile slope is 1000m, because the provider has sent out a new slope information 0% at offset value 2000
- The "current length of transmitted path" for the segment message (containing the road class) is only 500m, because the farthest road class information is at offset 1500 and the next one at offset 2500 is too far away and has not yet been transmitted.



**Figure 15: Definition of Path Lengths**

As the road class example in Figure 15 shows, the "current length of transmitted path" may become rather short if there is no attribute change to transmit. To avoid this, the horizon provider should send intermediate messages after a certain distance, that repeat the information but give more reliability for the ADAS client (see also ADAS v2 Horizon Provider (Av2HP), Av2HP Recommendation #6).

#### **3.4.4 Horizon Trailing Length**

Due to bandwidth restrictions, to message prioritization, and a need to have two profile points for linear interpolatable profile types etc., it is possible and it may be needed that events are transmitted for locations behind the vehicle position. The distance limit of allowed offsets behind the vehicle for which events are accepted is defined by the "Maximum Horizon Trailing Length".

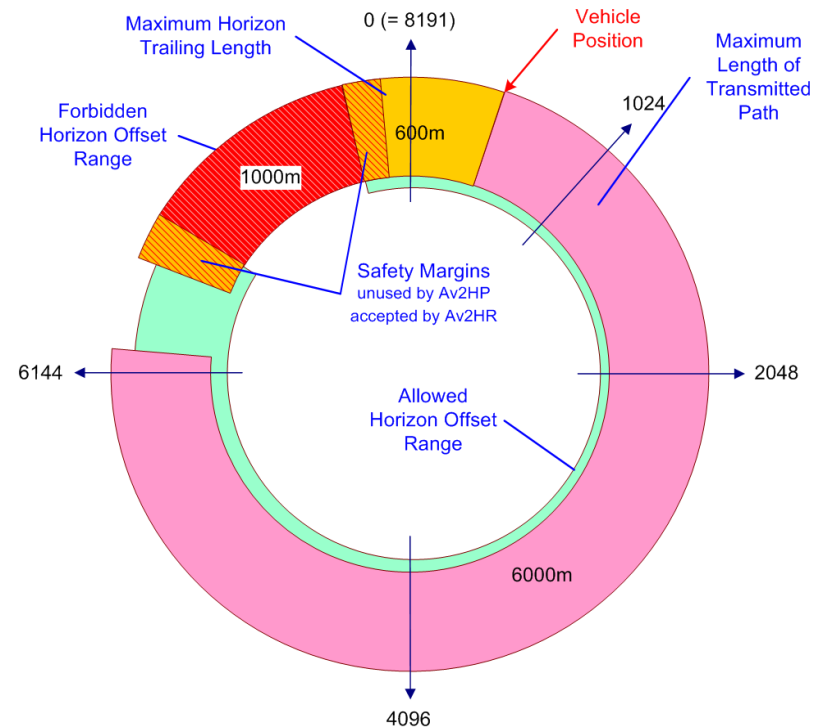
The ADASISv2 Horizon Provider may transmit events for offsets behind the vehicle within this "Maximum Horizon Trailing Length" reduced by a "Safety Margin" to respect that events are not necessarily processed by an ADASISv2 application in the order they are generated while the vehicle position changes.

Additionally the ADASISv2 Horizon Provider shall consider another "Safety Margin" for the "Maximum Length of Transmitted Path" for offsets in front of the vehicle.

In summary these parameters for the ADASISv2 Horizon Provider and all ADASISv2 Horizon Clients define an "Allowed Horizon Offset Range" with respect to the current vehicle position, for which events may be transmitted and accepted. The counterpart of the "Allowed Horizon Offset Range" is the "Forbidden Horizon Offset Range" to clearly distinguish between offsets which are behind the vehicle position and offsets which are in front of the vehicle position at the far end of the horizon.

The different lengths are illustrated in Figure 16.





**Figure 16: Visualisation of the ADASISv2 Horizon Length, Trailing Length and Forbidden Offset Range**

The “Maximum Horizon Trailing Length” and the “Forbidden Horizon Offset Range” limit the “Maximum Allowed Horizon Offset Range” and consequently the “Maximum Length of Transmitted Path”.

The fact that events are sent out in increasing offset order could in theory be used to transmit events at offsets exceeding the "maximum offset range". This is strongly discouraged, as it is prone to various implementation problems, including ambiguities when updating horizon data.

### 3.5 Path Profiles

A **Path Profile** is a property that has a value for any location along a path (e.g., curvature, form of way, number of lanes, speed limit, horizontal geometry). This value may also be <undefined>.

A *Path Profile* is made up of a quantity of “**Profile Spots**” and the specification of a **Profile Interpolation Type**.

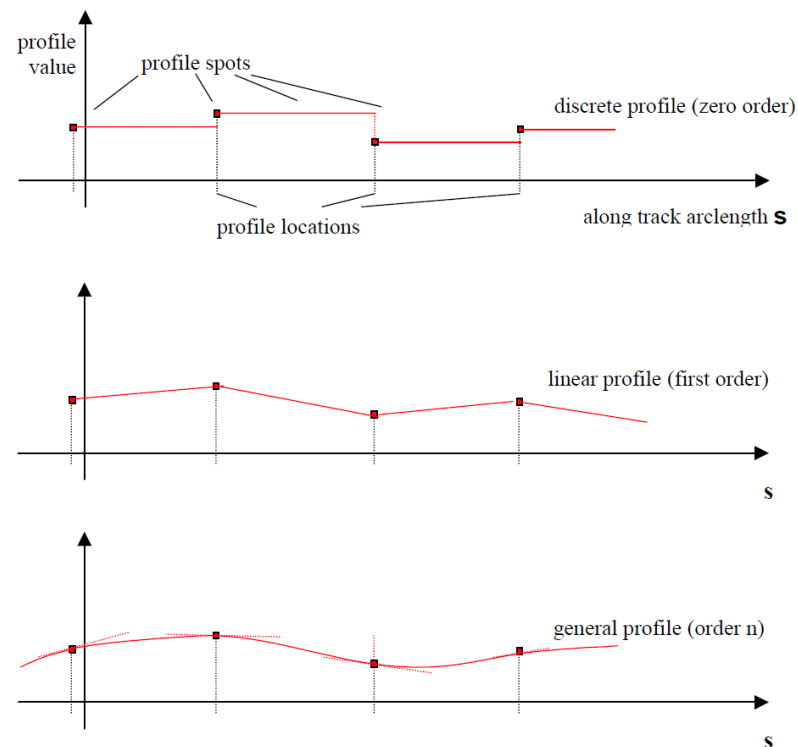


Figure 17: Profiles

As there are a variety of ways to describe a profile as a function of offset along the path, the *Profile Interpolation Type* specifies how the profile value is to be calculated from the *Profile Spots* at intermediate locations using some interpolation scheme (e.g., discrete, step-profile, linear or higher order interpolation ...).

Figure 17 depicts the key definitions involved in path profile representations, together with visualized examples for different *Profile Interpolation Types*.

- **Profile Type** is the property that the profile shall represent.
- **Profile Interpolation Type** is a code specifying how intermediate profile values are to be calculated.  
The following Profile Interpolation Types are identified as follows:
  - Discrete interpolation profile (zero order): The profile is specified by the set of locations of attribute changes together with the new value. Each value is valid until next location.
  - Linear interpolation profile (first order): The profile is specified by a set of locations together with the attribute value at the location, where the value must be linearly interpolated between the given locations. (Note that the polygonal 2D representation of the map geometry is a first order profile, since the geometry between the shape points is calculated by linear interpolation of latitude/longitude values).
  - Higher order interpolation profile (quadratic / cubic polynomial, spline ...): The profile is specified by a set of locations with defined sets of attribute values at the locations. The interpolation algorithm to be used to calculate the profile value between the given locations must be specified. (Note that this may be done by using the extended set of coordinates plus attributes from the two neighbouring points and/or additionally use values from second-degree neighbours; alternatively, *control points* can define additional parameters used in interpolation (see also section 5.7)).
  - Attachment (singularity spots): The attachment is specified by the set of locations of entities with an attribute value at each location. This is in fact a degenerated profile, since there are no entities and therefore no attribute values between the given location spots. A typical attachment is a Traffic Sign.

In ADASIS v2, *Profile Type* implies *Profile Interpolation Type*. For instance, a step-interpolated Slope profile is of a different type than a linear-interpolated Slope profile.

- **Profile Offset** is a position along a path, defined by its distance from the path origin, measured as the arc or poly-line length along the path.

- **Profile Spot** is the numeric description value of a property at a Path Offset (note that this may be a single value  $p_k$ , representing the property value itself at *Path Location k*). Depending on the interpolation type, a *Profile Spot* may also be defined as a vector  $p_k$  containing several values (including for example first or second order derivatives or longitude + latitude coordinates) necessary for smooth curve representation (e.g., for polynomial or spline interpolations). In that case, profile *control points* are used to accommodate additional parameters (see also section 5.7).

For more details, see also [3].

## 4 Messages

ADASIS v2 assumes one-way communication between an ADASIS v2 Horizon Provider and ADASIS v2 Horizon Clients. Six types of messages are defined in the ADASIS v2.

In this section, those messages are ordered by priority. Besides a list and descriptions of message fields, a usable CAN layout for each message is also suggested. Five bits are common to every ADASIS v2 message:

1. Three bits define the message type of this ADASIS v2 message. This field is critical in multiplexing setups (Table 2), in a non-multiplexing CAN environment message type can be deduced from the CAN identifier of message (Table 3) and this field is redundant. The TYPE field shall always occupy the same bits of each CAN frame.
2. Two bits of each message are a cyclic message counter for each specific message type (and each profile type). It will be used to detect missing frames at the client side.

The following message types are defined:

- POSITION message specifies the current position(s) of the vehicle.
- STUB message indicates the start of a new path that has origin at an existing one.
- SEGMENT message specifies the most important attributes of a part of the path.
- PROFILE SHORT message describes attribute of the path whose value can be expressed in 10 bits.
- PROFILE LONG message describes attribute of the path whose value can be expressed in 32 bits.
- META-DATA message contains utility data.

The value definition of each signal in the messages includes two additional special values "unknown" and "signal not available". The "unknown" value should be used by the ADAS horizon provider, if a certain information is not known for one road (e.g. unknown in the map data if a certain road is a tunnel or not). The "signal not available" value (abbreviation "N/A") should be used, if a certain data attribute is completely not available or implemented (e.g. a certain map does not have tunnel information at all or a certain software release has the tunnel information not yet implemented).

#### 4.1 Message Types and CAN frame Multiplexing

In each message, the 3-bit field **Message Type** determines semantics of rest of the message.

<i>Message Type</i>	<i>Message</i>
0	System Specific
1	POSITION
2	SEGMENT
3	STUB
4	PROFILE SHORT
5	PROFILE LONG
6	META-DATA
7	Reserved

**Table 1: Message Type field**

Therefore, the content of the *Message Type* field determines what fields are present in each CAN frame used in the protocol.

The minimum number of CAN identifiers that must be assigned to ADASIS v2 protocol is one.

For instance, the following convention can be used:

<i><b>CAN Identifier</b></i>	<i><b>Content of Message Type field</b></i>	<i><b>Message</b></i>
<b>100</b>	0	System specific
<b>100</b>	1	POSITION
<b>100</b>	2	SEGMENT
<b>100</b>	3	STUB
<b>100</b>	4	PROFILE SHORT
<b>100</b>	5	PROFILE LONG
<b>100</b>	6	META-DATA
<b>n/a</b>	7	Reserved

**Table 2: ADASIS v2 as fully multiplexed CAN protocol**

If multiplexing (polymorphism) of CAN frames is not desirable, different CAN identifiers can be assigned to the different messages. However, the *Message Type* field is still filled with proper values.

<i><b>CAN Identifier</b></i>	<i><b>Content of Message Type field</b></i>	<i><b>Message</b></i>
<b>100</b>	0	System specific
<b>101</b>	1	POSITION
<b>102</b>	2	SEGMENT
<b>103</b>	3	STUB
<b>104</b>	4	PROFILE SHORT
<b>105</b>	5	PROFILE LONG
<b>106</b>	6	META-DATA
<b>n/a</b>	7	Reserved

**Table 3: ADASIS v2 as non-multiplexed CAN protocol**

Most systems support single-level multiplexing in CAN traffic. If no other requirements are set, only the *Message Type* field can be used as the multiplexing field, since each message type has fixed structure.

Strictly speaking, each profile type accommodated in the PROFILE LONG and PROFILE SHORT require distinct CAN identifier, since the value fields are interpreted differently for each profile type. If, in such an environment, strict single-level multiplexing is required, one must reserve at least two CAN identifiers to accommodate ADASIS v2 protocol.



<i>CAN Identifier</i>	<i>Content of Message Type field</i>	<i>Message</i>	<i>Multiplexing field</i>
<b>199</b>	0	System Specific	
<b>200</b>	1	POSITION	MESSAGE TYPE
<b>200</b>	2	SEGMENT	MESSAGE TYPE
<b>200</b>	3	STUB	MESSAGE TYPE
<b>201</b>	4	PROFILE SHORT	MESSAGE TYPE + PROFILE TYPE
<b>201</b>	5	PROFILE LONG	MESSAGE TYPE + PROFILE TYPE
<b>200</b>	6	META-DATA	MESSAGE TYPE
<b>n/a</b>	7	Reserved	

**Table 4: ADASIS v2 as partially multiplexed CAN protocol**

As shown in Table 4, the POSITION, SEGMENT, STUB and META-DATA messages share the same CAN id and multiplexing is accomplished by looking at the *Message Type* field. PROFILE SHORT and PROFILE LONG messages share another CAN identifier. For those two message types, the multiplexing field is a combination of the *Message Type* and the Profile Type fields, which together are conveniently exactly 8 bits long.

If multiplexing (polymorphism) of CAN frames is not desirable, different CAN identifiers can be assigned to different messages. However, the *Message Type* field is still filled with proper values. If different profiles are also considered as different messages, the total number of CAN identifiers that needs to be used is 5 (POSITION + SEGMENT + STUB + META-DATA + System Specific) plus the total number of PROFILES, which may be up to 62 (31 SHORT profiles and 31 LONG profiles).

In conclusion, we see that, depending on configuration and requirements, the number of CAN message identifiers that needs to be allocated may vary between 1 (fully multiplexed protocol) to a theoretical maximum of 67 (each message type and each profile has own CAN identifier).

The developer should also take in account that the use of more than one CAN identifier may influence the order of receiving frames at the client side.

## 4.2 Intel CAN message layouts

CAN message layouts illustrated in following tables show the arrangement of ADASIS signals using the Motorola format (big endian). Due to its common adoption within the automotive industry, the Motorola format has been selected as the preferred CAN message format. If a particular implementation requires a layout based on the Intel format (little endian), signal start-bits must be adapted accordingly. One simple approach to accomplishing this is by shifting the byte ordering. Assuming that the least-significant bit is at position zero, this may be expressed mathematically by the following formula:

$$\text{Startbit}_{\text{Intel}} = \text{Startbit}_{\text{Motorola}} + 56 - 16 * \text{floor}(\text{Startbit}_{\text{Motorola}}/8)$$

The system designer must realize that these two formats are not compatible with each other. In order to guarantee consistency, the provider, and all reconstructors connected to the same CAN bus must use the same format.

## 4.3 Alternative bus configurations

The ADASISv2 specification defines CAN messages with 8 bytes payload data. The ADASIS forum has decided to design the specification for CAN, because it has the strictest constraints compared to other current bus systems like Flexray. CAN is widely established as basic technology for data communication in the vehicle network. Due to the long change cycles in vehicle technology, CAN will still be the most important data bus in the coming years. Other bus systems like Flexray and MOST are specialized on certain applications and will not replace the complete CAN network in the vehicle.

If the ADASIS messages should be transmitted over other bus systems like Flexray, the 8 bytes payload data will not raise major problems. If the ADASIS messages should be transmitted over TCP/IP, this can be realized by introducing 8-byte binary blocks to send from the provider to the clients.

#### 4.4 POSITION Message

The POSITION message describes current position(s) of vehicle in relation to ADASIS v2 Horizon Paths.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Message Type</b>	3	1		See Table 1
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type: Reconstructor can use this value to detect missing messages
<b>Path Index</b>	6	0..7 8..63		Index of current path: If this number differs from the one transmitted in the previous POSITION message with Position Index = 0, the vehicle has left the original path; Indices 0-7 have special meaning: see Table 7 for details
<b>Offset</b>	13	0..8190 8191	1 meter	Offset of current position from the current path's start point modulo 8191. Value 8191 is invalid.
<b>Position Index</b>	2	0..3		If Av2HP supports positioning alternatives, contains index of this Position Candidate. Candidate 0 is always most probable position candidate, candidate 1 second etc. If Av2HP does not support positioning alternatives, it will be always 0. See also section 6.4.
<b>Position Age</b>	9	0..510 511	5 millisecond	Time difference between the moment the message is sent and the moment the vehicle position is calculated by the positioning sub-system : <ul style="list-style-type: none"> <li>• <b>0...509:</b> 0 ... 2545 ms</li> <li>• <b>510:</b> &gt; 2545 ms</li> <li>• <b>511:</b> N/A</li> </ul>

Field	Length (bits)	Value Range	Units	Description
<b>Speed</b>	9	0..510 511	0.2 m/s	Speed of the vehicle projected to the path. Values are: <ul style="list-style-type: none"> <li>• <b>0:</b> ≤ -12.8 m/s</li> <li>• <b>1...63:</b> -12.6 ...-0.2 m/s</li> <li>• <b>64:</b> Standing still</li> <li>• <b>65...509:</b> +0.2...+89.0 m/s</li> <li>• <b>510:</b> ≥ +89.2 m/s</li> <li>• <b>511:</b> N/A</li> </ul>
<b>Relative Heading</b>	8	0..254 255	360/254 degrees	Heading relative to path: <ul style="list-style-type: none"> <li>• <b>0:</b> In path direction</li> <li>• <b>1:</b> 1.417 degrees right</li> <li>• <b>2..126:</b> Right from path direction</li> <li>• <b>126:</b> 178.583 degrees right</li> <li>• <b>127:</b> Opposite direction</li> <li>• <b>128:</b> 178.583 degrees left (181.417 degrees right)</li> <li>• <b>129..252:</b> Left from path direction</li> <li>• <b>253:</b> 1.417 degrees left (358.583 degrees right)</li> <li>• <b>254:</b> Unknown</li> <li>• <b>255:</b> N/A</li> </ul>
<b>Position Probability</b>	5	0..30 31	100/30 percent.	Position Probability is calculated by Positioning module of ADASIS v2 Horizon Provider. Different Av2HPs will use different algorithms to calculate this value.  Value 0 indicates <i>unknown</i> ; Value 31 means <i>N/A</i> .

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Position Confidence</b>	3	0..6 7		Position confidence: this field contains Av2HP implementation specific values. Use 0 for highest confidence, 6 for lowest confidence. Value 7 means N/A
<b>Current Lane</b>	3	0..6 7		Current lane; for details, see Table 8: Current Lane values.
<b>Reserved</b>	1			

**Table 5: POSITION message**



Table 6: CAN layout of POSITION message

<i>Value</i>	<i>Message</i>	<i>Description</i>
<b>0</b>		Unknown
<b>1</b>	POSITION	Position not in Digitized Area
<b>2</b>	POSITION	Position not on-road
<b>3</b>	POSITION	Av2HP Vehicle Positioning sub-system not calibrated
<b>4</b>	SEGMENT	Messages with this path index refer only to the current vehicle position segment (for simple applications that do not need a prediction horizon - see description in section 5.8 "ADASIS Mini")
<b>5</b>	STUB	(= Sub-Path index field of STUB message) -- indicates the existence of a crossroad branch that will not further be appended by path data - see section 5.2.
<b>6</b>	STUB	(= Sub-Path index field of STUB message) -- Indicates that the STUB message is used to transmit the HEADING CHANGE and RELATIVE PROBABILITY of the main path at an intersection (see also section 5.2)
<b>7</b>		Reserved

**Table 7: Path Indices with Special Meaning**

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>0</b>	Unknown
<b>1</b>	Emergency lane
<b>2</b>	Single-lane road
<b>3</b>	Left-most lane
<b>4</b>	Right-most lane
<b>5</b>	One of middle lanes on road with three or more lanes
<b>6</b>	Reserved
<b>7</b>	N/A

**Table 8: Current Lane values**

## 4.5 SEGMENT Message

This message specifies attributes of the part of the road ahead. The SEGMENT message packs a number of the most important PROFILES in one CAN frame.



<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Message Type</b>	3	2		See Table 1
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type: Reconstructor can use this value to detect missing messages
<b>Retransmission</b>	1	0 (=false) 1 (=true)		If <i>true</i> , this message was sent already at least once before
<b>Path Index</b>	6	4 8..63		Value is Index of path of which this segment is part
<b>Offset</b>	13	0..8190 8191	1 meter	Offset of this segment from start position of the path modulo 8191; Value 8191 is invalid
<b>Update</b>	1	0 (=false) 1 (=true)		Value <i>true</i> indicates that this message contains updated information of already transmitted SEGMENT. See section 5.6 for details.
<b>Functional Road Class</b>	3	0, 1..6 7		Functional Road Class indicates relative importance of the road in the routing network and lower values indicate higher priority. This information is usually map-provider specific. Regular values are 1-6; value 0 is unknown, value 7 indicates N/A.
<b>Form of Way</b>	4	0..14 15		Values are described in Table 11; see also [8] (page K-8)

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Effective Speed Limit</b>	5	0..30 31	km/h or mph	<p>Current Speed Limit: May be implicit (for instance, in cities), or explicit. Current time of day, day of week and weather conditions can taken into account by Horizon Provider in determining effective speed limit.</p> <p>May be expressed as km/h or mph, see section 4.9 META-DATA Message.</p> <p>Note that this might be a rounded value. E.g. if there is a speed limit of 19 mph, it will be rounded to 20 mph, because 19 is not directly represented. Exact values should be transmitted via PROFILE messages. Nevertheless, in most cases this will be the exact value.</p> <p>Coding of this value described in Table 12.</p>
<b>Effective Speed Limit Type</b>	3	0..6 7		Describes what kind of Speed Limit is in effect. See Table 13: Speed Limit Type.
<b>Number of lanes in driving direction</b>	3	0..6 7		<p>Number of lanes in driving direction: also implies driving direction.</p> <p>0 denotes one-direction road driven in wrong direction, 1-5 are verbatim values, and 6 indicate 6 or more lanes. Value 7 means <i>N/A</i>.</p>
<b>Number of lanes in opposite direction</b>	2	0..2 3		<p>Number of lanes in opposite direction:</p> <p>0 denotes one-direction road, 1 for one lane only; 2 indicates 2 or more lanes. Value 3 means <i>N/A</i>.</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b><i>Tunnel</i></b>	2	0..2 3		0: Segment is not part of a tunnel 1: Segment is part of a tunnel 2: Unknown 3: N/A.
<b><i>Bridge</i></b>	2	0..2 3		0: Segment is not part of a bridge. 1: Segment is part of a bridge. 2: Unknown 3: N/A
<b><i>Divided Road</i></b>	2	0..2 3		0: Segment is not part of a divided road/dual carriageway. 1: Segment is part of a divided road/dual carriageway. 2: Unknown 3: N/A
<b><i>Built-up Area</i></b>	2	0..2 3		0: Segment is not part of a built-up area. 1: Segment is part of a built-up area. 2: Unknown 3: N/A

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Complex Intersection</b>	2	0..2 3		<p>0: Segment is not part of complex intersection.            1: Segment is part of a complex intersection.            2: Unknown            3: N/A</p> <p>Complex intersection is defined by the GDF level 2 complex feature "Intersection" containing more than one junction and/or road element. "Complex intersection" shall be set when the vehicle drives on road elements with source node and target node belonging to the same complex feature ("plural junction" with "intersection internal" road elements)</p>
<b>Relative Probability</b>	5	0..30 31	100/30 percent	<p>Relative probability of this segment in relation to the previous segment on the same path; for the first segment on the path, this probability is equal as relative probability of the STUB, while for all subsequent segments till next crossing it will be 100%.</p> <p>Value 0 indicates that driving along this segment of the path is not allowed. Value 31 means unknown or N/A.</p>
<b>Part of Calculated Route</b>	2	0..2		<p>0: Segment is not part of Calculated Route            1: Segment is part of Calculated Route            2: Unknown            3: N/A</p>
<b>Reserved</b>	1			

**Table 9: SEGMENT message**

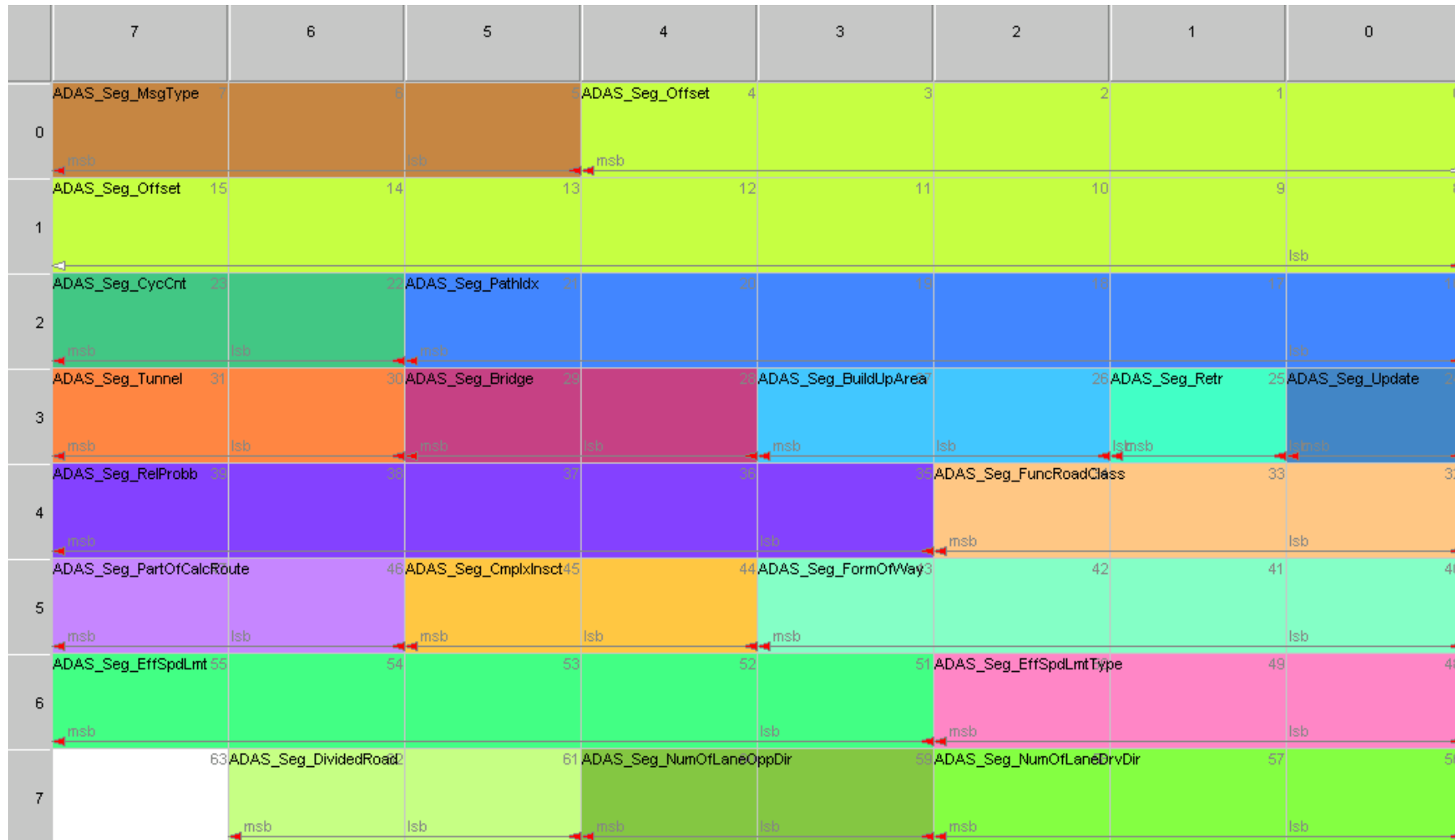


Table 10: CAN layout of SEGMENT message

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>0</b>	Unknown
<b>1</b>	Freeway or Controlled Access road that is not a slip road/ramp ("Controlled Access" means that the road looks like a freeway (grade separated crossings, physical center divider, motor vehicle only), not necessarily that the road is legally defined as a freeway. Note that different map makers may use a different wording for this signal.)
<b>2</b>	Multiple Carriageway or Multiply Digitized Road
<b>3</b>	Single Carriageway (default)
<b>4</b>	Roundabout Circle
<b>5</b>	Traffic Square/Special Traffic Figure
<b>6</b>	Reserved
<b>7</b>	Reserved
<b>8</b>	Parallel Road (as special type of a slip road/ramp)
<b>9</b>	Slip Road/Ramp on a Freeway or Controlled Access road
<b>10</b>	Slip Road/Ramp (not on a Freeway or Controlled Access road)
<b>11</b>	Service Road or Frontage Road
<b>12</b>	Entrance to or exit of a Car Park

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>13</b>	Entrance to or exit to Service
<b>14</b>	Pedestrian Zone
<b>15</b>	N/A

**Table 11: Form of Way (4-bit)**

<i>Value</i>	<i>Description</i>	<i>Value</i>	<i>Description</i>	<i>Value</i>	<i>Description</i>	<i>Value</i>	<i>Description</i>
<b>0</b>	Unknown (including variable speed sign controlled)	<b>8</b>	35 km/h [mph]	<b>17</b>	80 km/h [mph]	<b>26</b>	130 km/h [mph]
<b>1</b>	≤ 5 km/h [mph]	<b>9</b>	40 km/h [mph]	<b>18</b>	85 km/h [mph]	<b>27</b>	140 km/h [mph]
<b>2</b>	7 km/h [mph]	<b>10</b>	45 km/h [mph]	<b>19</b>	90 km/h [mph]	<b>28</b>	150 km/h [mph]
<b>3</b>	10 km/h [mph]	<b>11</b>	50 km/h [mph]	<b>20</b>	95 km/h [mph]	<b>29</b>	160 km/h [mph]
<b>4</b>	15 km/h [mph]	<b>12</b>	55 km/h [mph]	<b>21</b>	100 km/h [mph]	<b>30</b>	Unlimited
<b>5</b>	20 km/h [mph]	<b>13</b>	60 km/h [mph]	<b>22</b>	105 km/h [mph]	<b>31</b>	Invalid
<b>6</b>	25 km/h [mph]	<b>14</b>	65 km/h [mph]	<b>23</b>	110 km/h [mph]		
<b>7</b>	30 km/h [mph]	<b>15</b>	70 km/h [mph]	<b>24</b>	115 km/h [mph]		
		<b>16</b>	75 km/h [mph]	<b>25</b>	120 km/h [mph]		

**Table 12: Effective Speed Limit (5-bit)**



<i><b>Value</b></i>	<i><b>Description</b></i>
<b>0</b>	Implicit (for instance, default speed limit in the cities)
<b>1</b>	Explicit – on traffic sign
<b>2</b>	Explicit – by night
<b>3</b>	Explicit – by day
<b>4</b>	Explicit – time of day
<b>5</b>	Explicit – rain
<b>6</b>	Explicit – snow
<b>7</b>	Unknown

**Table 13: Speed Limit Type (3-bit)**

## 4.6 STUB message

The STUB message describes crossing on the path. Each stub is the origin for a new path.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Message Type</b>	3	3		See Table 1.
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type: Reconstructor can use this value to detect missing messages.
<b>Retransmission</b>	1	0 (=false) 1 (=true)		If <i>true</i> , this message has been sent at least once before.
<b>Path Index</b>	6	0 8..63		Index of parent path; zero indicates this STUB starts first path in the Horizon.
<b>Offset</b>	13	0..8190 8191	1 meter	Offset of the position of the crossing from the start path position modulo 8191. Offset 8191 is invalid.
<b>Update</b>	1	0 (=false) 1 (=true)		Value <i>true</i> indicates that this message contains updated information of already transmitted STUB. See section 5.6 for details.
<b>Sub-Path Index</b>	6	5,6,8..63		This is Index of new path. See also section 5.2.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Turn Angle</b>	8	0..254 255	360/254 degrees	<p>Angle of new path relative to parent path:</p> <ul style="list-style-type: none"> <li>• <b>0:</b> In path direction</li> <li>• <b>1:</b> 1.417 degrees right</li> <li>• <b>2..125:</b> Right from path direction</li> <li>• <b>126:</b> 178.583 degrees right</li> <li>• <b>127:</b> Opposite direction</li> <li>• <b>128:</b> 178.583 degrees left (181.417 degrees right)</li> <li>• <b>129..252:</b> Left from path direction</li> <li>• <b>253:</b> 1.417 degrees left (358.583 degrees right)</li> <li>• <b>254:</b> Unknown</li> <li>• <b>255:</b> N/A</li> </ul> <p>In general, this is angle between extension of incoming link and outgoing link. Exact method of calculation is not defined.</p>
<b>Relative Probability</b>	5	0..30 31	100/30 percent	<p>Relative probability of the new path on the crossing; sum of all relative stub probabilities (including continuation stub with sub-path index of 6) is 100 percent.</p> <p>Value 0 (= 0%) indicates that turn is not allowed. Value 30 (= 100%) indicates that this is the only possible turn. Value 31 means unknown or N/A.</p>
<b>Functional Road Class</b>	3	0, 1..6 7		<p>Functional Road Class of new path indicates relative importance of the road in the routing network and lower values indicate higher priority. This information is usually map-provider specific. Regular values are 1-6; value 0 is unknown, value 7 indicates N/A.</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Form of Way</b>	4	0..14 15		Form of Way of new path; values are described in Table 11; see also [8] (page K-8)
<b>Number of lanes in driving direction</b>	3	0..6 7		Number of lanes in driving direction of new path: Also implies driving direction. 0 denotes one-direction road driven in wrong direction, 1-5 are verbatim values, and 6 indicate 6 or more lanes. Value 7 means <i>N/A</i> .
<b>Number of lanes in opposite direction</b>	2	0..2 3		Number of lanes in opposite direction on new path:  0 denotes one-direction road, 1 for one lane only; 2 indicates two or more lanes. Value 3 means <i>N/A</i> .

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Complex Intersection</b>	2	0..2 3		<p>0: New path is not part of complex intersection.  1: New path is part of a complex intersection.  2: Unknown  3: N/A</p> <p>Complex intersection is defined by the GDF level 2 complex feature "Intersection" containing more than one junction and/or road element. "Complex intersection" shall be set when the vehicle drives on road elements with source node and target node belonging to the same complex feature ("plural junction" with "intersection internal" road elements)</p>
<b>Right of Way</b>	2	0..2 3		<p>0: Driving towards this stub has priority over all conflicting vehicle movements  1: Driving towards this stub might require yielding to conflicting vehicle movements  2: Unknown  3: N/A</p>
<b>Part of Calculated Route</b>	2	0..2		<p>0: New path from this point on is not part of Calculated Route  1: New path from this point on is part of Calculated Route  2: Unknown  3: N/A</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b><i>Last Stub at Offset</i></b>	1	0 (=false) 1 (=true)		Value <i>true</i> indicates that this STUB message is last one that refers to given parent path and offset; value <i>false</i> says that more STUB messages for this offset on parent path will be sent
<b><i>Reserved</i></b>	0			

Table 14: STUB message

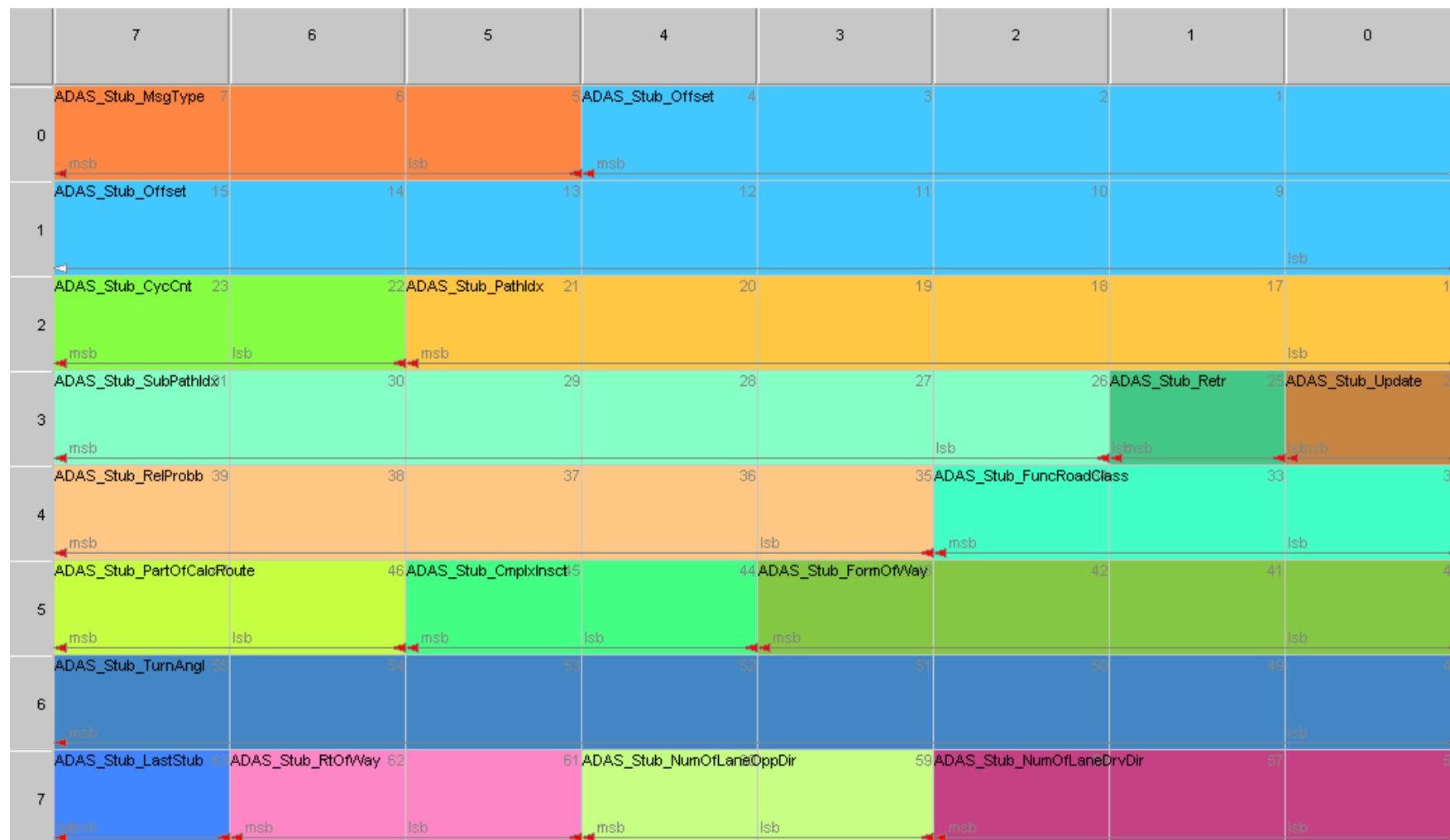


Table 15: CAN layout of STUB message

## 4.7 PROFILE SHORT Message

This message contains two 10-bit *Profile Spots* that are up to 1022 meters away. Interpretation of spot values as well as profile interpolation between spot values depends on the actual *Profile Type*. For all parts of a path where a profile is not defined or available, a default value is assumed (also defined by the *Profile Type*).

Field	Length (bits)	Value Range	Units	Description
<b>Message Type</b>	3	4		See Table 1
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type and profile type: Reconstructor can use this value to detect missing messages
<b>Retransmission</b>	1	0 (=false) 1 (=true)		If <i>true</i> , this message was sent already at least once before
<b>Path Index</b>	6	8..63		Index of the path where this profile(s) is located
<b>Offset</b>	13	0..8190 8191	1 meter	Offset of position of the first profile spot from start position modulo 8191. Offset 8191 is invalid
<b>Update</b>	1	0 (=false) 1 (=true)		Value <i>true</i> indicates that this message contains updated information of already transmitted PROFILE SHORT. See section 5.6 for details.
<b>Profile Type</b>	5	1..31 0		See Table 17: Standard PROFILE SHORT Types and section 5.7
<b>Control Point</b>	1	0 (=false) 1 (=true)		If <i>true</i> , values in this message specifies additional control points required for the interpolation of this profile; if <i>false</i> , actual profile values are specified. See also section 5.7.



<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Value 0</b>	10	0..1022 1023		Profile value at <i>Offset</i> ; a special value (usually 1023) signals an invalid, unknown or N/A value.
<b>Distance 1</b>	10	0..1022 1023	1 meter	Distance from first profile spot to the second one. If 0, no other profile values are defined in this message; offset 1023 is invalid.
<b>Value 1</b>	10	0..1022 1023		Profile value at <i>Offset+Distance1</i> ; a special value (usually 1023) signals an invalid, unknown or N/A value.
<b>Accuracy</b>	2	0..3		Accuracy of profile value is implementation specific.  0 means highest accuracy. 1 means medium accuracy. 2 means lowest accuracy. 3 indicates that accuracy is unknown.
<b>Reserved</b>	0			There are no unused bits.

**Table 16: PROFILE SHORT message**

<i>PROFILE Type</i>	<i>Description</i>	<i>Size (bits)</i>	<i>Multiplier</i>	<i>Offset</i>	<i>Value Interpretation</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>0</b>	N/A						
<b>1</b>	Curvature	10			See Table 31	Linear	1023 (unknown)
<b>2</b>	Route Number Types	10			<p>Bit-coded field; if bit is set, the corresponding Route number is assigned to the road (see section 10.2):</p> <p>Bit 0: Street with no Route number            Bit 1: Route number type 1            Bit 2: Route number type 2            Bit 3: Route number type 3            Bit 4: Route number type 4            Bit 5: Route number type 5            Bit 6: Route number type 6            Bit 7: Reserved (set to 0)            Bit 8: Reserved (set to 0)            Bit 9: Reserved (set to 0)</p> <p>See also [8] and Table 32</p>	Step	0 (N/A)
<b>3</b>	Slope	10	0.1	-51.1	Slope in percent.	Step	1023 (unknown)
<b>4</b>	Slope	10	0.1	-51.1	Slope in percent.	Linear	1023 (unknown)

<i>PROFILE Type</i>	<i>Description</i>	<i>Size (bits)</i>	<i>Multiplier</i>	<i>Offset</i>	<i>Value Interpretation</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>5</b>	Road accessibility	10			<p>Bit-coded field; if bit is set, road is accessible for particular class of actors:</p> <p>Bit 0: Passenger Cars            Bit 1: Pedestrians            Bit 2: Bus            Bit 3: Delivery            Bit 4: Emergency            Bit 5: Taxi            Bit 6: Through Traffic            Bit 7: Trucks            Bit 8: Reserved (set to 0)            Bit 9: Reserved (set to 0)</p>	Step	0 (no access)
<b>6</b>	Road condition	10			<p>Low 4 bits (0-3) are coded as follows:</p> <p>1xxx: condition good            0xxx: condition poor            x000: rigid (paved)            x100: flexible (paved)            x010: blocks (unpaved)            x110: gravel (unpaved)            x001: dirt (unpaved)            1111: unknown</p> <p>Upper 6 bits (9-4) are reserved for future use (set to 111111).</p>	Step	1023 (unknown)

<b>PROFILE Type</b>	<b>Description</b>	<b>Size (bits)</b>	<b>Multiplier</b>	<b>Offset</b>	<b>Value Interpretation</b>	<b>Interpolation Type</b>	<b>Default Value</b>
<b>7</b>	Variable Speed Sign Position	10			Bit Coded Field, see Table 18	Spot (Attachment)	0 (unknown)
<b>8</b>	Heading Change	10	360/254 degrees		Heading change relative to path: <ul style="list-style-type: none"> <li>• <b>0:</b> In path direction</li> <li>• <b>1:</b> 1.417 degrees right</li> <li>• <b>2..126:</b> Right from path direction</li> <li>• <b>126:</b> 178.583 degrees right</li> <li>• <b>127:</b> Opposite direction</li> <li>• <b>128:</b> 178.583 degrees left (181.417 degrees right)</li> <li>• <b>129..252:</b> Left from path direction</li> <li>• <b>253:</b> 1.417 degrees left (358.583 degrees right)</li> <li>• <b>254:</b> Unknown</li> <li>• <b>≥ 255:</b> N/A</li> </ul> Exact method of calculation of this value is implementation specific. Units are chosen to be the same as <i>Relative Heading</i> field of POSITION message and <i>Turn Angle</i> of STUB message.	Spot (Attachment)	254 (unknown)
<b>9-15</b>	Reserved for standard types						
<b>16-31</b>	Reserved for system specific types						

**Table 17: Standard PROFILE SHORT Types**

<i>Bit</i>	<i>Description</i>
<b>0</b>	Position of Variable Speed Sign is not known/defined
<b>1</b>	Variable Speed Sign is located left of the road
<b>2</b>	Variable Speed Sign is located right of the road
<b>3</b>	Variable Speed Sign is located above road, but lane is unknown/undefined
<b>4</b>	Variable Speed Sign is located above Emergency Lane
<b>5</b>	Variable Speed Sign is located above 1 <sup>st</sup> lane from the right (left, in left-driving countries)
<b>6</b>	Variable Speed Sign is located above 2 <sup>nd</sup> lane from the right (left, in left-driving countries)
<b>7</b>	Variable Speed Sign is located above 3 <sup>rd</sup> lane from the right (left, in left-driving countries)
<b>8</b>	Variable Speed Sign is located above 4 <sup>th</sup> lane from the right (left, in left-driving countries)
<b>9</b>	Variable Speed Sign is located above 5 <sup>th</sup> lane from the right (left, in left-driving countries)

**Table 18: Variable Speed Sign Positions**

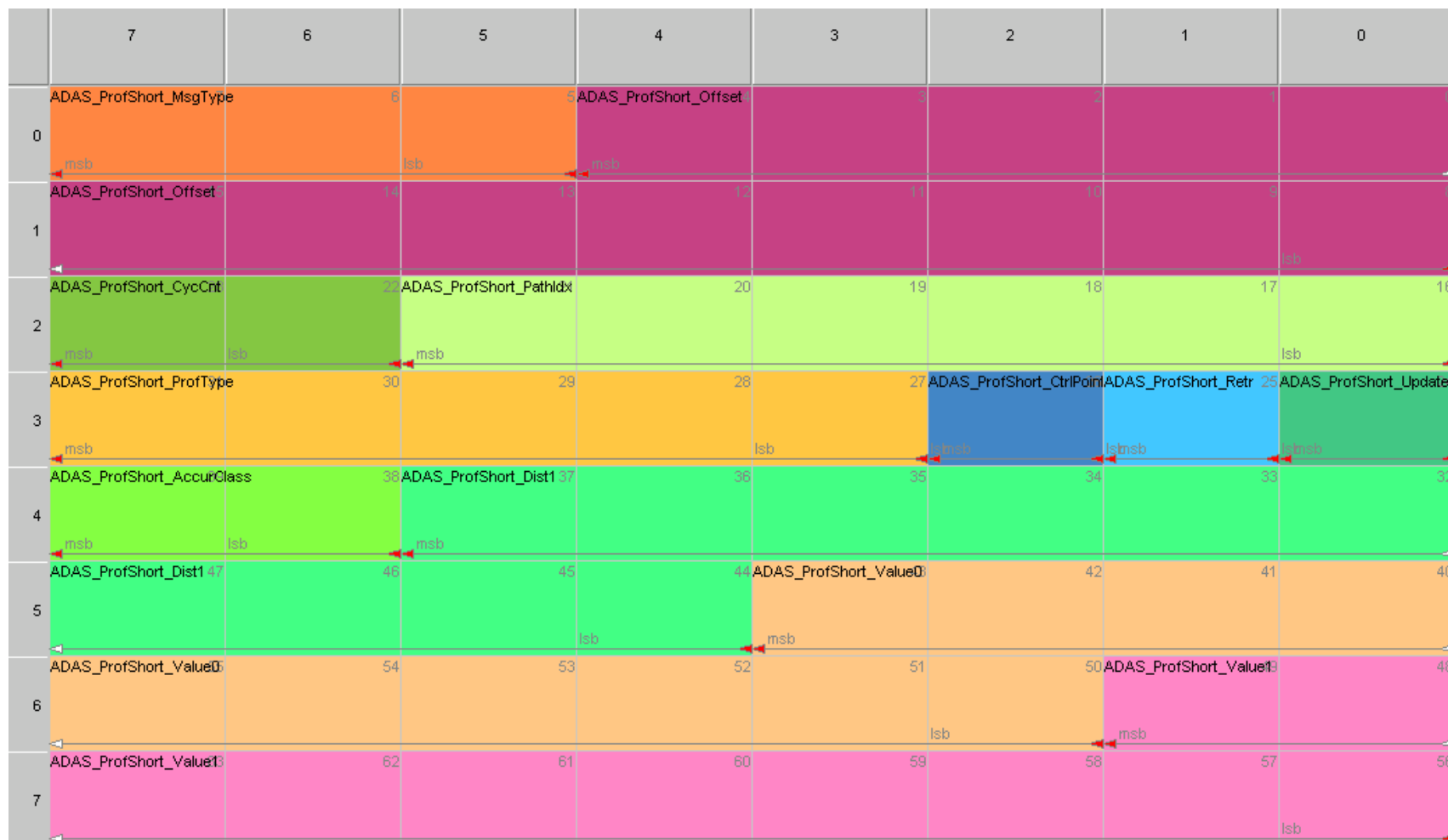


Table 19: CAN layout of PROFILE SHORT

## 4.8 PROFILE LONG Message

This message contains one 32-bit *Profile Spot*. Interpretation of spot values as well as profile interpolation between spot values depends on the actual *Profile Type*. For all parts of a path where the profile is not defined or available, a default value is assumed (also defined by the *Profile Type*).

Field	Length (bits)	Value Range	Units	Description
<b>Message Type</b>	3	5		See Table 1.
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type and profile type: Reconstructor can use this value to detect missing messages.
<b>Retransmission</b>	1	0 (=false) 1 (=true)		If <i>true</i> , this message was sent already at least once before.
<b>Path Index</b>	6	8..63		Index of the path where this profile(s) is located
<b>Offset</b>	13	0..8190 8191	1 meter	Offset of position of the first profile spot from start position modulo 8191. Offset 8191 is invalid.
<b>Update</b>	1	0 (=false) 1 (=true)		Value <i>true</i> indicates that this message contains updated information of already transmitted PROFILE LONG. See section 5.6 for details.
<b>Profile Type</b>	5	1..31 0		See Table 22: Standard PROFILE LONG Types and section 5.7.
<b>Control Point</b>	1	0 (=false) 1 (=true)		If <i>true</i> , values in this message specifies additional control points required for the interpolation of this profile; If <i>false</i> , actual profile values are specified. See also section 5.7.
<b>Value</b>	32	$0..2^{32}-2$ $2^{32}-1$		Profile value at <i>Offset</i> , a special value (usually $2^{32}-1$ ) signals an invalid, unknown or N/A value.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Reserved</b>	0			There are no unused bits.

**Table 20: PROFILE LONG message**



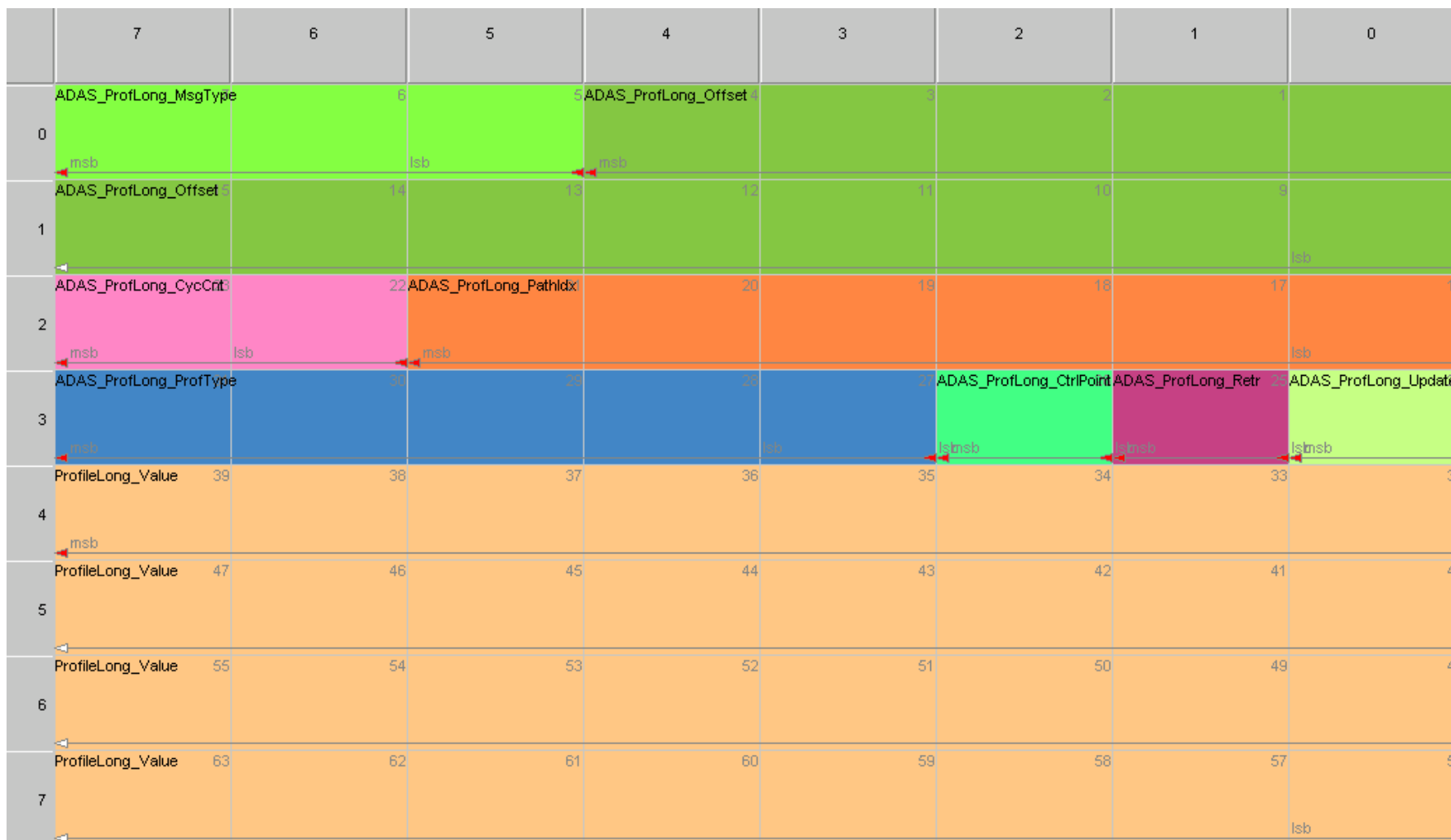


Table 21: CAN layout of PROFILE LONG message

<i>PROFILE Type</i>	<i>Description</i>	<i>Size</i>	<i>Multiplier</i>	<i>Offset</i>	<i>Value Interpretation</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>0</b>	N/A						
<b>1</b>	Longitude	32 bit	0.0000001	-180	Longitude in degrees	Linear	$2^{32}-1$ (N/A)
<b>2</b>	Latitude	32 bit	0.0000001	-90	Latitude in degrees	Linear	$2^{32}-1$ (N/A)
<b>3</b>	Altitude	32 bit	0.01	-1000	Altitude in meters above WGS84 reference ellipsoid  Allowed range: -1000m to +10000m	Linear	$2^{32}-1$ (N/A)
<b>4</b>	(Bézier) Control Point Longitude	32 bit	0.0000001	-180	Longitude in degrees; see also section 10.6.	Linear	$2^{32}-1$ (N/A)
<b>5</b>	(Bézier) Control Point Latitude	32 bit	0.0000001	-90	Latitude in degrees; see also section 10.6.	Linear	$2^{32}-1$ (N/A)
<b>6</b>	(Bézier) Control Point Altitude	32 bit	0.01	-1000	Altitude in meters above WGS84 reference ellipsoid; see also section 10.6.  Allowed range: -1000m to +10000m	Linear	$2^{32}-1$ (N/A)
<b>7</b>	Link Identifier	32 bit			Link Identifier; see also section 10.7.	Step	$2^{32}-1$ (N/A)

<i>PROFILE Type</i>	<i>Description</i>	<i>Size</i>	<i>Multiplier</i>	<i>Offset</i>	<i>Value Interpretation</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>8</b>	Traffic Sign	32 bit			Traffic Sign (see section 10.8).	Point	$2^{32} - 1$ (N/A)
<b>9</b>	Truck Speed Limits	32 bit			Truck Speed Limit; see detailed specification in section 10.9	Step	$2^{32} - 1$ (N/A)
<b>10</b>	Extended Lane	32 bit			Detailed information about lane connectivity, lane marking and lane type; see detailed specification in section 10.10		
<b>11-15</b>	Reserved for standard types						
<b>16-31</b>	Reserved for system specific types						

**Table 22: Standard PROFILE LONG Types**

## 4.9 META-DATA Message

This message is sent with low frequency and it describes basic characteristics of the system or semi-permanent attributes of current position.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Message Type</b>	3	6		See Table 1.
<b>Cyclic Counter</b>	2	0..3		Cyclic counter of this message type: Reconstructor can use this value to detect missing messages.
<b>Country Code</b>	10	0..1023		ISO 3166-1 Release 2007-03 country code of current position; 0 stands for <i>unknown</i> . Valid values are those specified by ISO 3166-1 numeric [14].
<b>Region Code</b>	15	0..32766 32767		For a given <i>Country Code</i> , specifies region of a country where vehicle is currently positioned.  0 stands for <i>unknown</i> , value 32767 means N/A  All other values are based on ISO 3166-2 [15] with the code mapping scheme in section 10.4.
<b>Driving Side</b>	1	0, 1		0 stands for <i>Driving Side Left</i> . 1 stands for <i>Driving Side Right</i> . Default is 1.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Speed Units</b>	1	0, 1		1 stands for <i>miles per hour (mph)</i> . 0 stands for <i>kilometers/hour (km/h)</i> . Default is 0.
<b>Major Protocol Version</b>	2	0..3		Contains Major protocol version; See section 2.1 for details.
<b>Minor Protocol Version</b>	4	0..15		Contains Minor protocol version; See section 2.1 for details.
<b>Minor Protocol Sub-Version</b>	3	0..7		Contains Minor protocol sub-version; See section 2.1 for details.
<b>Hardware Version</b>	9	0..511		This field is ADAS Horizon Provider specific and expresses the hardware and/or software version of the provider depending of the company specific numbering scheme. 0 stands for <i>unknown</i> .

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Map Provider</b>	3	0..6 7		<p>Map Provider of current area (map tile, region, country ...).</p> <ul style="list-style-type: none"> <li>• <b>0:</b> Unknown,</li> <li>• <b>1:</b> Nokia HERE (NAVTEQ),</li> <li>• <b>2:</b> TomTom (Tele Atlas),</li> <li>• <b>3:</b> Zenrin,</li> <li>• <b>4:</b> IPC,</li> <li>• <b>5:</b> NavInfo,</li> <li>• <b>6:</b> Other,</li> <li>• <b>7:</b> N/A.</li> </ul> <p>Please note if the whole map may be partitioned into parts that can be released on different times or even by different providers. In that case, value if this field may vary between messages.</p>
<b>Map Version (Year)</b>	6	0..62 63	Year-2000 mode 63	<p>Content of this field is Year of release of map of current area. For 2007, this field will contain value 7.</p> <p>Please note if the whole map may be partitioned into parts that can be released on different times or even by different providers. In that case, value of this field may vary between messages. 63 stands for N/A</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Map Version (Year Quarter)</b>	2	0..3	Quarter-1	<p>Content of this field is Year quarter of release of map of current area.</p> <p>For example, NAVTEQ Q4/2007 map will be described as</p> <p><i>MAP PROVIDER = 1</i>  <i>MAP VERSION (Y) = 7</i>  <i>MAP VERSION (Q) = 3</i></p> <p>Please note if the whole map may be partitioned into parts that can be released on different times or even by different providers. In that case, value if this field may vary between messages.</p>
<b>Reserved</b>	3			

**Table 23: META-DATA message**

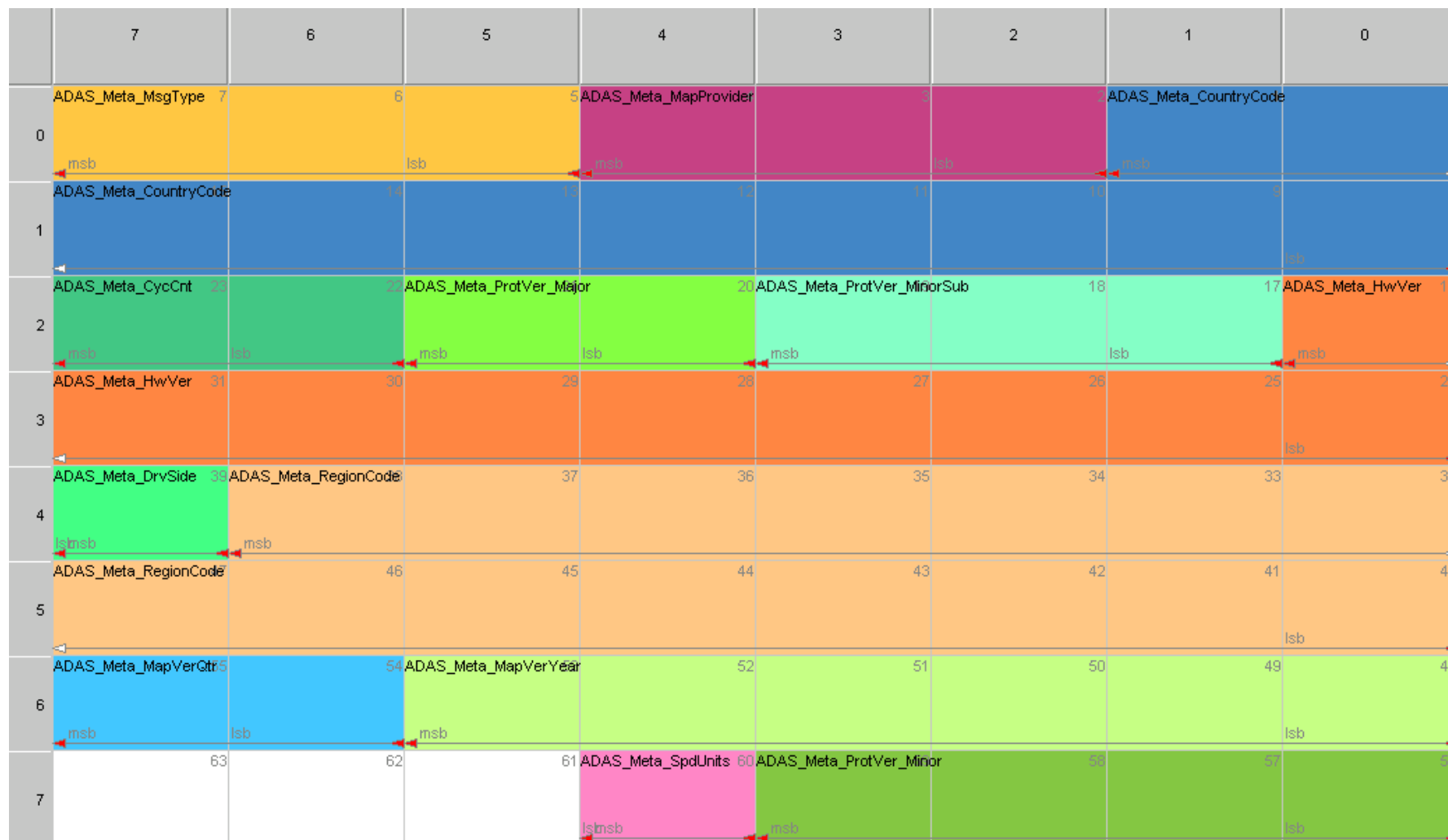


Table 24: CAN layout of META-DATA message



#### 4.10 SYSTEM SPECIFIC Message

This message allows the ADASIS provider to send additional system specific messages.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Message Type</b>	3	0		See Table 1.
<b>System Specific</b>	...	...	...	...

This message shall not be used to transmit additional map information. If additional map attributes are needed, it is required to use the system specific profile types in the PROFILE\_SHORT or PROFILE\_LONG message.

However, this message with message type 0 is introduced because one ADASIS partner uses it to transmit check-sums for the ADASIS content. The need for check-sums is not seen by all ADASIS partners, therefore this is defined as system specific message.

## 5 Use of ADASIS v2 Protocol

### 5.1 Required messages

Which messages the Av2HP will actually send in a given configuration is up to the application set-up.

- POSITION message is the only one that is actually necessary to be sent in any configuration, since it defines vehicle position on the ADASIS v2 Horizon. In ADASIS v2 Mini (section 5.8), only required message is SEGMENT.
- STUB message has several roles in the protocol:
  1. It defines the parent/child relationship between paths;
  2. It marks the start of a new path;
  3. It describes attributes of the first part of the path (number of lanes, for instance)
  4. A collection of several STUBs at the same path and same offset describes an intersection.

If none of the information above is required by any client application in the network, there is no need for the Av2HP to generate and send STUB messages. However, in that case the start of a new path must be implicitly recognized by the Client. ADASIS v2 defines such mechanism only for the single-path horizon case (section 3.3).

- SEGMENT message contains most important profiles packed together. If neither of those profiles is used by any application, the Av2HP does not need to send SEGMENTS.
- PROFILE messages will be sent if specific applications have need for the corresponding profile data.
- META-DATA message is not required to send if proper set of defaults are hard-coded on the client side. META-DATA, however, contains a number of semi-static fields (country, region, speed limit units ...) that may change during driving as well as basic version information of the Horizon Provider. If any application uses this information, the META-DATA message must be transmitted by the Av2HP.

## 5.2 STUB types

STUB messages describe the start of a new path that has its origin on an existing one.

Fields PATH INDEX and OFFSET define the exact position on the existing path where new one starts. As such, crossings are defined as several STUBs at the same OFFSET of the PATH INDEX. In any case, PATH INDEX and SUB-PATH INDEX of STUB message must be different. SUB-PATH INDEX may be (see also Table 7: Path Indices with Special Meaning):

- **Value 5:** For these STUBs, no physical path is created, i.e. no messages will be appended. This Stub type is typically used for systems with very low bandwidth or storage capacity or for very low probability branches or for disallowed turns at intersections, barely to indicate their existence. If the vehicle turns to a stub with Path Index 5, the Av2HP must initiate a new path. Typically, Av2HPs providing only the 'single-path-horizon-with-stubs' are sending STUB messages with a SUB-PATH INDEX value of 5.
- **Value 6:** As opposed to normal STUBs connecting side-branches to the main path, a STUB with a SUB-PATH INDEX of 6 provides information for the continuation of the main path such as TURN ANGLE and RELATIVE PROBABILITY. Most of the content of such a STUB message (such as NUMBER OF LANES) may be redundant as there is already a SEGMENT message with the same data. We will refer to this kind of stub as a *continuation stub*.
- **Values 8 to 63:** "Normal" STUB message, defining the origin of a new path; subsequent messages referring to this Index value will provide more data about it.

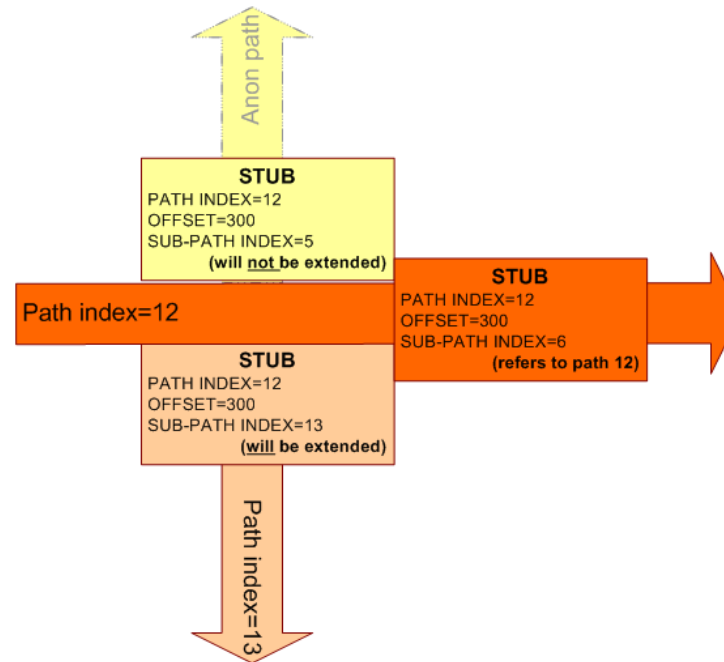


Figure 18: Three types of STUB message

### 5.3 Implicit STUBs

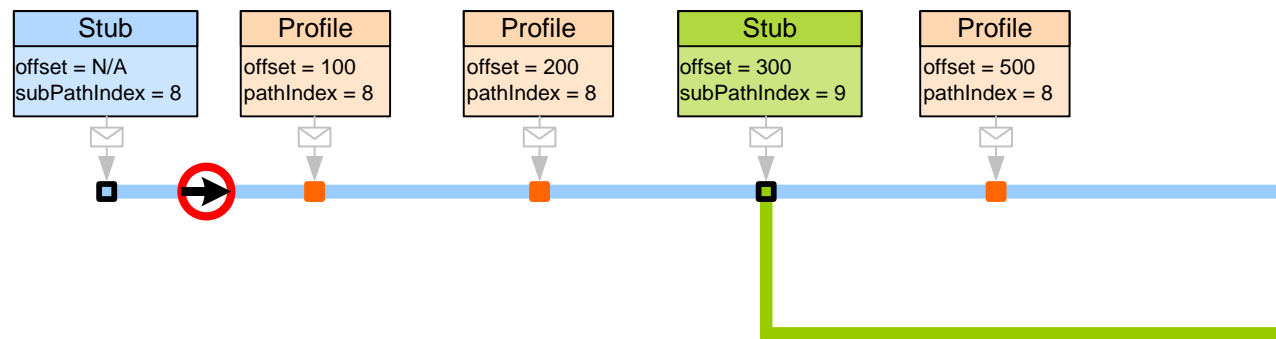
In general, the Av2HP uses STUB messages to inform the Av2HR that a new path entity is to be created. In that case, POSITION, SEGMENT and PROFILE messages will refer to a path index that was already contained as sub-path index in a previously sent STUB message. The POSITION message may refer to path index that was not previously created using a STUB; in other words, at the very beginning, the Av2HP may send a POSITION as the first message in the communication. It should be avoided, however, to have a POSITION message for a path that only will be introduced in a later STUB message. Because the POSITION path index may be reused before receiving the corresponding STUB message, the application may incorrectly position the vehicle on the 'previous' path.

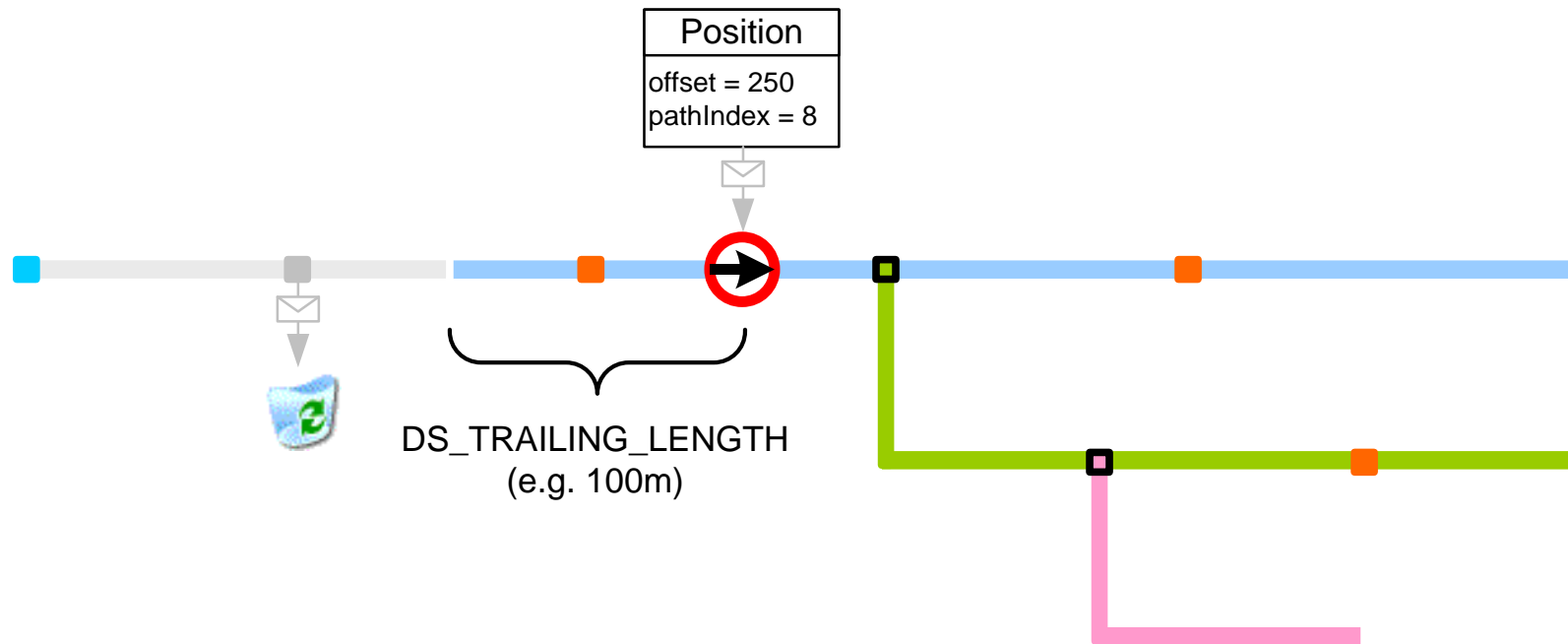
If the Av2HP is configured to only send a single-path horizon, the path entity may be created implicitly at the Av2HR when POSITION, SEGMENT or PROFILE messages refer to a new path index. As a result, the single-path horizon will use only two path indices, say 8 and 9. While the vehicle is driving the original (most-probable) path, all messages will use one same path index (say 8). From the moment when POSITION, SEGMENT or PROFILE messages refer to a different path index (say 9), the Av2HR shall conclude that the vehicle has left the original path, abandon all information of the 'old' path and start to transmit data for the new one.

## 5.4 Implicit entity removal on client

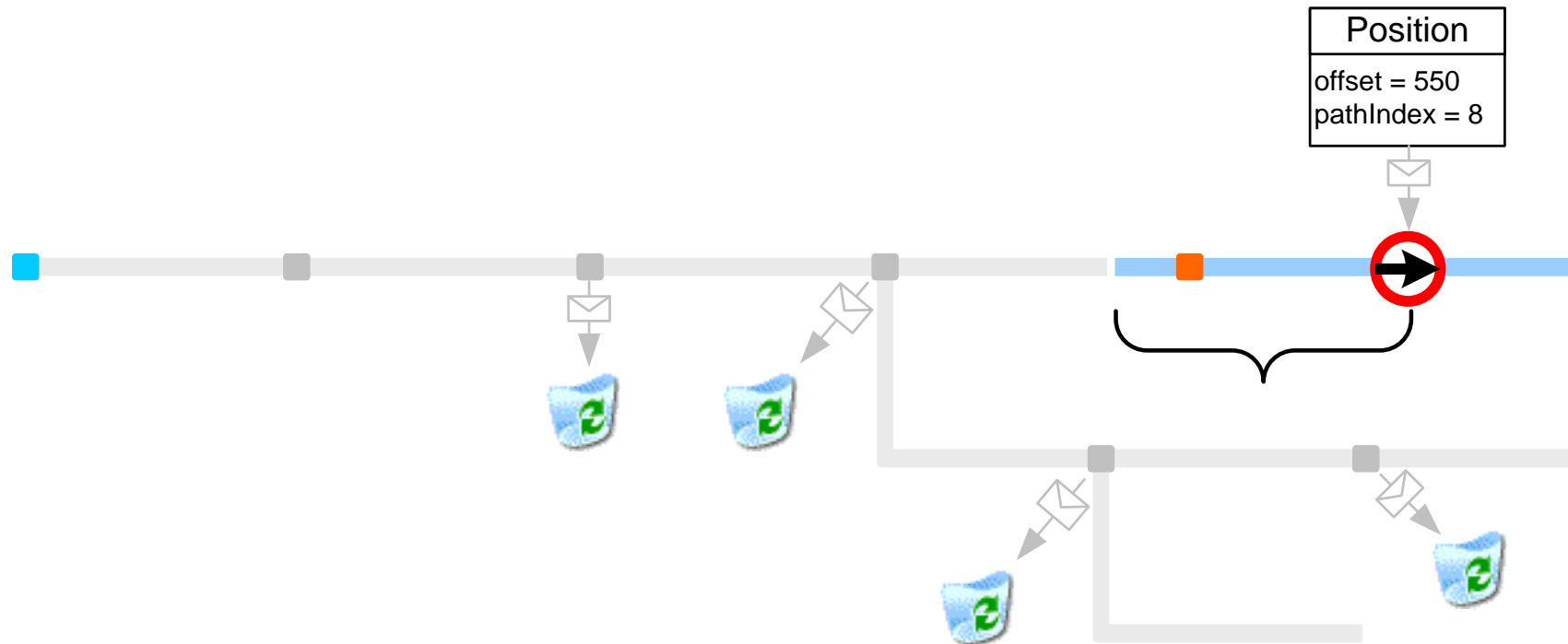
This section describes how the AV2HR handles deletion of profiles and stubs behind the current vehicle position. The guidelines also show how the Av2HP must handle special use cases to be consistent with the reconstructor content to avoid "memory leaks".

As a precondition, the Av2HP builds up a path with several profiles and one sub path. It is recommended to start creating the horizon at a positive (non-zero) offset along the first path, as some applications may always look behind the current position. This is particularly important at start-up, as the current vehicle position is still inaccurate and could therefore jump backwards.

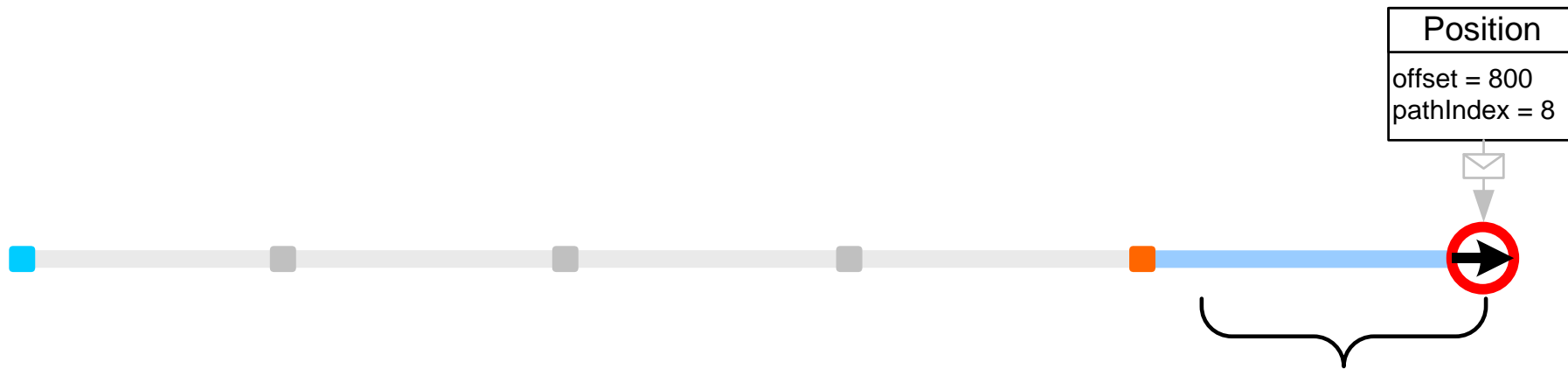




As the vehicle moves along the path, the Av2HP sends further profiles and one sub-path. The Av2HR should have a defined trailing length value “maximum horizon trailing length” (called DS\_TRAILING\_LENGTH in the reference Av2HR source code), which defines the distance back from the vehicle before entities can be deleted from the data store of Av2HR. This value must be consistent with the corresponding setting of the Av2HP.



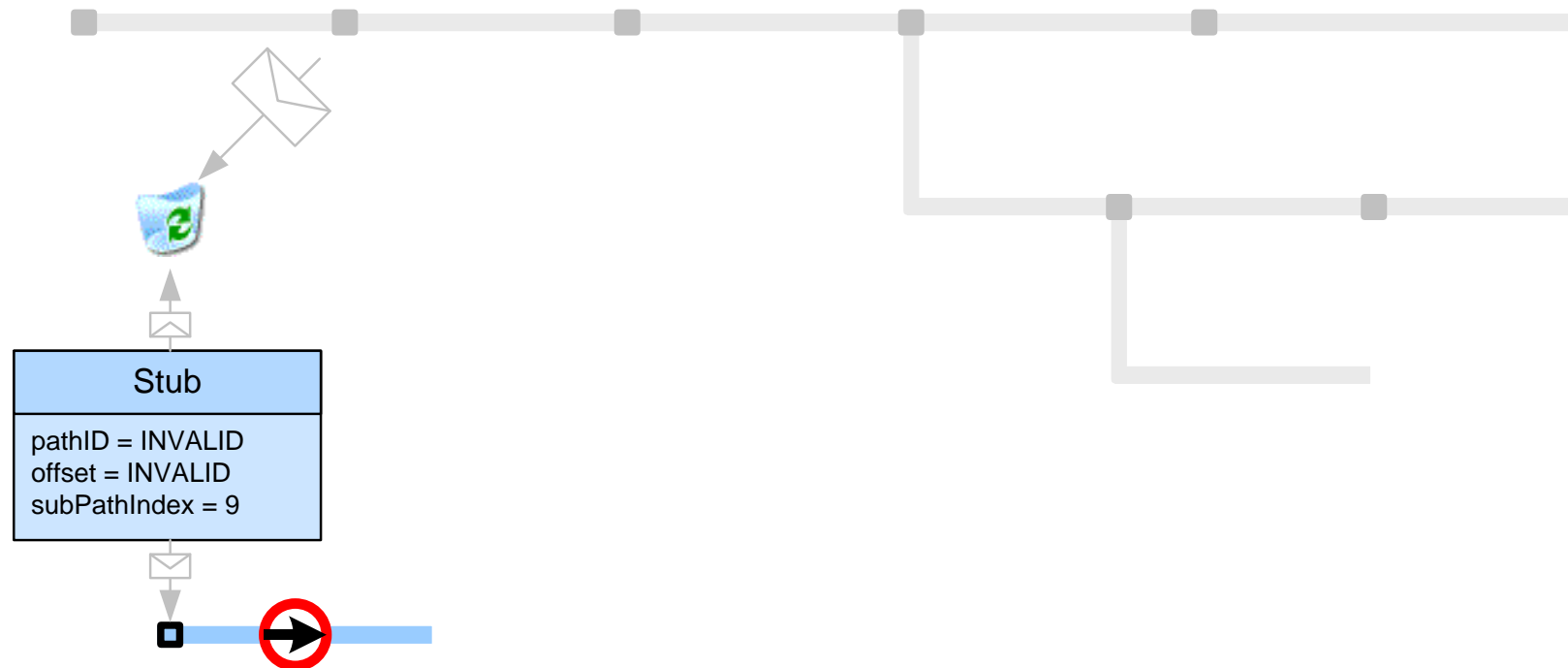
Another use case is when the vehicle passes a junction as defined by stubs. Not only the profiles and segments behind the vehicle can be removed, but since all sub-paths reachable from these stubs are no longer reachable by the vehicle, they can also be removed.



The stub information of the current path and the last valid profile behind the vehicle are two exceptions. Stub information will not be removed as long as the vehicle stays on this path, because it is needed for the application to retrieve path information concerning the current path. The last valid profile will also not be removed, because this could define the current profile value at the vehicle offset (e.g. current speed limit).

At this point it is important to ensure that, apart from transmission latencies, the provider's and the reconstructor's view of the current horizon shall not be different. The system setup must ensure that the reconstructor does not run out of memory while the provider is still sending data.





To completely delete the reconstructor's content, the provider shall send a new stub with path index == INVALID and offset == INVALID. With this information, the reconstructor will completely initialize its data. The provider should always send this special initial stub before the horizon is built up from scratch. This could be the very first time after system start-up, if the vehicle position cannot be matched to the ADAS Horizon, or on request, e.g. by a complete retransmission request of the Av2HR application. This is especially useful if the ADAS Horizon application is starting up later than the Av2HP, or if the application gets an out-of-memory error code from the reconstructor. A retransmission request mechanism is not specified by the ADASIS protocol and must be defined system-specifically.

## 5.5 ADAS Horizon Path Creation in case of Vehicle Position Jumps and U-Turns

### 5.5.1 *Backward Jump of Position Offset*

The current vehicle position may jump backwards because of positioning and map-matching errors.

To allow backward position jumps it is recommended to start creating a new ADAS horizon in the Av2HP at a certain distance behind the current vehicle position. Moreover, when removing ADAS horizon entities behind the vehicle, it is recommended to keep all entities in the Av2HP and all ADASIS v2 horizon reconstructor unto a certain distance behind the current vehicle position.

If the vehicle position jumps backwards more than this certain distance, the Av2HP deletes the current path completely and must start a new path transmission also containing the new vehicle position. This new path is *not* connected to the former path by a STUB message. The allowed distance to jump backwards shall be set to an appropriate value in order to avoid frequent path deletion and creation steps.

### 5.5.2 *Vehicle makes a U-Turn Maneuver*

If a vehicle changes the direction in which a road is driven, the position offset of the vehicle with respect to the current path will decrease. When detecting a driving direction change the Av2HP shall delete the current path and create a new path in the new driving direction. This new path is *not* connected to the former path by a STUB message.

### 5.5.3 *Vehicle drives backward*

Normally the position offset of a vehicle on the ADAS horizon will increase in time. If a vehicle drives backward, the position offset will decrease until the maximum backward jump distance has been reached.

After that this situation can be handled as a U-turn maneuver creating a new path in the moving direction of the vehicle.

## 5.6 Updating of content after transmission

Most of the information that is transferred in SEGMENT, STUB and PROFILE messages is read from the digital map database and is of static nature in the sense that it does not change once it is transmitted. However, some of the data can change dynamically during the lifetime of the path. For instance:

- *Part of Calculated Route* SEGMENT attribute may change its value if the driver diverts from the original route and the system recalculates another one which follows a different path.
- *Relative Probability* of the STUB may change if the ADAS Horizon is constructed incrementally and new path data is revealed that influences probability calculations; for instance, when new route is calculated.
- SEGMENT *Effective Speed Limit* and *Effective Speed Limit Type* may change because of changed weather conditions; sudden rain, for instance;
- The PROFILE message may also contain Speed Limit or other sign information that depends on external factors such as weather conditions.

When changes are detected, Av2HP will inform the clients about new values of previously transmitted data.

To accomplish an update of the data on the client, *Update* flag of SEGMENT, STUB and PROFILE messages must be set to *true*; so that the client can correctly identify newly received messages as updates of already received ones, and not as new data that is to be inserted into the existing path network.

The OFFSET value of an update message shall be the same as for the corresponding original message. Consequently, the messages will not be sent out in increasing offset order in case of update messages.

If data needs to be updated that is behind the current vehicle position by more than the "maximum horizon trailing length", then it is not possible to directly update the data.

As a consequence of the possible change of the *Relative Probability* of a stub during runtime, the definition of the most probable path may also change. For example in Figure 6 let Path 2 be the most probable path and the main path with a calculated route along Path 2. If the navigation system would do a re-routing because of a traffic jam at the end of path 2, the most probable path could be Path2->Path1. This means that the most probable path may change the path index during runtime.

## 5.7 Profile Interpolation using Analytic Curves

In the ADASIS v2, *Profile* is a characteristic of a path that has a defined value for every location along the path. Typical examples of profiles are road curvature or road heading, but even absolute 2-D road geometry can be described as a profile (section 6.6).

### 5.7.1 Basic approach

Profiles are defined by specifying *profile support points* and by specifying the *interpolation* method that is to be used to calculate the profile between support points. The Interpolation method is defined by profile type (Table 17 and Table 22). In general, it is allowed that a single physical profile (say, curvature) is actually transferred using multiple profile types (say, CURVATURE/LINEAR and CURVATURE/STEP) that partitions the path into parts that are interpolated using, for instance, linear interpolation, and parts where step interpolation should be applied.

Profiles in ADASIS v2 protocol are communicated in a PROFILE messages (section 4.7 and 4.8). This message can be used in different ways to accommodate both simple as well as complex functions describing a particular profile. Even if a function that is more complex is used, the client can restrict itself to applying simpler interpolations if not enough CPU power or memory is available.

The System can communicate PROFILES in a way to minimize traffic (Figure 19), or it can decide to send redundant data and to increase the robustness of the protocol (Figure 20). Based on received messages, the ADASIS v2 Horizon reconstructor can handle both variants at the same time.

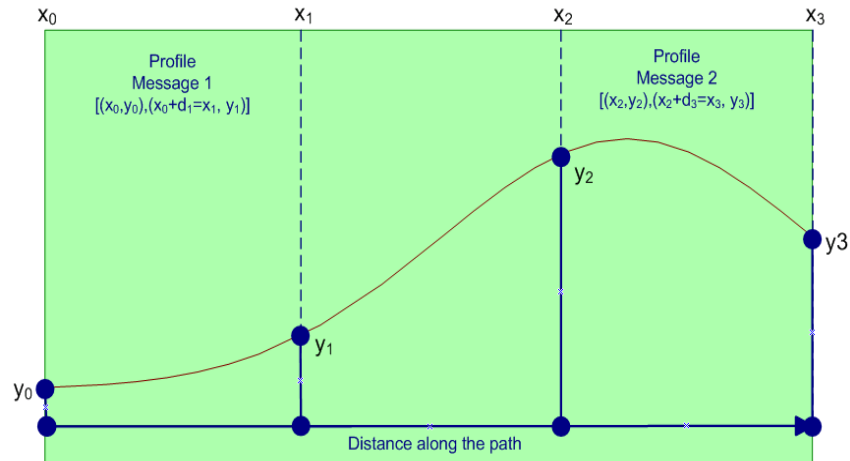


Figure 19: PROFILEs in ADASIS v2 – optimized for bandwidth

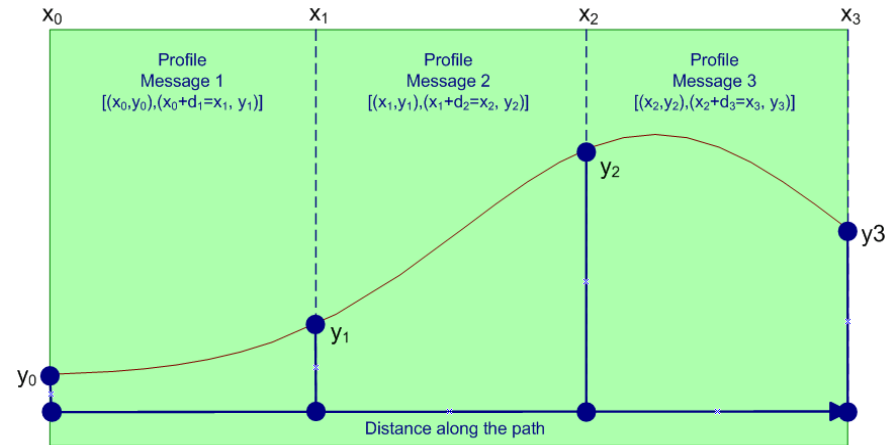


Figure 20: PROFILEs in ADASIS v2 – optimized for robustness

For ADASIS v2 Clients, the most commonly used interpolations of profile functions are *step* interpolation (Figure 21) and *linear* interpolation (Figure 22).

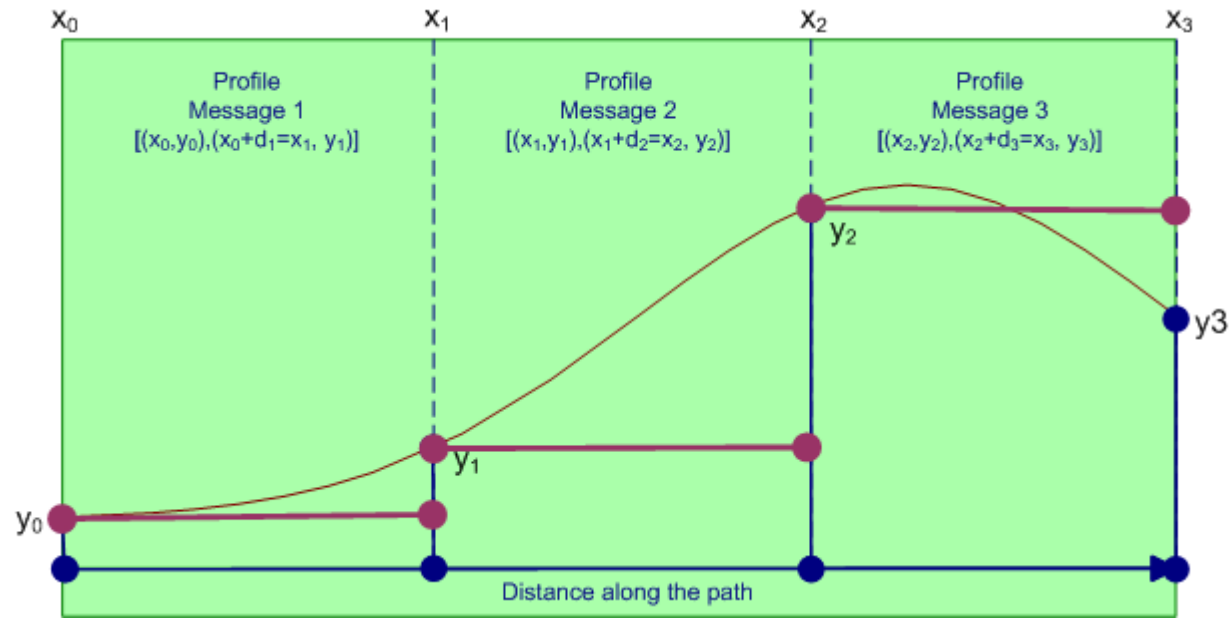


Figure 21: Step Interpolation

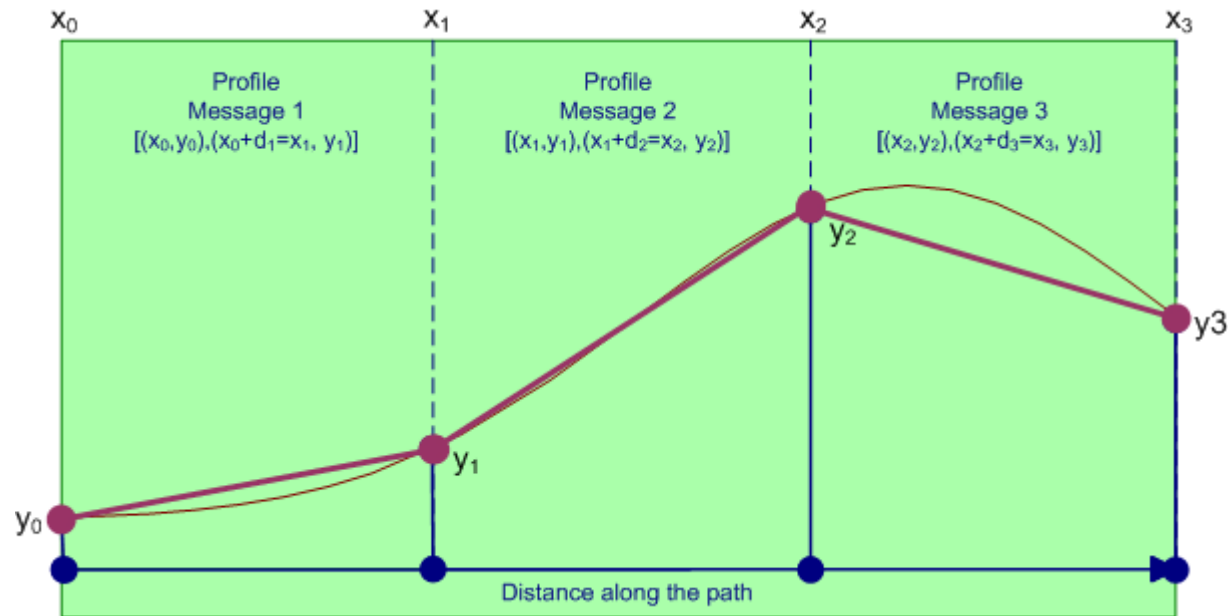


Figure 22: Linear Interpolation

### 5.7.2 Complex interpolation functions

The CONTROL flag in the PROFILE message enables transfer of additional parameters that defines interpolation functions. For instance, step and linear interpolation are, generally speaking, polynomial interpolations of degree 0 and 1. By sending an additional PROFILE message with a *control* point, one can send an additional profile parameter that defines second-degree polynomial interpolation (Figure 23).

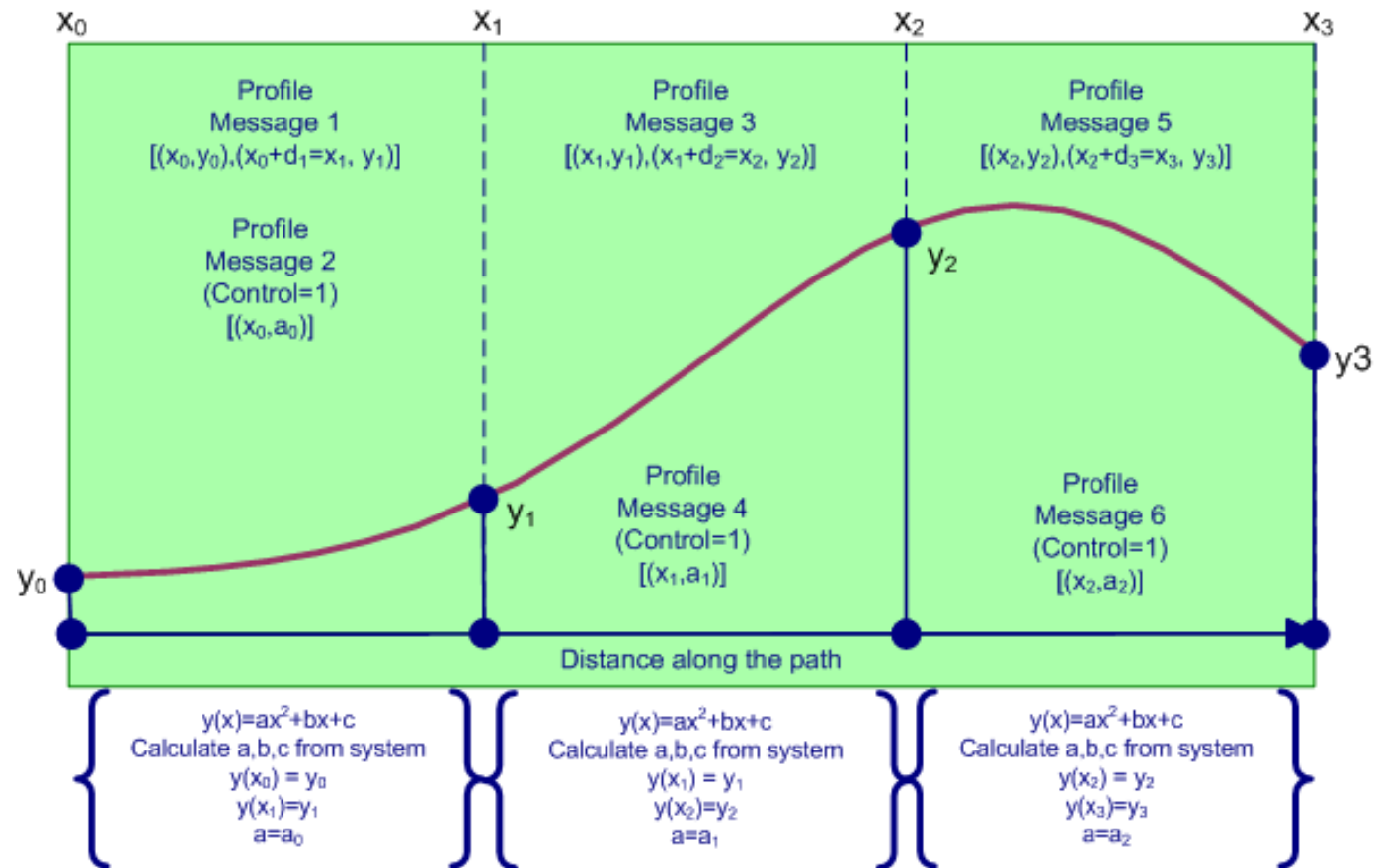


Figure 23: Second-degree polynomial interpolations



The same mechanism can define a polynomial of arbitrary degree for the profile interpolation. For instance, to use a 5<sup>th</sup> degree polynomial,  $y(x) = a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$  the following PROFILE messages need to be sent.

Field	Value	Comment
<b>Message Type</b>	4	PROFILE SHORT
...	...	...
<b>Offset</b>	$x_0$	Left boundary of interpolated interval
<b>Control Point</b>	false	
<b>Value 0</b>	$y_0$	$y_0=f(x_0)$ , left boundary value.
<b>Distance 1</b>	$x_1 - x_0$	Length of interpolated interval
<b>Value 1</b>	$y_1$	$y_1=f(x_1)$ , right boundary value.
...	...	...

Table 25: 5<sup>th</sup>-degree polynomial PROFILE interpolation – Message 1

Field	Value	Comment
<b>Message Type</b>	4	PROFILE SHORT
...	...	...
<b>Offset</b>	$x_0$	Left boundary of interpolated interval identifies interval for which interpolation is defined.
<b>Control Point</b>	true	
<b>Value 0</b>	$a_2$	
<b>Distance 1</b>	0	Indicates 1 <sup>st</sup> pair of interpolation parameters
<b>Value 1</b>	$a_3$	
...	...	...

Table 26: 5<sup>th</sup>-degree polynomial PROFILE interpolation – Message 2

<i>Field</i>	<i>Value</i>	<i>Comment</i>
<b>Message Type</b>	4	PROFILE SHORT
...	...	...
<b>Offset</b>	$x_0$	Left boundary of interpolated interval identifies interval for which interpolation is defined.
<b>Control Point</b>	true	
<b>Value 0</b>	$a_4$	
<b>Distance 1</b>	1	Indicates 2 <sup>nd</sup> pair of interpolation parameters
<b>Value 1</b>	$a_5$	
...	...	...

**Table 27: 5<sup>th</sup>-degree polynomial PROFILE interpolation – Message 3**

While  $a_2$ ,  $a_3$ ,  $a_4$ , and  $a_5$  are explicitly defined,  $a_1$  and  $a_0$  needs to be calculated from boundary conditions defined in message 1.

This approach has the following advantages:

- The Client can interpret the profile as 1<sup>st</sup> order by simply ignoring CONTROL point messages; it can, consider only the first one and use linear interpolation.
- Missing parameters are easy to detect because the combination of OFFSET and DISTANCE 1 fields uniquely identifies the interpolation polynomial; the fallback solution is linear interpolation.

- The choice of interpolation functions is not limited to polynomials of up to 5<sup>th</sup> degree; every function whose parameters can be expressed as a combination of boundary conditions and explicit numbers can be used. If the calculation of parameters for out of boundary conditions is too complex, PROFILE/CONTROL messages will contain all parameters explicitly. It is still useful to send profile values on both ends of the interpolation interval in case the client wants to use linear interpolation.

## 5.8 ADASIS v2 Mini

Where clients are only interested in the attributes of the current link, Av2HP could decide not to send the ADAS Horizon in the form of Paths. Instead, a SEGMENT message with a PATH INDEX of 4 describes the attributes of the link where the vehicle is currently located. Since only one message is sent (or perhaps sent at regular intervals), we can refer to such an Av2HP as a minimal version of the system.

This minimal version is intended as an additional information layer to provide basic map information for the current vehicle position (e.g. to change the set-up of the air-conditioning in tunnels).

Since SEGMENT message of ADASIS v2 Mini is clearly different from “regular” SEGMENT messages, single Av2HP can support both regular ADASIS V2 and ADASIS v2 Mini in the same time.

In the case of the ADASIS v2 Mini message, some signals of the SEGMENT message do not make sense and need to be set to default values according to the following table:

<b>Signal</b>	<b>Value in ADASIS v2 Mini</b>	<b>Comment</b>
<b>Message Type</b>	2	Same as for SEGMENT
<b>Cyclic Counter</b>	0 – 3	Increased for each ADASIS v2 Mini message separately to the SEGMENT message.
<b>Retransmission</b>	false	No retransmission concept
<b>Path Index</b>	4	Indicates, that this SEGMENT message is an ADASIS v2 Mini message
<b>Offset</b>	8191	Invalid offset because ADASIS v2 Mini message refers to the current vehicle position
<b>Update</b>	False	No Update concept
<b>Relative Probability</b>	31	Invalid.  The relative probability signal shall not be used to indicate the probability of the current position. If the vehicle is not map-matched or if the map-matched position is too unsure, the provider shall not transmit ADASIS v2 Mini messages or shall use 'invalid' values for all signals.

**Table 28: Default Values in ADASIS v2 Mini Message**

All other signals shall contain the respective map attributes for the current vehicle position.

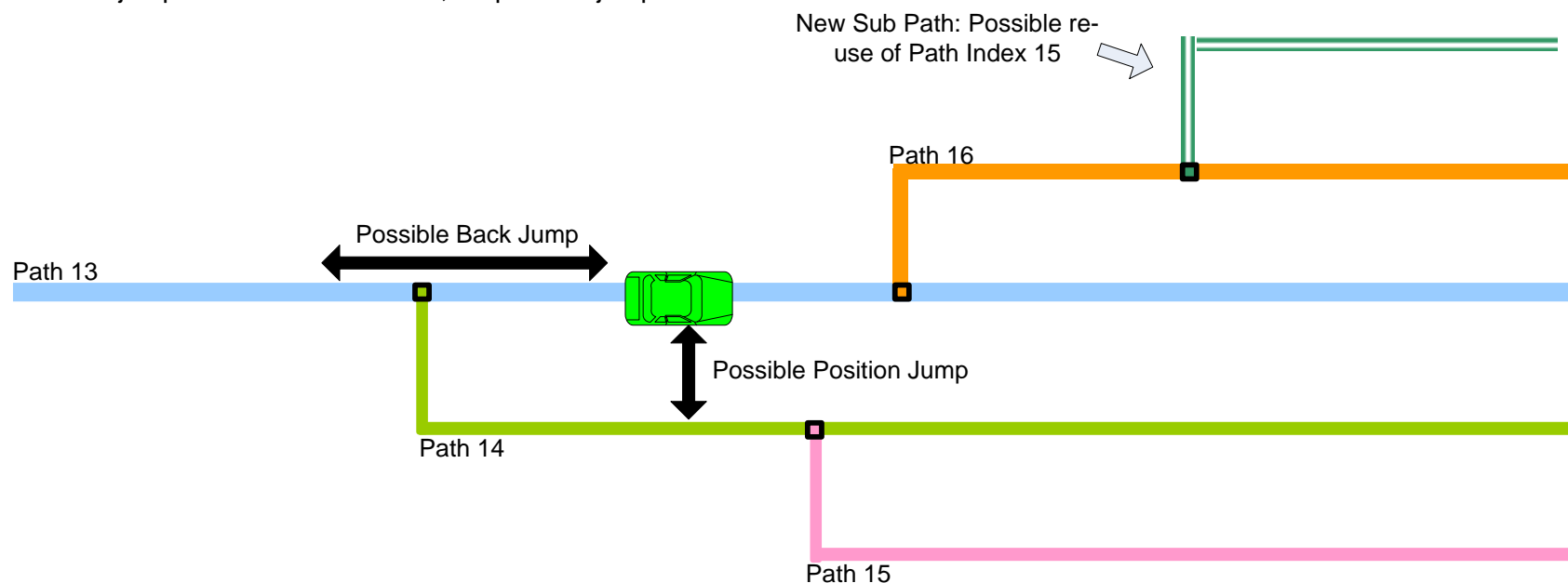
## 6 *Guidelines and Recommendations*

### 6.1 ADAS v2 Horizon Provider (Av2HP)

**Av2HP Recommendation #1** During system start, META-DATA messages should be transmitted more often to allow rapid start-up for clients requiring availability of basic configuration, map data, and ADAS v2 Horizon Provider characteristics.

**Av2HP Recommendation #2** For ADAS applications that are sensitive to a loss of SEGMENT message in the middle of the path, LENGTH profile should be generated by the Provider. This PROFILE message will contain the offset at the start of the segment and length of the segment and enables the Av2HR to identify parts of the paths where values of attributes and profiles contained in the SEGMENT message are missing.

**Av2HP Recommendation #3** The horizon provider should wait a certain distance until he re-uses path indices for new paths. If a path index e.g. 15 has already been used (not for the main path), the vehicle position should have passed the root of path 15 attached to the main path by at least the configured maximum back jump distance until this path index 15 can be re-used to attach a completely new path at the far end of the horizon. Also, the current vehicle position should be with a high probability not on a root with sub path index 15 until path index 15 can be re-used. This increases the robustness and prevents possible index mix-ups in the case of position jumps. For back jumps see also section 5.5, for position jumps see section 6.4.



**Figure 24: Re-Use of Path Indices**

**Av2HP Recommendation #4** The horizon provider should not use the complete maximum offset range. It should reduce the maximum path length of transmitted path by a certain distance to reduce the risk of mis-interpretations of cyclic offsets.

**Av2HP Recommendation #5** It is recommended to start creating the horizon with the vehicle position located at some positive offset, because some applications might use the possibility to look backwards. A second reason is that especially at start-up; the current vehicle position is still inaccurate and could therefore jump backwards.

**Av2HP Recommendation #6** If a certain attribute does not change for a long stretch of the horizon, the Horizon Provider should generate additional "no-change" messages to indicate that the information is still valid.

Example: If a vehicle follows a motorway, the map attributes in the SEGMENT message may not change for long distances e.g. 10 km. In the strict sense, it would be sufficient to send out this message only once. However, it would increase the reliability of the ADAS application if the Horizon Provider sends out a new message with the same content at regular intervals (e.g. every 1 km). Otherwise, it would not be possible for the Horizon Reconstructor to distinguish whether the map information is still valid or if the Horizon Provider has stopped sending out that information because of e.g. a software error.

## 6.2 Message Timings

System requirements will usually limit the number of messages that can be sent within a defined period, usually within one second. In addition, there should be a predefined minimum delay between messages. In this case, the Horizon Provider must handle these limitations.

While POSITION, SEGMENT, STUB, and PROFILE messages will be generated depending on the current vehicle position, the META-DATA message should be sent periodically. The time between two META-DATA messages, (for instance, one message every 5 seconds) is part of the configuration of the system.

It may be defined in the system requirements that the Horizon Provider should send exactly  $N$  messages per second (for instance,  $N=10$ ). If there are no new messages that need to be sent, the Horizon Provider should choose and re-transmit one of the already sent messages with a certain priority scheme (e.g. randomly, sorted by distance, ...). The retransmitted messages shall be marked with the "retransmission flag" = true and shall have exactly the same data content as the initially transmitted message (including the value of the cyclic counter).

In case no messages are available for re-transmission (for instance, at start of the system), the Horizon Provider may fill the message quota with META-DATA message.

## 6.3 POSITION message priority

Because of the importance of the position in applications, the lowest CAN id may be assigned to the POSITION message.

This may, however, in rare situations lead to inconsistent results at the reconstructed side.

Let us suppose that the POSITION message has a lower CAN id than the STUB message. Av2HP may generate messages in the following order:

1. STUB with path index 11;

2. STUB with path index 10;
3. POSITION with path index 10;
4. STUB with path index 11 (path index reused; old path with index 11 is obsolete);
5. POSITION with path index 11;

If messages are received in the same order, then the position will be correctly interpreted at all times:

- a) No valid position prior to message 3,
- b) Position at path with index 10 after messages 3 and 4;
- c) Position at path with index 11 after message 5;

Because of higher priority of the POSITION over the STUB message, the order in which Av2HR receives messages may be:

1. POSITION with path index 10;
2. POSITION with path index 11;
3. STUB with path index 11;
4. STUB with path index 10;
5. STUB with path index 11.

From the Av2HR perspective:

- a) Position is not valid after message 1;
- b) Position is not valid after message 2;
- c) Position refers to wrong path after message 3;
- d) Position refers to wrong path after message 4;
- e) Position finally refers to right path after message 5.



To achieve better consistency of positioning, POSITION message can have a higher CAN id than the STUB message. In that case, the order of receiving may be:

1. STUB with path index 11;
2. STUB with path index 10;
3. STUB with path index 11;
4. POSITION with path index 10;
5. POSITION with path index 11.

Now, the position will be either invalid (messages 1-3), or consistent (message 4 and 5).

This approach, however, may result in an unwanted latency of POSITION message.

## 6.4 Use of Alternative Positions in prevention of ADAS Horizon re-initialization

Within the Av2HP, more than one Map-Matched position might exist, especially in the case of a dense road network, where after passing an intersection it may not be clear which path the vehicle has taken. Especially if two alternatives have similar turn angles, such as at bifurcations, and run in parallel for a certain distance.

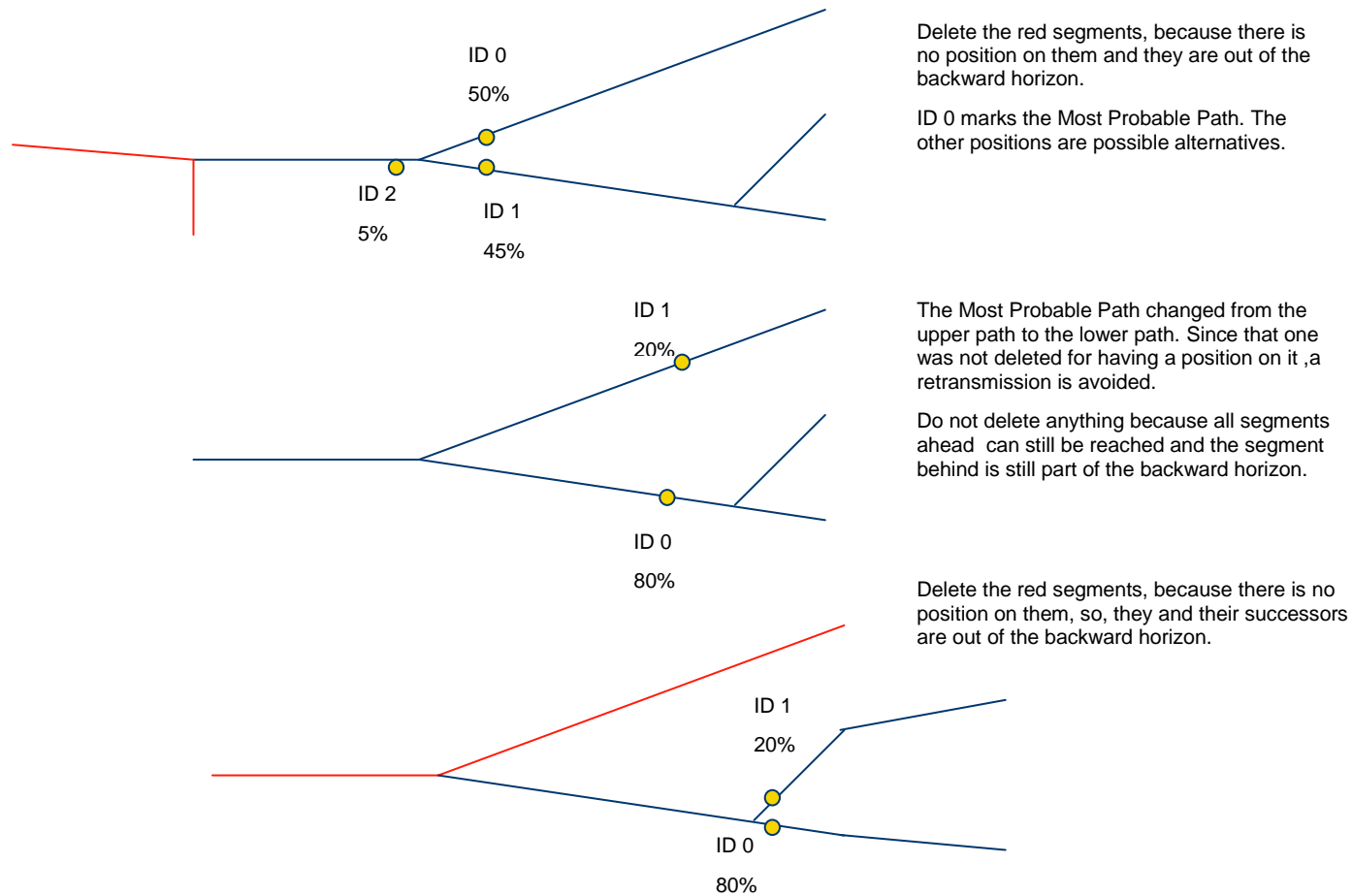
Since no explicit deletion message is defined for segments and stubs, the Av2HR needs to decide when to delete segments. This could be done for all data that is more than, for instance, 100m behind the vehicle and for the alternative paths that have origins on these deleted segments and therefore cannot be reached anymore. Such an approach has the disadvantage that segments associated with probable alternative positions, e.g. at bifurcations, can be deleted in the Av2HR. If the Av2HP position jumps to one of these alternative paths, it does not know if this new path is still known by the Av2HR or has already been deleted. It therefore needs to retransmit the data that may have been deleted and potentially the application will be “blind” during the transmission time. This retransmission would also increase the CAN traffic.

To overcome such an effect an option could be to keep the segments and alternatives behind the current position until the current position probability has exceeded a threshold of, for example, 90%. Unfortunately, in dense city networks this might never be the case, so deletion would not occur for a long time and the number of available segment indices would be exceeded.

For these situations, it is meaningful to transmit map matched position alternatives. If the sum of these position probabilities equals 100%, the Av2HR knows that it can delete all segments that are lying behind the addressed segments, as well as all their unreachable successors. The applications shall only use the first available position (index 0) as being the most probable alternative. The other position alternatives shall normally not be considered by the application. However, lower probability position alternatives may later turn to be the most probable one, so they are maintained by the Reconstructor for housekeeping of the path memory and coping with varying probabilities.

This approach avoids unnecessary retransmissions but requires a higher bandwidth since several position messages will be transmitted within one second instead of only one.

If future applications would need to know all (up to four) different positions at the same time, the specification provides for the possibility of using all alternatives.



**Figure 25: Alternative positions**

## 6.5 Profile interpolation: pre-calculated data or real-time approximation

The function and its parameters that describe each profile along the path can be stored explicitly in the database. In this case, the task of the ADASIS v2 Horizon Provider is to read and format those parameters and send them to the Av2HP client.

Alternatively, profiles may be defined point-wise, for instance on each shape point.

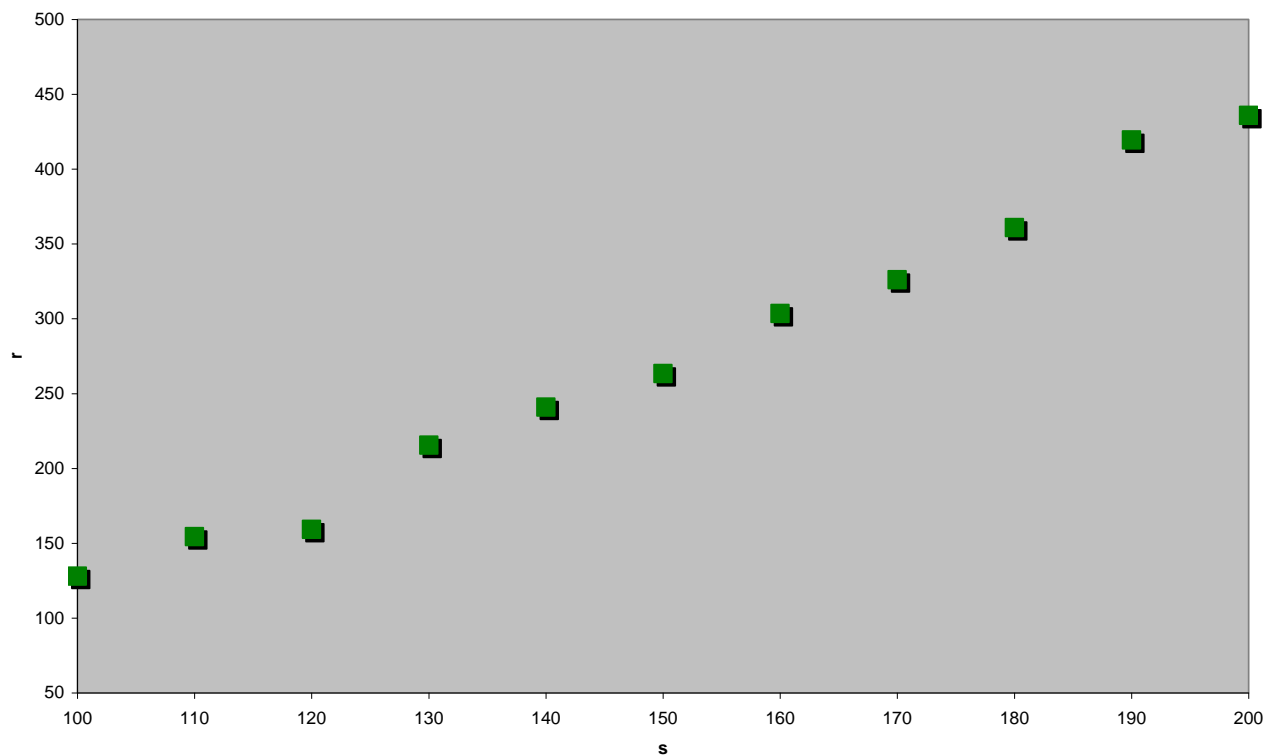
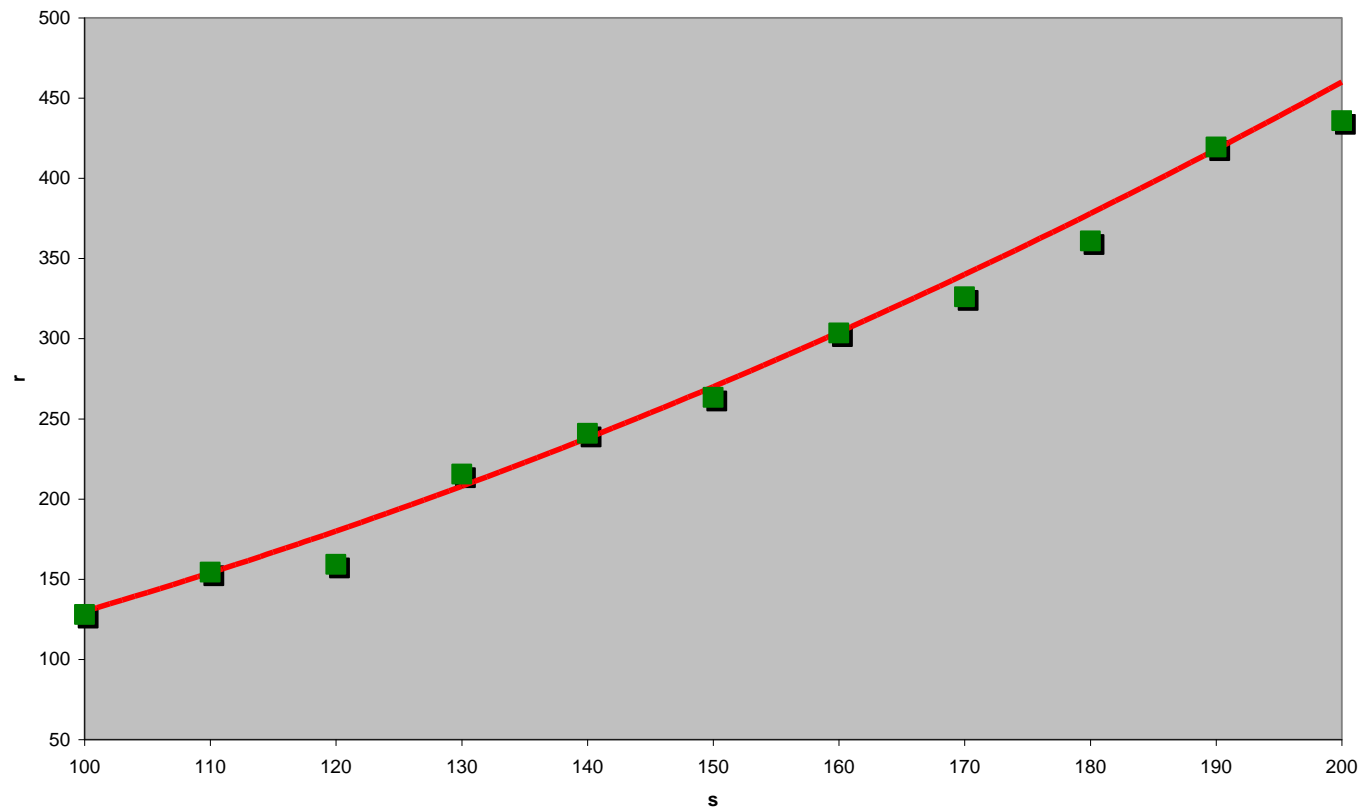


Figure 26: Shape-point based profile example (here: curvature radius  $r$ )

The straightforward implementation of the ADASIS v2 Horizon Provider will send one PROFILE message for each interval between shape points, and define step or linear interpolation for points in between.

Horizon Providers that are more sophisticated can try to find the best interpolation method in order to minimize network traffic.



**Figure 27: Optimized shape-point based profile**

For instance, as shown in Figure 27, a 2<sup>nd</sup> degree polynomial may approximate data very well and the horizon provider can decide to send, instead of many short linearly-interpolated segments, one long interval on which a single, but more complex, interpolation can be applied.

## 6.6 Absolute Road Geometry

The absolute road geometry, expressed in terms of longitude/latitude coordinates has no explicit place in the ADASIS v2 protocol. We assume that in ADAS applications this information has very limited usability. Nevertheless, absolute road geometry can be easily published by ADASIS v2 Horizon Provider as two PROFILES: LONGITUDE profile and LATITUDE profile.

For instance, when we want to transmit coordinates of shape points present in classical digital map databases, the following process can be implemented:

- Determine what shape points are along specific path;
- Calculate offsets of shape points along the specific path;
- For shape point at offset  $x$  on the specific path, generate signed 32-bit profile value of LONGITUDE;
- For shape point at offset  $x$  on the specific path, generate signed 32-bit profile value of LATITUDE.

Linear interpolation of LONGITUDE and LATITUDE PROFILES will in effect result in classical, poly-line based road geometry. Other interpolation types can describe road geometry in ways that are more precise. For instance, second or third degree polynomial interpolations will represent roads as splines and this approach may reduce the number of shape points that need to be transmitted.

Alternatively, instead of sending 32-bit longitude/latitude values, the provider may send only 10-bit coordinate deltas relative to previous shape points. While the number of messages is reduced thereby, an additional PROFILE type must be created for sending origin points from time to time. In addition, such method is less robust because loss of message will invalidate remaining PROFILES.

## 6.7 Sample system-specific PROFILES

ADASIS v2 system can support up to 30 different 10-bit profile types (1-31). Of those, profiles with types 1-15 will always follow the standard as described in Table 22, while profiles 16-31 are system specific. Table 29 illustrates one possible definition of profile types.

<i>PROFILE Type</i>	<i>Description</i>	<i>Value Interpretation</i>	<i>Profile Range</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>0</b>	Invalid				
<b>1-15</b>	See Table 17				
<b>16</b>	Banking	Super-elevation in 0.05 degrees	-25.55° ... -25.55°	Linear	0
<b>17</b>	Banking (high resolution)	Super-elevation in 0.01 degrees	-5.11° ... -5.11°	Linear	0
<b>18</b>	Lane 1+2 Width	Low 5-bit width of lane 1 in 0.25m units; high 5-bit width of lane 2 in 0.25m units	0 ... 25.5 m per lane	Step	Country and road type dependent
<b>19</b>	Lane 3+4 Width	Low 5-bit width of lane 3 in 0.25m units; high 5-bit width of lane 4 in 0.25m units	0 ... 25.5 m per lane	Step	Country and road type dependent

<i>PROFILE Type</i>	<i>Description</i>	<i>Value Interpretation</i>	<i>Profile Range</i>	<i>Interpolation Type</i>	<i>Default Value</i>
<b>20</b>	Lane 5+6 Width	Low 5-bit width of lane 5 in 0.25m units; high 5-bit width of lane 6 in 0.25m units	0 ... 25.5 m per lane	Step	Country and road type dependent
<b>21</b>	Alternate radius	Curve radius in meters.			
<b>22-31</b>	Unused				

**Table 29: System Specific PROFILE SHORT Types**

## 6.8 Change of Speed Limit units in run-time

The speed limit signal in the SEGMENT message needs to be interpreted in combination with the speed unit signal in the META-DATA message. The META-DATA message can only transmit the speed limit unit for the current position, whereas the speed limit of the SEGMENT message is predictive information.

In rare occasions, e.g. at the border from USA to Canada, the speed limit units can change between kilometer per hour to miles per hour or vice versa during run-time. If the path crosses such a border, the speed limits along the path will be in the unit of the current vehicle position, thus the speed limit values are partly in one unit and have to be converted partly from the other unit to current unit. The horizon reconstructor has no possibility to distinguish the two different speed units along the path. Only at the moment where the vehicle crosses the border, the META-DATA message will indicate the changed speed unit and the reconstructor can adapt accordingly.

Depending on the application, this non-predictive speed unit transmission can lead to malfunctions. Therefore it is recommended, to end a path if a change of speed unit would occur and to restart the complete horizon at a border where a change of speed unit occurs. This situation is very rare and it can be assumed, that the vehicle will need to stop at such a country border anyhow and that applications would not be affected by the re-start of the horizon provider.



## **7 Error Detection and Recovery**

The 2-bit Cyclic Counter that is part of each message can be used by the reconstructor to detect missing CAN frame; therefore ADASIS v2 protocol enables clients to detect up to 3 missing frames of each message type. The cyclic counter is not on the physical layer of the messages, but on the logical content layer. Each message type and each profile type has its own cyclic counter.

The ADASISv2 protocol provides the mechanism to repeat messages if there is free bandwidth. A re-sent message contains exactly the same content as the initial message except the special "retransmission" flag that is set to "true". If the horizon reconstructor has detected a missing frame, it can listen to the re-transmission messages in order to complete the message sequence.

This protocol is completely uni-directional and does not define messages that the client could send back to the ADAS Horizon Provider to request the specific missing information. Depending on requirements, specific implementations shall implement custom error-recovery mechanism, such as message retransmission request: Av2HR sends to Av2HP message requesting retransmission of one or more messages or even re-initialization of complete ADASIS v2 Horizon.

### **7.1 Missing POSITION message**

Since in all configurations, POSITION message is periodically transmitted, the loss of one frame will only increase the position latency, as the client simply must wait for the next message.

### **7.2 Missing SEGMENT message**

As consequence of SEGMENT message, Client will either have no attributes contained in the message about part of the path (in case of first segment of the path), or it will extend attributes from previous segment to the missing part. While first situation is relatively benign (application is aware of unknown data; it must be designed to handle such situation properly anyway), second case can present significant issue for some ADAS applications.

### 7.3 Missing STUB message

Missing STUB messages will pose a problem for applications for which knowledge about crossings is critical, since the corresponding road-branch at the intersection will be missing.

Moreover, since STUB messages actually inform the client about the creation of new sub-paths, the STUB entity is also used for later deletion of the path connected to it. Subsequent SEGMENTS and PROFILES that refer to a missing STUB sub-path identifier may therefore be wrongly associated to the wrong path.

### 7.4 Missing PROFILE message

Since PROFILE messages are part of the path to which they apply, the application can easily recognize where PROFILE data are not or only partly available and deal with this situation. For critical applications, redundant PROFILES may be generated and transmitted as well.

In case of a missing PROFILE message containing CONTROL points, the client can choose the fallback to linear interpolation of the profile on that part of the path.

### 7.5 Missing META-DATA a message

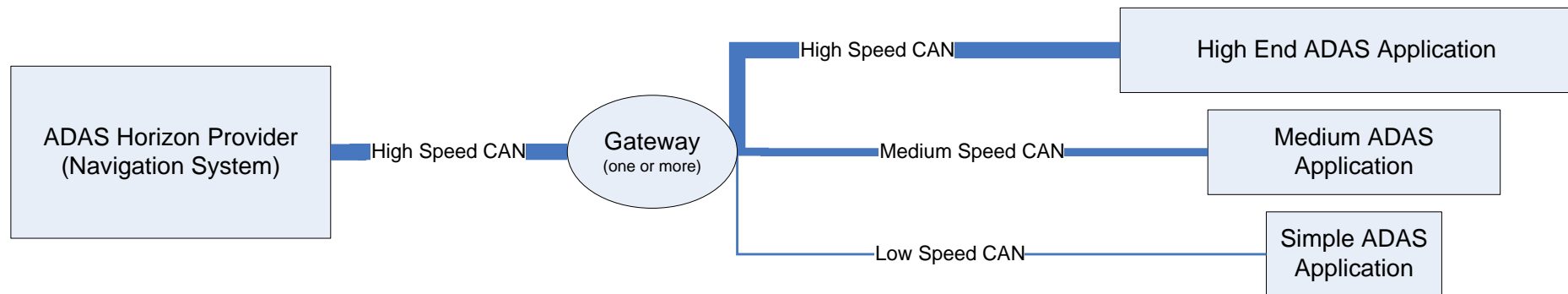
META-DATA messages are sent in periods of several seconds, but contain data that rarely change. Therefore, loss of this message type is not critical for the client function – he simply will catch up when receiving the subsequent META-DATA message. An exception is at the very start of the system, during which the client should rapidly pick up some configuration data from that message. To prevent problems, it is recommended, to transmit META-DATA messages at higher rate after system start-up.

## 8 Examples of use

### 8.1 Different ADAS Horizon Requirements in the Vehicle Network

The typical vehicle set-up has one ADAS Horizon Provider and may contain multiple clients for different applications. The applications may differ in many ways:

- Different map information requirements
- Different control units with different storage capacity and processing power
- Different bus connection bandwidth



**Figure 28: Possible Vehicle Network with different ADAS Applications**

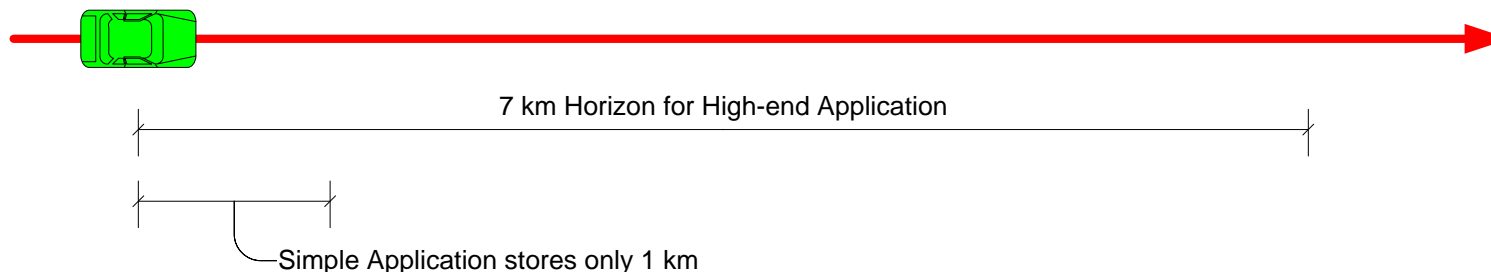
The heterogeneous applications listening to one ADAS horizon broadcast stream require compromises in the parameter settings of the ADAS Horizon Provider to fulfil the different requirements as well as possible.

The different combination of clients in each vehicle will lead to many different possible configuration variants of the ADAS horizon provider. The following section describes two examples and solutions with the ADASIS v2 protocol.

### 8.1.1 Example Set-up 1: Cyclic offsets with different path lengths

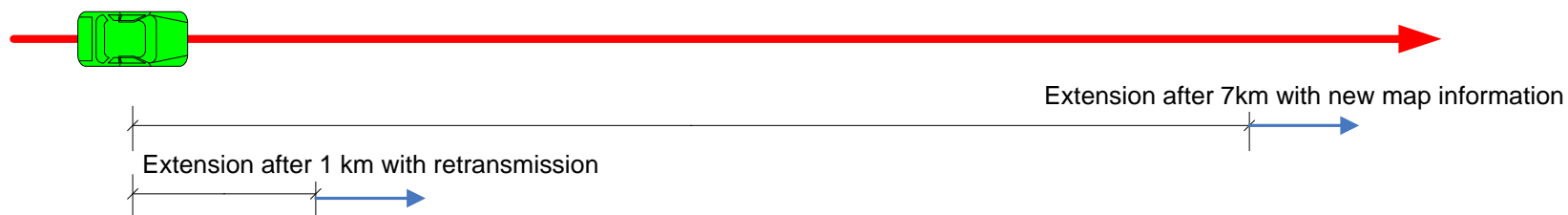
One possible requirement could be that the simple ADAS application can only store a 1000 m ADAS horizon due to storage capacity constraints whereas the high-end ADAS application requires an ADAS horizon of 7 km. The ADAS horizon provider can use the message retransmission concept to fulfil both requirements.

At system start-up, the ADAS Horizon Provider will transmit the complete 7 km, but the simple application will only store 1 km (see Figure 29).



**Figure 29: Different Horizon lengths after start-up**

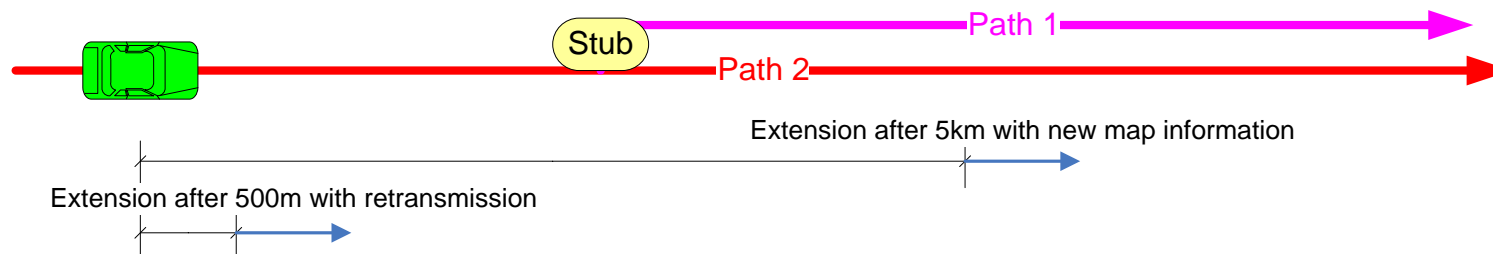
As soon as the maximum length of the transmitted path has been reached, the Horizon Provider needs to follow both lengths (7 km and 1 km) internally. As the vehicle moves forward, the Horizon Provider will attach new information at the end of the 7 km horizon and retransmit already sent information at the end of the 1 km horizon (see Figure 30). The high-end application can filter out the retransmitted information immediately and the simple application discards all information with a distance to the vehicle > 1km.



**Figure 30: Extension of short Horizon with Message Retransmission**

#### 8.1.2 Example set-up 2: Multi-path and main path only

Another possible requirement could be that a simple application can only store information for 500m of the main path and that a more sophisticated application requires a transmitted path length of 5 km with stubs and sub-paths.



**Figure 31: Message Retransmission for Sub Paths**

In this case, the Horizon Provider would need to attach new information to the long 5 km horizon and at the same time retransmit information for the short 500m horizon (as described in section 8.1.1). As an additional requirement, the Av2HP needs to retransmit the complete information of the sub path (Path 1).

If the vehicle drives from Path 2 to Path 1, the high end application would be able to continue without any "blind spot" in the map information. As soon as the vehicle position changes from Path 2 to Path 1, the simple application would have to discard all horizon information, accept a short gap in the horizon and wait for retransmission messages for the beginning of Path 1.

## 8.2 Example Applications and possible Configurations

There are several questions with respect to client requirements a system architect must answer during the definition of an Av2-Horizon configuration:

1. Can the applications accept lack of data for a few seconds? If yes, the single-path horizon can be used. If not, is it acceptable to send just 1<sup>st</sup>-level sub-paths, by which 'preview blindness' situations are strongly reduced, but not eliminated. In the ultimate case when all sub-path information is required to be present on the client-side, a multi level sub-path scheme must be chosen.
2. Do applications need crossing information? If yes, STUBs must be sent. Depending on the answer to the first question, one must choose between single-path horizon with stubs only or single-path horizon with stubs and 1<sup>st</sup>-level sub-paths etc.
3. Do applications rely on attributes of the SEGMENT message? If not, SEGMENT messages need not to be sent.
4. Which additional attributes are of interest? If SEGMENT and STUB messages contain all necessary data, PROFILES are not needed.

As we will see below, besides POSITION message, in many configurations Av2HP needs to send just one message type in addition – STUBs, SEGMENTs or PROFILEs – but there may be no need to send anything else.

However, if there are multiple Av2 clients, the provider must send the total combination of all data required by any of them. Even the less demanding clients must be able to handle all received data in a consistent way, i.e. either process or ignore the sent data.

For instance, a single-path horizon application must be also able to track STUB messages in case the provider is configured to send them (e.g. for another client connected in the same environment).

### 8.2.1 Speed Limit Warning

Speed Limit Warning is the simplest form of an application, since it is not safety-critical. Here, the single-path horizon set up would be sufficient. The Av2HP can send only SEGMENT messages and POSITION message to satisfy all requirements. Data fusion with traffic sign recognition may bring advantages of combining up to date information from the camera with implicit speed restrictions provided by the navigation system.



### 8.2.2 Curve Warning

This application may be seen as safety-critical, but in most cases a single-path horizon with 1st-level sub-paths may serve. SEGMENT messages do not contain any information of interest, so only POSITION, STUBs (for notification about new paths) and PROFILE SHORT messages (for CURVATURE profile) are needed.

### 8.2.3 Intersection Warning

Since this application is focused on intersections only, no SEGMENT or PROFILE information are of general use. Av2HP need to send only POSITION and STUB messages.

### 8.2.4 Lane Change Assist

This application provides warnings for approaching the target lane in combination with a blind spot detection sensor. To enhance the function, additional warning with respect to road information (no passing signs, lane numbers etc.) could be given. This application could work with POSITION, SEGMENT and PROFILE messages only, although the system would perform better in full horizon mode having alternative paths available.



### 8.2.5 Vehicle Dynamics Adjustment

The driver chooses an operation mode (economy, normal, sport) in which adjustments of several dynamic settings are adapted, e.g. damper control, pre-fill of brakes, or even other ADAS applications like curve warning. The ADASIS v2 Horizon could support this function by curve data, road condition, road type or slopes. For this kind of application the Av2HP must send the full set of message types.

### 8.2.6 Fuel Saving Application

Based on the slope profile of the road ahead, potential stop points and road geometry, not only eco driving assistance but also real fuel saving applications can be realized. Especially for hybrid energy management the eHorizon sensor enables a high efficient usage of battery. The Av2HP sends only POSITION and PROFILE information, potential stop points can be compiled on the provider side so that there is no need for STUB messages.

### 8.3 Long-Range Horizon with decreased Offset Resolution

The maximum length of the ADAS horizon is typically limited to the maximum offset value, i.e. 8190 m. For some applications (e.g. predictive energy management), this is not sufficient and longer information about the upcoming path of the vehicle would be needed.

A solution to implement a long-range ADAS horizon is to decrease the offset resolution from 1 m to e.g. 128 m. By doing this, the ADAS long range horizon would cover routes of 1048 km length, which is sufficient for all applications. Most applications that require a long range horizon are in the area of predictive energy management and need only aggregated information, e.g. about the elevation profile. For those, an offset resolution of only 128 m is no relevant loss of accuracy.

The ADAS long range horizon with decreased offset resolution is an addition to the standard ADAS horizon, i.e. new CAN messages with new CAN IDs need to be defined. The short range ADAS horizon and the long range ADAS horizon can be used independently by the interested applications. The CAN messages have exactly the same layout and the software at the horizon provider and at the horizon reconstructor can be reused with attention to the offset resolution differences between the long range and short range ADAS horizon.

In short, the ADASIS long range horizon by changing the offset resolution can be described like this:

- Significantly lower offset resolution than standard ADASIS v2, e.g. 128 m instead of 1 m.
- Only along calculated route.
- Only some relevant and suitable aggregated profile information.
- Specific POSITION message for long range horizon.
- Addition to standard short-range ADAS horizon with a separate set of CAN IDs.
- No specific META DATA message for long range horizon (META DATA of short range horizon shall be used).

The ADAS horizon provider should synchronize the position offset of the short-range and the long-range horizon, that means that e.g. after system start-up, the "offset"=1 in the POSITION message of the long-range horizon (with 128m resolution) is the same position as the "offset"=128 in the POSITION message of the short-range horizon (with 1m resolution). This allows the applications to take the detailed information for the first 8km after the current vehicle position from the ADAS short-range horizon and the information beyond 8 km from the ADAS long-range horizon.



## 9 Current Usage of ADASIS by Different Partners

The ADASIS v2 protocol specification defines the protocol between ADAS horizon provider (Av2HP) and ADAS horizon reconstructor (Av2HR) – it does not define how the protocol is filled by the provider. Based on the ADASIS v2 protocol specification, the provider has some different possibilities, how to fill the messages (e.g. with or without multiplexing).

The following table gives an overview how some partners of the ADASIS forum currently use the ADASIS v2 protocol and how the ADAS horizon providers are implemented. It shows the typical usage of ADASIS v2 in different implementations.

**Table 30: Typical usage of ADASIS v2 Protocol**

	Partner 1	Partner 2	Partner 4	Partner 5	Partner 6	Partner 7
Updating of content after transmission (see section 5.6)	No	No	No	No	No	No
Multiplexed CAN messages (see section 4.1)	Yes	Yes	Yes	Yes	Yes	Yes
Signal Coding (see section 4.2)	Motorola	Motorola	Intel (Autosar)	Motorola	tbd	Motorola
Position Alternatives (see section 6.4)	(Yes)	(Yes)	(Yes)	No	(Yes)	(Yes)
Position Confidence (see	Yes	No	(Yes)	No	(Yes)	YES

section 4.4)						
Retransmission Request = Restart Request (see section 5.4)	Yes	Yes	No	No	No	No
Filling of "free" CAN slots (see section 6.2)	Invalid message	Meta data message	Message retransmission	None	Message retransmission	None
Sending of unknown profile data	Unknown message	Unknown message	Unknown message	Unknown message	Unknown message	Unknown message
Segments longer than 8190m	Split internally	Split internally	No-change message (see section 6.1)	No-change message (see section 6.1)	Limitation of maximum length	Limit to maximum length
ADASIS v2 Mini (see section 5.8)	No	Yes	Yes	Yes	No	Yes
ADASIS Long Horizon (see section 8.3)	No	No	Yes	No	No	No
Increasing Offsets based on ... (see ch 3.4.1)	Profile Type	Message Type	Profile Type	Message Type	Profile Type	Message Type
User specific profiles (see section 6.7)	Yes	Yes	Yes	Yes	Yes	Yes
Control Points (see section 5.7.2)	No	(Yes)	No	No	No	(Yes)

Restricted links (horizon goes into oneway against driving direction)	(Yes)	(Yes)	No	No	tbd	(Yes)
Single/Multipath (see section 3.3)	Full Horizon	Full Horizon	Main Path with Stubs and upto 3 Sub Paths of shorter lengths at each time	Main Path with Stubs	Full Horizon	Main Path with Stubs

## ***10 Appendix – Details for Certain Profiles***

### **10.1 Curvature Interpretation**

The curvature profile is modelled as a piecewise linear function. The curvature values changes by fixed steps which are constant within in 16 consecutive blocks of 64 bit span each (see coloured blocks in Table 31).

High curvature values for sharp curves (i.e. small radius) are coded with a low resolution whereas low curvature values (i.e. big radius) are coded with a high resolution. The  $\Delta R$  value in the following tables illustrates the effect on resolution of the radius.

The coding is based on simple functions with power of 2 factors to allow easy calculation also on simple processing units without floating point calculation capability.

The piecewise linear functions are defined by a set of simple systematic formula. Negative values shall be mirrored by the curvature value 0. Positive values represent right curves, negative values left curves. Left curves with curvature  $< -0.16192$  shall be coded with value 0, right curves with curvature  $> 0.16192$  shall be coded with value 1022. Value 1023 means "unknown".

Some values at the very high and very low end of the radius scale are not very meaningful from application point of view, but are defined in the table to follow the regular coding scheme.

**Table 31: Curvature Interpretation (10-bit)**

<i>Message Value</i>	<i><math>\Delta c</math> [1/m]</i>	<i>Curvature c [1/m]</i>	<i>Radius R [m]</i>	<i><math>\Delta R</math> [m]</i>
<i>0...510 negative value range symmetrical to range 512 - 1022</i>				
511	0.00001	0.00000	$\infty$	$\infty$
512		0.00001	100000.000	50000.000
513		0.00002	50000.000	16666.667
...		...	...	...
573		0.00062	1612.903	25.602
574	0.00002	0.00063	1587.302	24.802
575		0.00064	1562.500	47.348
576		0.00066	1515.152	44.563
...		...	...	...
637		0.00188	531.915	5.599
638	0.00004	0.00190	526.316	5.482
639		0.00192	520.833	10.629

640		0.00196	510.204	10.204
...			...	
701		0.00440	227.273	2.048
702		0.00444	225.225	2.011
703		0.00448	223.214	3.916
704		0.00456	219.298	3.781
...	0.00008		...	
765		0.00944	105.932	0.890
766		0.00952	105.042	0.875
767		0.00960	104.167	1.708
768		0.00976	102.459	1.653
...	0.00016		...	
829		0.01952	51.230	0.417
830		0.01968	50.813	0.410
831		0.01984	50.403	0.800
832		0.02016	49.603	0.775
...	0.00032		...	
893		0.03968	25.202	0.202

894		0.04000	25.000	0.198
895		0.04032	24.802	0.388
896		0.04096	24.414	0.376
...	0.00064		...	
957		0.08000	12.500	0.099
958		0.08064	12.401	0.098
959		0.08128	12.303	0.191
960		0.08256	12.112	0.185
...	0.00128		...	
1021		0.16064	6.225	0.049
1022		> 0.16192	< 6.176	
1023	-	unknown	unknown	-

The coding based on curvature values  $c$  leads to an Integer representation having a resolution of  $10^{-5}$  [1/m]. All coding and decoding calculation involves only add, subtract or bit-shift operations. Especially decoding can therefore be handled by simple controller units not having an integrated Floating-Point-Unit.

#### 10.1.1 Coding method

Notes:

- $c$  is the *curvature* value in 1/m units to be represented

- *value* is the coded curvature, represented by a 10 bit integer value
- $|x|$  represents the absolute value of  $x$
- $\text{SIGN} ( c )$  represents the value -1 or +1 when *curvature*  $< 0$  or  $\geq 0$ , respectively
- $\text{ROUND} (x)$  represents the (signed) integer closest to  $x$

```
if  $|c| < 0.00064$ 
    then  $value = 511 + \text{ROUND} ( c * 100000 )$ 
if  $0.00064 \leq |c| < 0.00192$ 
    then  $value = 511 + \text{SIGN} ( c ) * 32 + \text{ROUND} ((c * 100000) / 2)$ 
if  $0.00192 \leq |c| < 0.00448$ 
    then  $value = 511 + \text{SIGN} ( c ) * 80 + \text{ROUND} ((c * 100000) / 4)$ 
if  $0.00448 \leq |c| < 0.00960$ 
    then  $value = 511 + \text{SIGN} ( c ) * 136 + \text{ROUND} ((c * 100000) / 8)$ 
if  $0.00960 \leq |c| < 0.01984$ 
    then  $value = 511 + \text{SIGN} ( c ) * 196 + \text{ROUND} ((c * 100000) / 16)$ 
if  $0.01984 \leq |c| < 0.04032$ 
    then  $value = 511 + \text{SIGN} ( c ) * 258 + \text{ROUND} ((c * 100000) / 32)$ 
if  $0.04032 \leq |c| < 0.08128$ 
    then  $value = 511 + \text{SIGN} ( c ) * 321 + \text{ROUND} ((c * 100000) / 64)$ 
if  $0.08128 \leq |c| < 0.16192$ 
    then  $value = 511 + \text{SIGN} ( c ) * 384 + \text{ROUND} ((c * 100000) / 128)$ 
if  $0.16192 \leq |c|$ 
    then  $value = 511 + \text{SIGN} ( c ) * 511$ 
```

### 10.1.2 Decoding method

Notes:



- *value* is the 10-bit coded curvature value
- a *value* of 1023 signifies 'curvature unknown'
- $C = \text{value} - 511$  (signed Integer)
- $\text{SIGN}(C)$  represents the value -1 or +1 when  $C < 0$  or  $C \geq 0$ , respectively

if  $|C| \leq 64$  then  $\text{curvature} = C / 100000$

if  $64 < |C| \leq 128$  then  $\text{curvature} = 2 * (C - \text{SIGN}(C) * 32) / 100000$

if  $128 < |C| \leq 192$  then  $\text{curvature} = 4 * (C - \text{SIGN}(C) * 80) / 100000$

if  $192 < |C| \leq 256$  then  $\text{curvature} = 8 * (C - \text{SIGN}(C) * 136) / 100000$

if  $256 < |C| \leq 320$  then  $\text{curvature} = 16 * (C - \text{SIGN}(C) * 196) / 100000$

if  $320 < |C| \leq 384$  then  $\text{curvature} = 32 * (C - \text{SIGN}(C) * 258) / 100000$

if  $384 < |C| \leq 448$  then  $\text{curvature} = 64 * (C - \text{SIGN}(C) * 321) / 100000$

if  $448 < |C| \leq 511$  then  $\text{curvature} = 128 * (C - \text{SIGN}(C) * 384) / 100000$

## 10.2 Route Number Types

Table 32: Route numbers types for selected countries

Country	1	2	3	4	5	6
Andorra	European E#	Autopista A#	Carretera General CG#	Carretera Secundària CS#		
Australia	National Highway (1)  NI#,I#	National Route (2)  N#	State Route (3)  N#  State Highway (5)  S#	Metro Road (4)  X#,M#,MI#,F#	Metro Road RIC/MGM (6)  MI#  AI Road (8)  AI#	M-Road (7)  M#  A Road (9)  A#  B Road (10)  B#  C Road  C#

Country	1	2	3	4	5	6
						Tourist Roads are coded as named
Austria	European E#	Autobahnen A#	Schnellstraßen S#	Bundesstraßen B#	Landesstraßen L#  Landeshauptstraßen LH#	#
Bahrain	no route numbers exist					
Belgium	European E#	Autosnelwegen A#	Gewestwegen N#			B#  R#
Brazil	Federal BR-#	State SP-#				
Canada	Primary Provincial Routes  HWY-#	Secondary Provincial Routes  HWY-#	Regional  RR-#			

Country	1	2	3	4	5	6
	Autoroutes AUT-#	Autoroutes RTE-#	County CR-# District DR-#			
Czech Republic	European E#	Dálnice D#	Silnice I. tridy #	Silnice II. tridy #	Silnice III. Tridy #	
Denmark	European E#	Primærrute #	Sekundærrute #			
France	European E#	Autoroutes A#	Routes Nationales N#	Routes Départementales D#	Routes Communales C# Routes Vicinales V# Voies Rurales R#	B# M# T#

Country	1	2	3	4	5	6
Finland	European E#	Valtatie #	Kantatie #	Seututie #		
Germany	European E#	Autobahn A#	Bundesstraßen B#	Landesstraßen L# Staatsstraßen S# ST#	Kreisstraßen K#	F#
Ireland	Motorway M#	National N#	Regional R#			
Italy	European E#	Autostrada A#	Strada Statale SS#	Strada Provinciale SP#		
Kuwait	numbers only					
Liechtenstein	European E#	Autostraßen A#	Hauptstraßen #			
Luxembourg	European	Autoroutes	Routes National	Chemins Repris		

Country	1	2	3	4	5	6
	E#	A#	N#	CR#		
Monaco	European  E#	Autoroutes  A#	Routes Nationales  N#	Routes Départementales  D#	Routes Communales  C#  Routes Vicinales  V#  Voies Rurales  R#	B#  M#  T#
Netherlands	European  E#	Autosnelwegen  A#	Nationale Wegen  N#			R#  S#
Norway	European  E#	Riksveg  #	Fylkeveg  #			
Oman	no route numbers exist					
Portugal	European  E#	Autoestrada  A#	Itinerário Principal  IP#	Itinerário Complementar  IC#	Estrada Nacional  N#	Estrada Municipal  M#

Country	1	2	3	4	5	6
San Marino	European E#	Autostrada A#	Strada Statale SS#	Strada Provinciale SP#		
Saudi Arabia	2 digit number #	3 digit number #	4 digit number #			
Singapore	no route numbers exist					
Slovak Republic	European E#	Diaľnice D#	Cesty I. triedy #	Cesty II. triedy #	Cesty III. triedy #	
South Africa	National N#	Regional R#	Metro M#			
Spain	European E#	Autopista A#	Carretera Nacional N#	Carretera Comarcal C#	Carretera Local Provincial 1 st letter of Province + P#  Carretera Local Vecinal 1 st or 2 nd letters of Province + V#	
Sweden	European E#	Riksväg #10-99	Länsväg #100-499			

Country	1	2	3	4	5	6
Switzerland	European E#	Autostraßen A#	Hauptstraßen #			
Taiwan	Federal (1 digit) #	Provincial (2 digits) #	3 digits #			
UK	European E#	Motorways M#	A-Roads A#	B-Roads B#		
United Arab Emirates	E E#	Dubai D#	Sharjah S#			
US	Interstate I-#	Federal US-#	State	County CR-#		
Vatican City	European E#	Autostrada A#	Strada Statale SS#	Strada Provinciale SP#		



### 10.3 Heading Change Definition

Heading Changes are changes in direction along a path. If the road geometry is represented as poly-line, Heading Changes can be calculated by taking 3 consecutive shape points and calculating the line segment angles between the first and the second shape point and the second and third respectively. The angle between the first and the second line segment is called Heading Change.

Following figures illustrate Heading Changes:

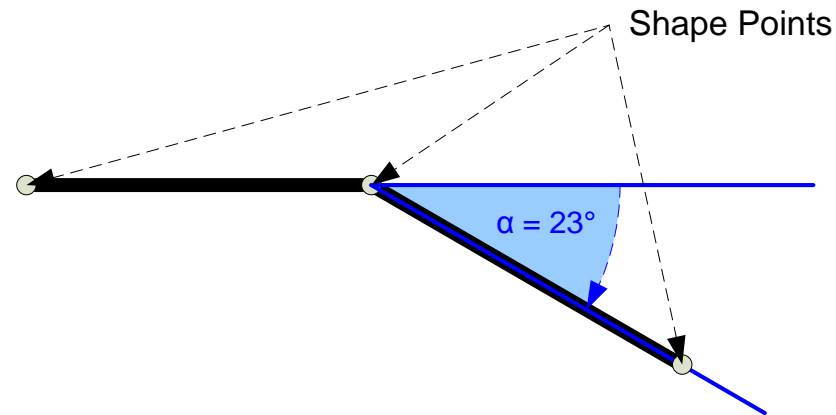
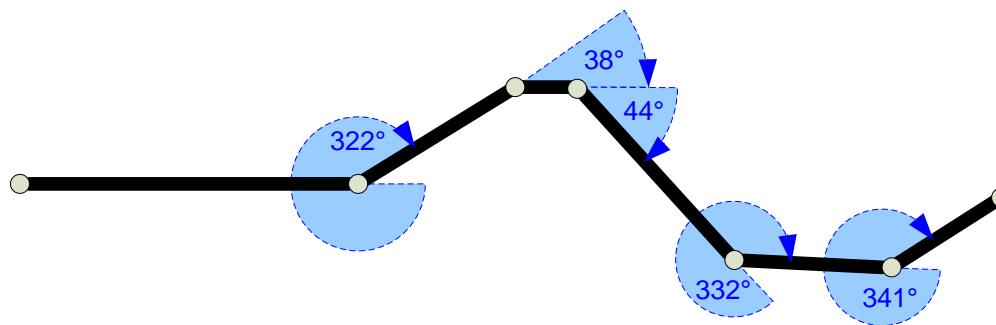


Figure 32: Calculation of Heading Change angle



**Figure 33: Heading Changes along a Path**

## 10.4 Coding Scheme for “Region Code”

The value in the field “Region Code” is based on ISO 3166-2:2007 “Codes for the representation of names of countries and their subdivisions - Part 2: Country subdivision code”. This standard defines codes for all relevant administrative subdivisions of all countries. The country subdivision code consists of two parts “CC-SSS”: The first part “CC” is the alpha-2 code element from ISO 3166-1 representing the country and is already covered in the META-DATA message by field “Country Code”, the second part “SSS” represents the top-level subdivision of this country.

The subdivision code part “SSS” uses up to three alphanumeric characters. To represent this code a field of length 24 bits assuming 8 bits per character is needed. All subdivision codes defined by ISO 3166-2 use either digit characters (“0”...“9”) or uppercase alphabetic characters (“A”...“Z”). To minimize the needed field length, each subdivision code character can be represented by the codes 1...26, a missing character, a space character or the digit “0” can be represented by code 0. This code range can be represented by 5 bits according to Table 33 assuming that the subdivision code part of ISO 3166-2 is either completely numeric or completely alphabetic to make the resulting code unique. A review of the ISO 3166-2 document showed, that currently all countries uses either a completely numeric or a completely alphabetic scheme for the top-level regions.

The value of the field “Region Code” is encoded using the following mapping scheme:

**Step 1:** The upto three characters of the subdivision code part of ISO 3166-2 are first aligned and padded to a code of three characters: If all ISO 3166-2 code characters are digits, they are right-aligned and padded with “0” on the left. In all other case (i.e. the ISO 3166-2 code characters contain at least one alphabetic character), they are left-aligned and filled with space characters on the right. (The different alignment helps to avoid code clashes if numeric and alphabetic subdivision codes exist for the same country.)

**Step 2:** The resulting three characters are coded according to Table 33. The three 5 bit codes are packed in the region code field of 15 bits length by the formula  $((code_{char1} * 32 + code_{char2}) * 32 + code_{char3})$ , i.e., by shifting the code of the first character by 10, the code of the second character by 5, the code of the third character by 0 bits to the left.

The value of the field “Region Code” is decoded accordingly. Because the encoding of the alphabetic characters “A”...”J” and the numeric characters “1”...”9” uses the same 5 bit codes, the decoded subdivision code can be ambiguous in very rare cases.

### Encoding Examples:

ISO 3166-2 subdivision code of “Hessen” in “Germany”: “DE-HS” – “DE” is represented by a country code of 276 (ISO 3166-1 numeric) and the padded subdivision code “HS\_” is encoded by the region code of  $((8_{for\ “H”} * 32 + 20_{for\ “S”}) * 32 + 0_{for\ Space}) = 8832$ .

ISO 3166-2 subdivision code of “Vorarlberg” in “Austria”: “AT-8” – “AT” is represented by a country code of 40 and “008” is encoded by the region code of  $((0_{for\ “0”} * 32 + 0_{for\ “0”}) * 32 + 8_{for\ “8”}) = 8$ .

ISO 3166-2 subdivision code of “Alsace” in “France”: “FR-A” – “FR” is represented by a country code of 250 and “A\_\_” is encoded by the region code of  $((1_{for\ “A”} * 32 + 0_{for\ Space}) * 32 + 0_{for\ Space}) = 8192$ .

ISO 3166-2 subdivision code of département “Ain” in “France”: “FR-01” – “FR” is represented by a country code of 250 and “001” is encoded by the region code of  $((0_{for\ “0”} * 32 + 0_{for\ “0”}) * 32 + 1_{for\ “1”}) = 1$ .

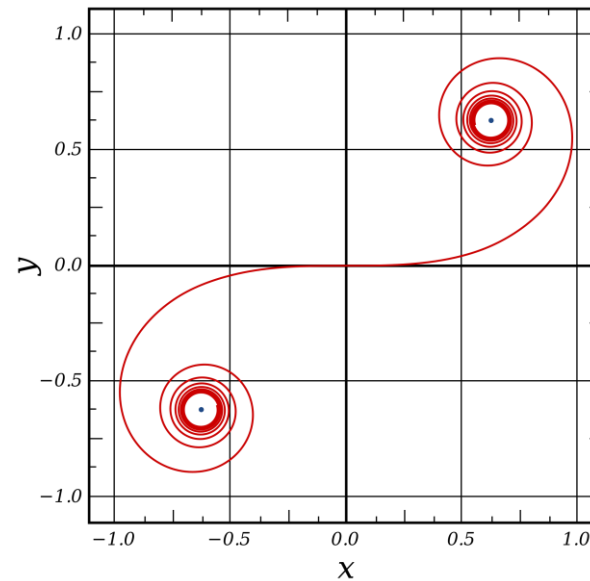
<i>ISO 3166-2 alphabetic character</i>	<i>ISO 3166-2 numeric character</i>	<i>5 bit code in field “Region Code”</i>
<i>none/Space</i>	<i>none/Space/0</i>	0
<i>A</i>	<i>1</i>	1

<i>ISO 3166-2 alphabetic character</i>	<i>ISO 3166-2 numeric character</i>	<i>5 bit code in field "Region Code"</i>
<b><i>B</i></b>	<b><i>2</i></b>	2
<b><i>C</i></b>	<b><i>3</i></b>	3
<b><i>D</i></b>	<b><i>4</i></b>	4
<b><i>E</i></b>	<b><i>5</i></b>	5
<b><i>F</i></b>	<b><i>6</i></b>	6
<b><i>G</i></b>	<b><i>7</i></b>	7
<b><i>H</i></b>	<b><i>8</i></b>	8
<b><i>I</i></b>	<b><i>9</i></b>	9
<b><i>J</i></b>		10
<b><i>K ... Y</i></b>		11 ... 25
<b><i>Z</i></b>		26
		27...31 – not used

**Table 33: Coding Scheme for field "Region Code" to map each ISO 3166-2 alphanumeric character to a 5 bit code**

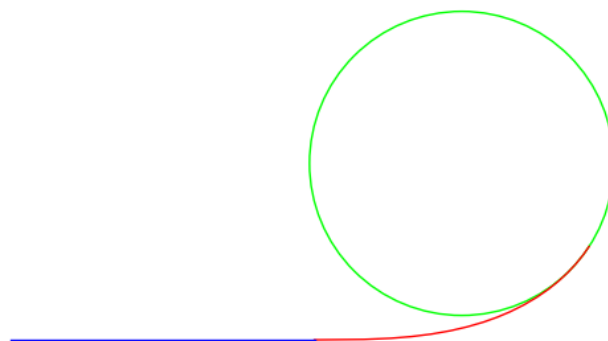
## 10.5 Clothoid Profiles

Clothoids (Figure 34) are 2-D parametric curves with the property that their curvature is a linear function of its arc length. The Clothoid is also known as a Cornu spiral.



**Figure 34: Unit clothoid**

In some systems, Clothoids are used for the modelling of road geometry instead of the classical poly-line approach. In this case, parts of the Clothoids are used to model parts of the road. In theory, an ideal curve consists of a straight part, a transition curve and a circular arc (Figure 35). Clothoids are ideal transition curves since their curvature is linearly increased along the clothoid.



**Figure 35 Straight (blue); transition curve (clothoid, red); circular arc (green)**

Unfortunately, the majority of roads are not built according to this ideal and in practice one needs a number of Clothoids to represent a single stretch of road.

#### **10.5.1 Clothoid for Relative Geometry**

If absolute road geometry (longitudes and latitudes) are not of interest to the application, relative clothoidal geometry of a single path can be realized only by using the standard curvature profile (see section 10.1). Since clothoid *are* defined as curves for which the curvature is a linear function of distance, this approach will perfectly describe the target clothoid. One must be aware, however, that in this case OFFSETs must also be expressed as real distance between points, rather than Euclidian distance between two 2-D points. If necessary, additional, system-specific profiles containing arc distances between offsets may be introduced.

Another issue can be the continuity of Clothoids when the previous one does not end where the next starts. If the end curvature of the first clothoid is equal to the start curvature of the next one, no additional work needs to be done. Otherwise, two profiles need to be used: one containing the clothoid start curvatures, and another with the clothoid end curvatures. This approach will allow for discontinuities in curvature profiles.

### 10.5.2 Clothoid for Absolute Geometry

Each Clothoid segment can be represented by the following parameters (Figure 36):

- End points  $P_s(lat, long), P_E(lat, long)$ :
- End curvatures  $\kappa_s, \kappa_E$  or end heading angles  $\theta_s, \theta_E$
- Length  $L$

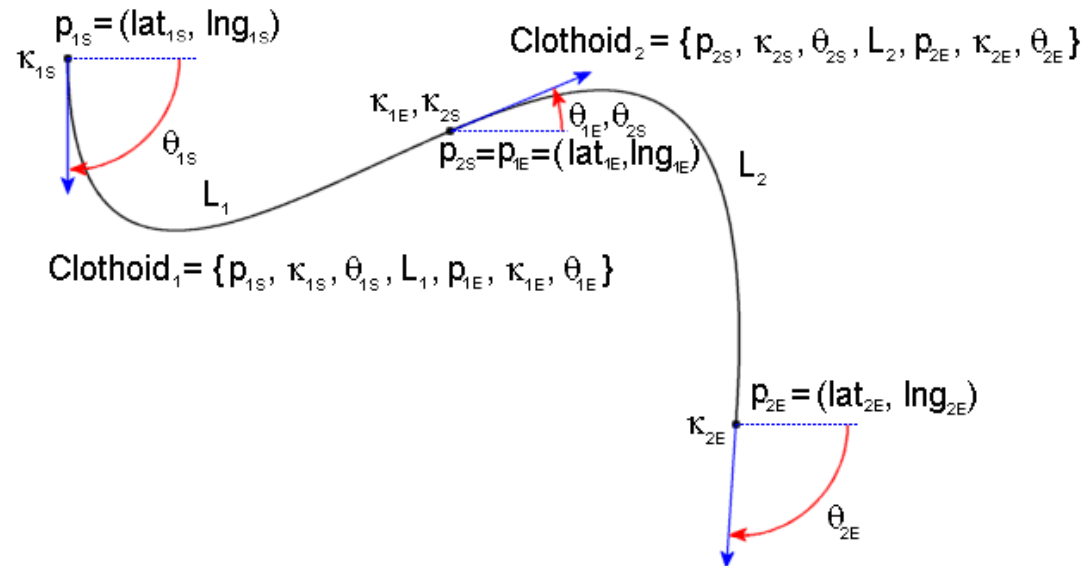


Figure 36: Clothoid Representation of Roads

Therefore, in addition to standard LONGITUDE/LATITUDE profiles, one needs to transfer three additional parameters:  $(\kappa_s, \kappa_E, L)$  or  $(\theta_s, \theta_E, L)$ . It will not usually be possible to achieve  $C^2$  continuity so  $\kappa_s$  of the second clothoid will not be equal to  $\kappa_E$  of the first one. In other words, five numbers will be necessary to describe one part of the road using the clothoid model (assuming that one clothoid ends where the next one starts).

A consistent method of transferring clothoid data using this protocol is:

- Use of standard LONGITUDE/LATITUDE profiles;
- Define additional CLOTHOID PROFILE LONG. The spot value of this profile is clothoid length  $L$ , but an additional two control points are required:  $\kappa_s, \kappa_E$  or  $\theta_s, \theta_E$ .

## 10.6 Cubic Bézier Spline Profile

For some systems, Poly-line geometry of roads defined by shape points with Longitude and Latitude Profiles is sufficient. For others, resolution, range and overall characteristics of Curvature and/or Slope profiles may not be satisfactory. In that case, absolute and relative road geometry can be described using continuous functions such as Clothoids (section 10.5), B-splines or Cubic Bézier Splines.

Some map providers offer an alternative road geometry definition in form of Cubic Bézier Splines. To transfer those data using ADASIS v2 protocol, PROFILE LONG of types 1, 2, 3, 4, 5 and 6 are used to define a Cubic Bézier Spline Profile.

A Bézier curve is defined by its control points  $P_i$ :

$$B(t) = \sum_{i=0}^n P_i b_{i,n}(t) \quad \text{where} \quad b_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, \dots, n$$

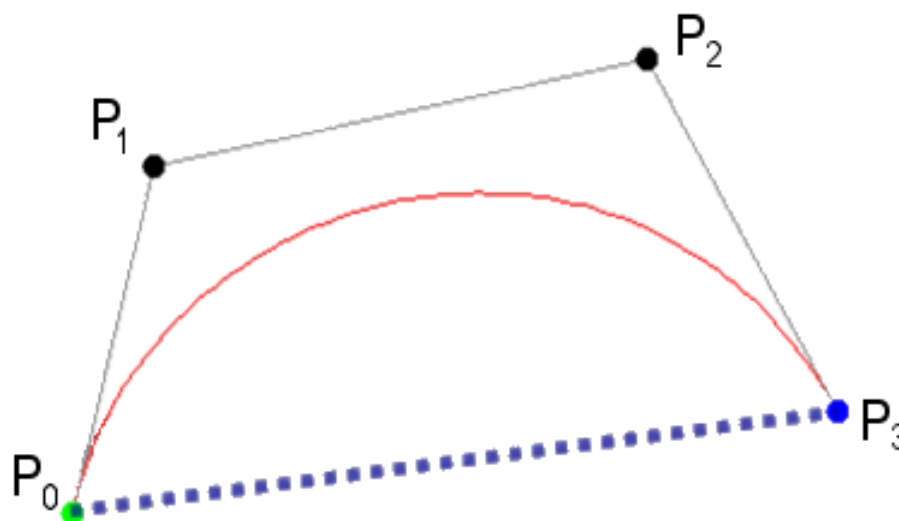
where  $b_{i,n}(t)$  are the Bernstein polynomials.

A Cubic Bézier curve is defined by its 4 control points  $[P_0, P_1, P_2, P_3]$ :

$$B(t) = \sum_{i=0}^3 P_i b_{i,3}(t) = (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t) t^2 P_2 + t^3 P_3, \quad t \in [0,1]$$

In case of a digital map, each control point is actually triple consisting of Longitude, Latitude and Altitude.





**Figure 37: Cubic Bézier Spline (red) and its control points**

As we can see from Figure 37, control points  $P_0$  and  $P_3$  are indeed standard geometry shape points that can be transferred using usual Longitude, Latitude and Altitude Profiles (PROFILE LONG types 1,2 and 3). To transfer  $P_1$  and  $P_2$  one need to use (Bézier) Control Point Longitude, Latitude and Altitude Profiles (PROFILE LONG types 4, 5 and 6).

- $P_0$  longitude, latitude and altitude PROFILE LONG messages are characterized by same offset  $X_0$ ;
- $P_3$  longitude, latitude and altitude PROFILE LONG messages are characterized by same offset  $X_3$ ;
- $P_1$  Bézier Control Point longitude, latitude and altitude PROFILE LONG messages are characterized by same offset  $X_1$ ;
- $P_2$  Bézier Control Point longitude, latitude and altitude PROFILE LONG messages are characterized by same offset  $X_2$ ;
- It must be  $X_0 < X_1 < X_2 < X_3$ . In ideal case, difference  $X_3 - X_0$  is equal to the length of the Bézier curve arc. Offsets  $X_1$  and  $X_2$  do not have other meaning or purpose except as indication of order of Control Points.

If ADASIS v2 Horizon Providers publishes all above values, a client can choose

- To use  $P_0$  and  $P_3$  longitude and latitudes, getting classical 2-D poly-line geometry;
- To use  $P_0$  and  $P_3$  longitude, latitudes and altitudes, getting classical 3-D poly-line geometry;
- To use  $P_0$  and  $P_3$  longitude and latitudes together with  $P_1$  and  $P_2$  control longitudes and control latitudes describing high-resolution 2-D Cubic Bézier road geometry;
- To use all data that defines full 3-D Cubic Bézier road geometry.

In the same manner, road geometry described in form of Quadratic Bézier splines can be represented in the ADASIS v2 Protocol. In that case, only a single Bézier Control Point  $P_1$  between end points  $P_0$  and  $P_3$  is defined. Bézier splines of higher orders are defined by having more than two Bézier Control points.

Please note that a single path will consist of several Bézier splines. Since the first point of the next spline is the same as the last point of the previous one, it needs to be transferred only once.

## 10.7 Link Identifier Profile

A Link Identifier Profile is used to uniquely identify parts of the path during multiple executions of the application. Besides the possibility to identify common segments of different paths, this feature may be used by the application to permanently store data related to specific roads. For instance, the application may need to store information associated to the link with identifier 34567. This information will be available for all current and subsequent paths that contain the link in question (see Figure 38).

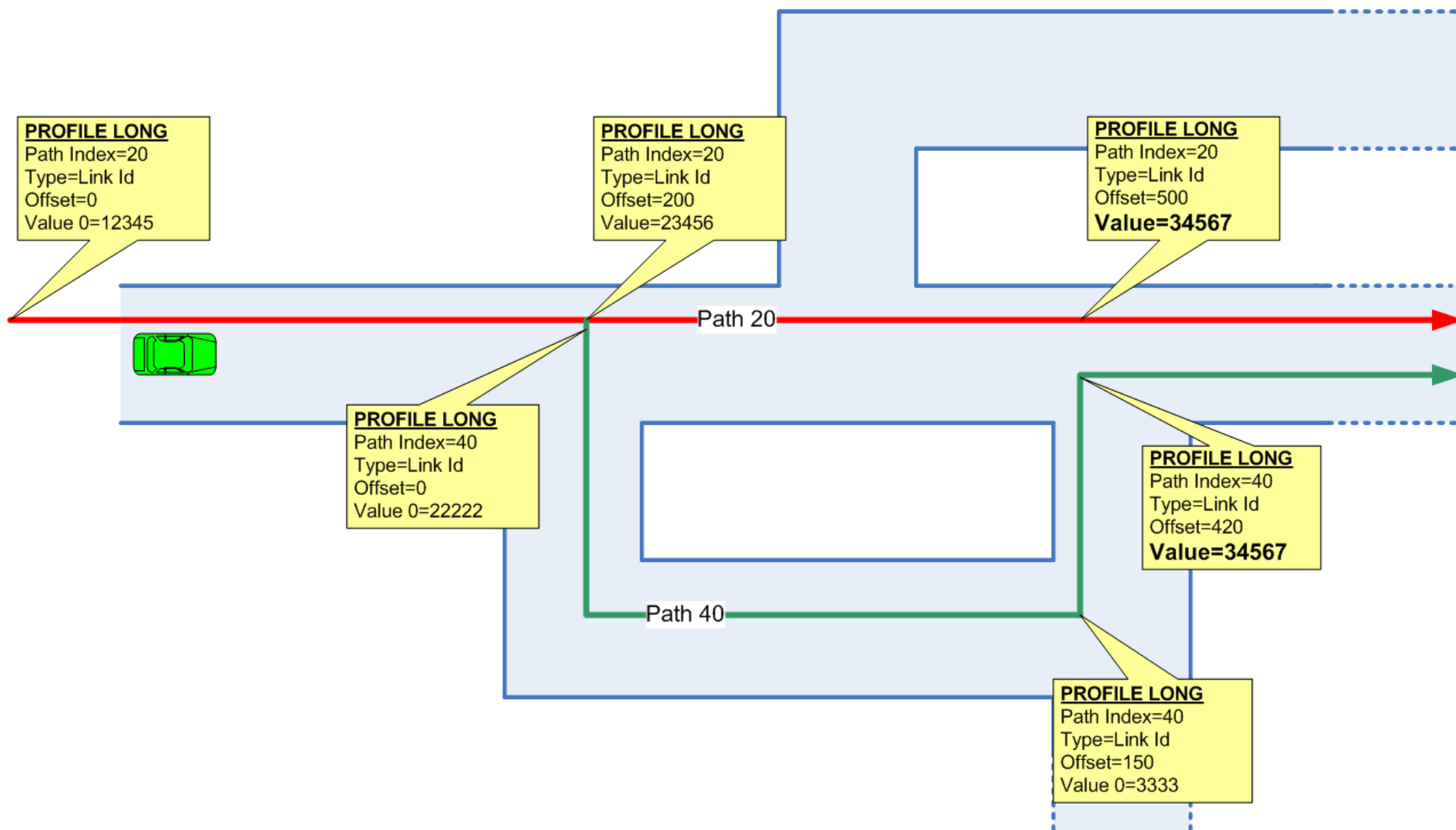


Figure 38: Permanent Link Id Profile

Internally to ADASIS v2 Horizon Provider, this profile may correspond to the database links that build up a single path.

GDF, which is a standard interchange format for map data, includes an object feature called *Published Version Identifier* (PVID). For the database links, we can speak of a *Link Identifier*. This identifier is unique in a single map and does not change between map releases.

The GDF PVID field can theoretically hold identifiers in a range between 0 and 9,999,999,999 (0x2540BE3FF), which is outside of range of 0 to 4,294,967,294 (0xFFFFFFFF) that fits into the dedicated PROFILE LONG message.

Currently (status December 2010), for example the NAVTEQ Map maximum PVID world-wide is around 900,000,000 – well within 32-bit identifier range. Taking into account the current rate of map coverage expansion and change rate one can expect that 32-bit range will be exhausted in few years.

In other words, contemporary systems can use GDF PVID as source of data for the Link Identifier Profile. To achieve uniqueness, future ADASIS v2 Horizon Providers may use combination of Link Identifier Profile and the Country Code from META-DATA message. Alternatively, new PROFILE LONG type may be defined that will contain additional 32-bit of the Identifier. Of course, those two PROFILE LONG will contain same Path Index and same Offset.

Map Databases that do not include GDF PVID will take some other identifier as ADASIS v2 Link Identifier.

## 10.8 Traffic Sign and Situation Profile

The “traffic sign and situation profile” conveys information primarily about roadside traffic signs, but also about situations that are equivalent to a traffic sign without an actual sign being visible, and about some special situations, e.g. speed bumps, that can be handled in a similar way.

Traffic sign information is expected to be used both for general ADAS purposes (where the meaning of the sign is of importance) and to augment camera based traffic sign recognition (where the visual appearance of the sign is of importance).

Traffic signs use the PROFILE\_LONG message with the following decomposition of the Value field:

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Description</i>
<b>Sign Type</b>	8	0..254	See Table 35: Traffic signs and situations
<b>Value</b>	8	0..254 255	Depending on sign, see Table 35. 0 if the traffic sign does not have a value. A value of 255 is used in update messages to invalidate or delete a sign.
<b>Lane</b>	4	0 1..14 15	The lane that this sign is valid for. For signs above the road and on the road surface, also the lane above/on which the sign is located. (There are cases where signs to the right or left of the road are valid for individual lanes, see E1 in the Vienna Convention [16].) 0: Not specific to any lane. 1..14: lane number 15: Unknown, or N/A
<b>Vehicle Specific</b>	2	0..3	This field marks a sign that is valid only for specific vehicles. This includes signs valid only for vehicles with specific dimensions or loads: 0: Not vehicle specific. 1: Applies to this vehicle but not to all vehicles. 2: Does not apply to this vehicle but possibly to other vehicles. 3: It's not known whether the sign applies to this vehicle.
<b>Time Specific</b>	2	0..3	This field marks a sign that is valid only on specific days or at specific times: 0: Not time dependent. 1: Applies now but not all the time. 2: Does not apply now but at other times. 3: It's not known whether the sign applies now.

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Description</i>
<b>Condition</b>	4	0..15	<p>This field describes other conditions and dependencies. Time and vehicle type/load dependencies are not taken into account here.</p> <p>0: Always valid.  1: Has other conditions that are fulfilled for all vehicles.  2: Has other conditions that are fulfilled for this vehicle but not necessarily for all vehicles.  3: Has other conditions that are not fulfilled for this vehicle but possibly for other vehicles.  4: Has other conditions that are not currently fulfilled for any vehicle.  5: Sign is not valid at all (e.g. visible here but referring to another road).  6: Valid only if snow.  7: Valid only if fog.  8: Valid only if rain.  9: Valid only by day.  10: Valid only by night.  11: Valid depending on turn direction / future path.  12: Validity depends on conditions or combinations of conditions that cannot be adequately described by other values, and it is not known whether the sign applies.  13: Reserved.  14: Reserved.  15: Unknown whether validity depends on conditions.</p> <p>If validity depends on snow, fog, or rain conditions, turn direction, day/night, or other conditions, and it is known whether the condition is fulfilled, then values 1 or 4 should be used, but not values 6 to 12. In other words, values 0 to 5 are used in cases where the Horizon Provider has sufficient information to decide about sign validity, while values 6 to 12 imply that the driver (or another system which is a user of the information) needs to decide.</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Description</i>
<b>Sign Location</b>	3	0..6 7	0: Unknown 1: Right of the road 2: Left of the road 3: Above the road 4: On the road surface 5: Sign is not physically located at this position, see remark 1 6: The stretch of road that a sign applies to ends at this position, see remark 1 7: N/A

**Table 34: Traffic Sign and Situation: Value (32 bit)**

**Remark 1:**

A sign may be physically located at a position different from the position it applies to. Typically this is marked using additional panel H,1. In this case it is recommended to transmit the sign marked with Condition = Not Valid At All at the position where it is physically located, and to transmit it again with Sign Location = 5 (not physically located here) at the position it applies to.

If a sign is marked by additional panel H,2 to apply only to some stretch of road, for the end of this stretch of road a traffic sign message may be transmitted with Sign Location = 6.

There may be special situations equivalent to a traffic sign, e.g. in Germany vehicles turning from a forest road onto a major road have to yield to traffic on the major road. The horizon provider may transmit a GIVE WAY sign marked with Sign Location = 5 (not physically located here). The system specification should explicitly list the cases where a traffic sign message can be transmitted that does not correspond to any physical sign.

For the list of traffic signs, the international standard referenced below is the Vienna Convention on Road Signs and Signals published by the UN, with the European Supplement by the UN ECE [16].

<i>Traffic sign or situation</i>	<i>Vienna Convent ion</i>	<i>Number</i>	<i>Remarks</i>
<b>LEFT BEND</b>	A1,a	10	
<b>RIGHT BEND</b>	A1,b	9	
<b>DOUBLE BEND LEFT FIRST</b>	A1,c	14	
<b>DOUBLE BEND RIGHT FIRST</b>	A1,d	13	
<b>STEEP DESCENT</b>	A,2	68	If value is not 0, then it is the grade in percent.
<b>STEEP ASCENT</b>	A,3	67	If value is not 0, then it is the grade in percent.
<b>CARRIAGEWAY NARROWS</b>	A4,a	40	
<b>CARRIAGEWAY NARROWS LEFT</b>	A4,b	42	
<b>CARRIAGEWAY NARROWS RIGHT</b>	A4,b reversed	41	
<b>RIVER BANK</b>	A,6	29	
<b>RIVER BANK LEFT</b>	A,6 reversed	30	
<b>UNEVEN ROAD</b>	A,7a	69	
<b>HUMP</b>	A7,b	70	



<i>Traffic sign or situation</i>	<i>Vienna Convention</i>	<i>Number</i>	<i>Remarks</i>
<b>DIP</b>	A7,c	71	
<b>SLIPPERY ROAD</b>	A,9	66	
<b>FALLING ROCKS RIGHT</b>	A,11	62	
<b>FALLING ROCKS LEFT</b>	A,11 reversed	61	
<b>PEDESTRIAN CROSSING</b>	A,12	51	
<b>CHILDREN</b>	A,13	52	
<b>CYCLISTS</b>	A,14	54	
<b>DOMESTIC ANIMALS CROSSING</b>	A,15a	4	
<b>WILD ANIMALS CROSSING</b>	A,15b	5	
<b>ROAD WORKS</b>	A,16	6	
<b>LIGHT SIGNALS</b>	A,17	31	Traffic sign warning of light signals – not the signals themselves (see below)!
<b>INTERSECTION</b>	A,18	35	
<b>INTERSECTION WITH MINOR ROAD</b>	A,19	36	
<b>TWO-WAY TRAFFIC</b>	A,23	55	

<i>Traffic sign or situation</i>	<i>Vienna Convention</i>	<i>Number</i>	<i>Remarks</i>
<b>TRAFFIC CONGESTION</b>	A,24	75	
<b>RAILWAY CROSSING WITH GATES</b>	A,25	56	
<b>RAILWAY CROSSING WITHOUT GATES</b>	A,26	57	
<b>TRAMWAY</b>	A,27	59	
<b>RAILWAY CROSSING</b>	A,28	58	
<b>DANGER</b>	A,32	0	
<b>GIVE WAY</b>	B,1	32	
<b>STOP</b>	B,2	33	
<b>PRIORITY ROAD</b>	B,3	34	
<b>PRIORITY FOR ONCOMING TRAFFIC</b>	B,5	82	
<b>PRIORITY OVER ONCOMING TRAFFIC</b>	B,6	81	
<b>OVERTAKING PROHIBITED</b>	C,13aa C,13ab	46	
<b>OVERTAKING BY GOODS VEHICLES PROHIBITED</b>	C,13ba C,13bb	20	Goods vehicles above 3.5 tons, unless stated otherwise on an additional panel
<b>SPEED LIMIT</b>	C,14	87	Value encodes speed as in the SEGMENT message (see table 12).

<i>Traffic sign or situation</i>	<i>Vienna Convention</i>	<i>Number</i>	<i>Remarks</i>
<b>END OF ALL PROHIBITIONS</b>	C,17a	79	
<b>END OF SPEED LIMIT</b>	C,17b	88	Value encodes speed as in the SEGMENT message (see table 12).
<b>END OF PROHIBITION ON OVERTAKING</b>	C,17c	47	
<b>END OF PROHIBITION ON OVERTAKING FOR GOODS VEHICLES</b>	C,17d	21	
<b>DIRECTION TO THE RIGHT</b>	D,1	38	
<b>DIRECTION TO THE LEFT</b>	D,1 reversed	39	
<b>PASS RIGHT SIDE</b>	D,2	3	
<b>PASS LEFT SIDE</b>	D,2 reversed	2	
<b>BEGINNING OF BUILT-UP AREA</b>	E,7	77	
<b>END OF BUILT-UP AREA</b>	E,8	83	
<b>TUNNEL</b>	E,11a	24	
<b>RESIDENTIAL AREA</b>	E,17a	7	In European Supplement.
<b>END OF RESIDENTIAL AREA</b>	E,17b	8	In European Supplement.

<i>Traffic sign or situation</i>	<i>Vienna Convention</i>	<i>Number</i>	<i>Remarks</i>
<b>LANE MERGE RIGHT</b>	G,12a	44	
<b>LANE MERGE LEFT</b>	G,12a reversed	43	
<b>Railroad crossing</b>		251	Not a sign but the location of the actual railroad crossing
<b>Zebra/pedestrian crossing</b>		252	Not a sign, but the actual Zebra crossing
<b>Speed bump</b>		253	Not a sign, but the actual speed bump
<b>Traffic light</b>		254	<p>The location of a traffic light.</p> <p>If the current phase of the traffic light is known, it can be transmitted as value:</p> <ul style="list-style-type: none"> <li>1: green</li> <li>2: red</li> <li>3: amber</li> <li>4: red + amber</li> <li>5: red flashing</li> <li>6: amber flashing</li> <li>7: dark</li> </ul>

**Table 35: Traffic signs and situations: Type**

If other traffic signs or situations not listed here are to be used in an ADASIS implementation, it is recommended to number them according to the NDS standard; ADASIS will strive to avoid numbering conflicts. For signs not covered by NDS, or for implementors without access to the NDS standard, it is recommended to start numbering them from 129 up.

Implementation specific signs should be listed in the system specification.

## 10.9 Truck Speed Limit Profile

Truck speed limits use the 32 bit long profile with the following decomposition:

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Profile Long Header</b>	32			See section 4.8.
<b>Speed</b>	8	0..253 254 255	km/h or mph	Speed Restriction in km/h or mph (depends on signal SPEED_UNIT in META-DATA message)  254 means speed >253 255 means N/A
<b>Weight</b>	8	0..126 127	0.5 tons	This signal indicates if the speed limit applies to vehicles with a maximum weight. Each value must be multiplied by 0.5 tons. The unit “tons” is region specific and can be either metric tons, long tons (UK) or short tons (US).  0: no weight restriction 1: 0.5 tons 2: 1 ton ... 125: 62.5 tons 126: >62.5 tons 127: N/A

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b>Hazardous Goods</b>	3	0..6 7		<p>Indicates if the speed limit applies to vehicles with hazardous goods</p> <p>0: Speed Limit applies for all vehicles independent of hazardous goods  1: Speed Limit applies for vehicles with explosive goods only  2: Speed Limit applies for vehicles with goods that are harmful for water  3..5: Reserved  6: unknown  7: N/A</p>
<b>Weather Restriction</b>	3	0..6 7		<p>Indicates if it is a weather dependent speed limit</p> <p>0: No weather restriction  1: Speed limit only valid if snow  2: Speed limit only valid if fog  3: Speed limit only valid if rain  4: Reserved  5: Reserved  6: Unknown  7: N/A</p>
<b>Type</b>	2	0..2		<p>Indicates the type of the speed limit</p> <p>0: Explicit legal speed limit given by a sign at the road  1: Implicit legal speed limit given by road traffic regulations  2: The maximum speed limit that is allowed without distinguishing if it is explicit or implicit  3: unknown or N/A</p>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Units</i>	<i>Description</i>
<b><i>Time Valid</i></b>	2	0..2		Indicates if time dependent speed limit is valid at this moment  0: Speed limit is not valid at this moment 1: Speed limit is valid at this moment 2: Unknown if it is valid at this moment 3: N/A
<b><i>Time Dependent</i></b>	1	0..1		Indicates if speed limit is time dependent  0: Speed limit is not time dependent 1: Speed limit is time dependent
<b><i>Reserved</i></b>	5			

**Table 36: Truck Speed Limit**

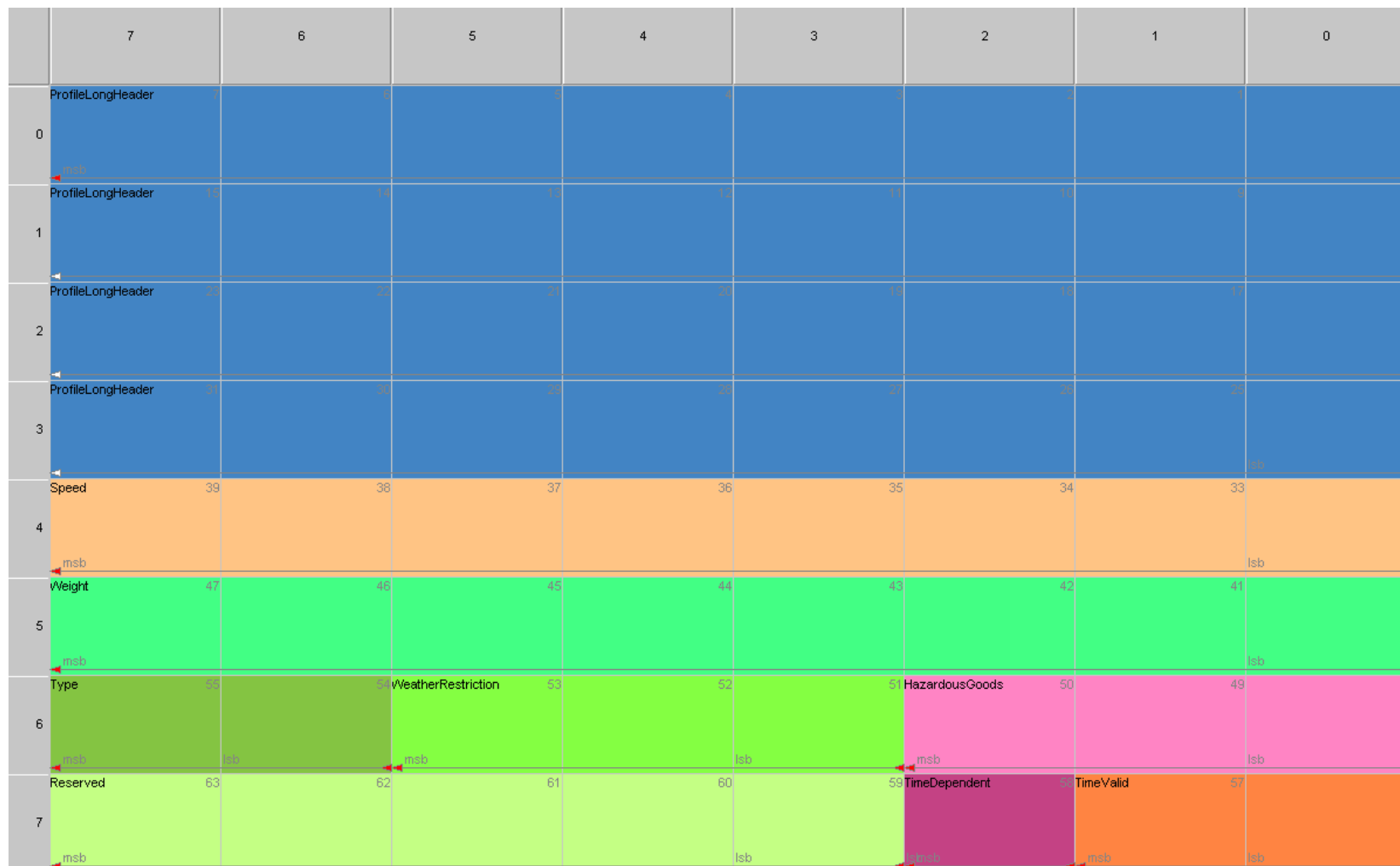


Table 37: CAN layout of LONG\_PROFILE Message "Truck Speed Limit"



## 10.10 Extended Lane Profile

In the SEGMENT message there is already a basic lane information about the number of lanes in and against driving direction. With this LONG PROFILE “Extended Lane” more detailed lane information especially about lane connectivity and lane markings can be transmitted.

“Extended Lane” uses the 32 bit long profile with the following decomposition:

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Description</i>
<b>Profile Long Header</b>	32		See section 4.8.
<b>Lane Number</b>	4	0 1..14 15	Number of the lane which is described in this message 0 is not allowed 15 means <i>N/A</i>
<b>First Predecessor Lane</b>	4	0 1..14 15	Number of first predecessor lane. (see section 10.10.1) 0 means no predecessor 15 means <i>N/A</i>
<b>Last Predecessor Lane</b>	4	0 1..14 15	Number of last predecessor lane. (see section 10.10.1) 0 means no predecessor 15 means <i>N/A</i>
<b>Priority Predecessor Lane</b>	4	0 1..14 15	Number of priority predecessor lane. (see section 10.10.1) 0 means no predecessor lane has priority 15 means <i>N/A</i>

<i>Field</i>	<i>Length (bits)</i>	<i>Value Range</i>	<i>Description</i>
<b>Arrow Marking</b>	8	bit field 255	<p>Bit coded field. If a bit is set, this lane has a road marking with an arrow into this direction.</p> <p>00000001: straight  00000010: slight right  00000100: right  00001000: hard right  00010000: U turn  00100000: hard left  01000000: left  10000000: slight left  ...  11111111: N/A</p>
<b>Lane Type</b>	4	0..15	Additional information about the lane type (e.g. exit lane, restricted lane, ...). see Table 38
<b>Line Marking</b>	3	0..7	Line marking for the inner side of this lane (i.e. left side for right-hand traffic). see Table 39
<b>Yield Flag</b>	1	0..1	<p>True if traffic on this lane has to yield to incoming traffic from a neighboring road when entering this segment (see section 10.10.1.9)</p> <p>0: default value</p>

#### 10.10.1 Lane Connectivity

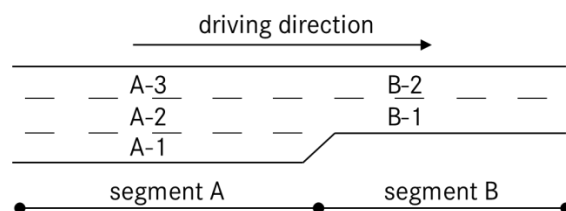
For each „Lane Number“ the three signals „First Predecessor Lane“, „Last Predecessor Lane“ and „Priority Predecessor Lane“ describe how this lane can be reached along a path.

This lane can be reached from any lane between the first and last predecessor lane on the previous segment along the path. If priority predecessor lane is set (i.e. it is not 0), traffic on that lane has priority over the other lanes driving into this lane. A predecessor connection means a straight forward or forced connection from one lane to the next. It does not mean a regular lane change.

All lanes are counted beginning with 1 at the outermost lane, i.e. for the most right lane for right-hand traffic and the most left lane for left hand traffic. If the hard shoulder is to be modeled, it gets lane number 1 and all other lane numbers are increased by 1.

The use of the lane connectivity signals is illustrated in the following examples. The driving direction arrow illustrates the path.

#### 10.10.1.1 Example: Lane ends



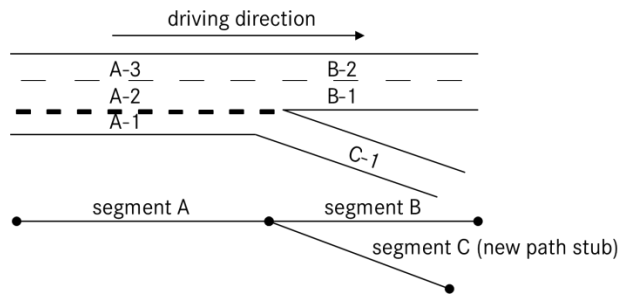
Lane B-2: first predecessor 3, last predecessor 3, priority predecessor 3

Lane B-1: first predecessor 1, last predecessor 2, priority predecessor 2

Note that it is also possible to drive from lane A-3 to lane B-1 but A-3 is NOT listed as a predecessor of B-1 because A-3 is a through-lane with lane markers continuing as lane B-2. This change would therefore be a regular lane change.

On the contrary, lane A-1 ends and one must continue on lane B-1. A-1 is therefore a predecessor of B-1. Traffic from A-1 must yield to traffic on A-2 so A-2 is the priority predecessor of B-1.

#### 10.10.1.2 Example: Exit Lane



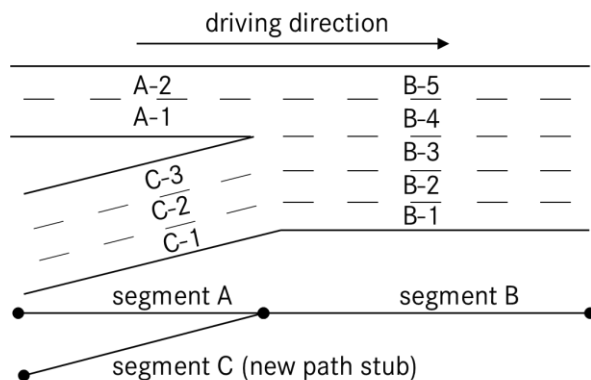
Lane B-2: first predecessor 3, last predecessor 3, priority predecessor 3

Lane B-1: first predecessor 2, last predecessor 2, priority predecessor 2

Lane C-1: first predecessor 1, last predecessor 1, priority predecessor 1

In contrast to 10.10.1.1, lane A-1 does not end but continues as lane C-1 on segment C. Therefore A-1 is not a predecessor of B-1.

#### 10.10.1.3 Example: Merging Highways

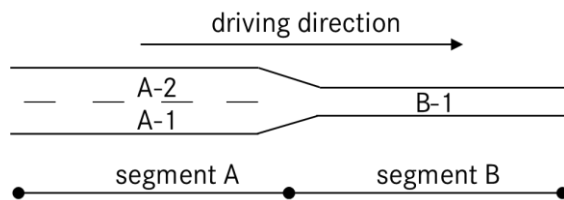


Lane B-5: first predecessor 2, last predecessor 2, priority predecessor 2

Lane B-4: first predecessor 1, last predecessor 1, priority predecessor 1  
 Lane B-3: first predecessor 0, last predecessor 0, priority predecessor 0  
 Lane B-2: first predecessor 0, last predecessor 0, priority predecessor 0  
 Lane B-1: first predecessor 0, last predecessor 0, priority predecessor 0

In this example, our vehicle moves from segment A to segment B, segment C is a stub along the path into which it is forbidden to enter. Along the path, only lanes B-4 and B5 have predecessors as the other 3 lanes are through-lanes coming from segment C. Segment C has no predecessor lanes on segment A as it is forbidden to turn from A to C.

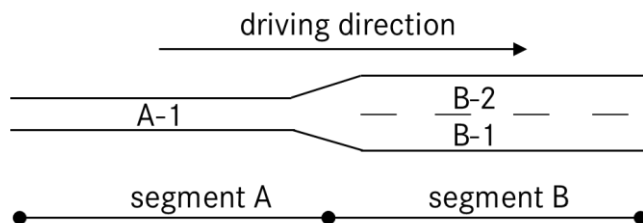
#### 10.10.1.4 Example: Merging Lanes without Priority



Lane B-1: first predecessor 1, last predecessor 2, priority predecessor 0

Two lanes form one lane without priority for the one or the other lane. Therefore, priority predecessor is 0.

#### 10.10.1.5 Example: Single Lane widens on both sides into Two Lanes

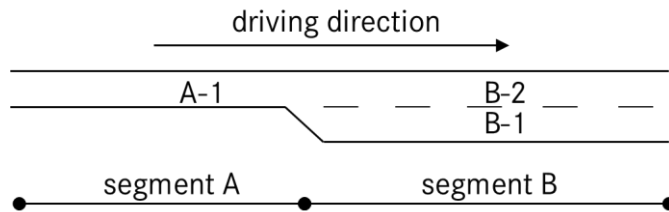


Lane B-2: first predecessor 1, last predecessor 1, priority predecessor 0

Lane B-1: first predecessor 1, last predecessor 1, priority predecessor 0

Priority predecessor is set to 0 for both lanes on segment B. This is to express that there is no straight-forward through lane. This is analogous to the merging lanes in 10.10.1.4.

**10.10.1.6 Example: New Lane on the Right Side**

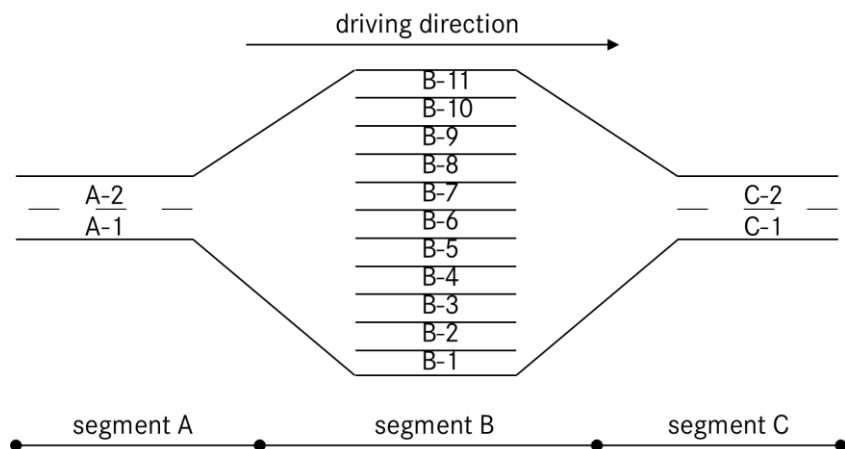


Lane B-2: first predecessor 1, last predecessor 1, priority predecessor 1

Lane B-1: first predecessor 1, last predecessor 1, priority predecessor 0

The straight forward predecessor for lane B-2 is A-1. However, B-1 is a new lane and has no straight-forward predecessor and consequently the priority predecessor is 0. This is analogous to the merging scenario in 10.10.1.1.

#### 10.10.1.7 Example: Toll Plaza



Lane B-11: first predecessor 1, last predecessor 2, priority predecessor 0

...

Lane B-2: first predecessor 1, last predecessor 2, priority predecessor 0

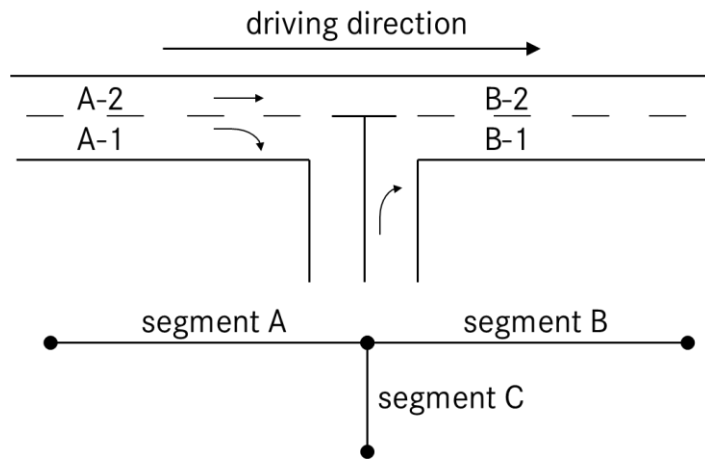
Lane B-1: first predecessor 1, last predecessor 2, priority predecessor 0

Lane C-2: first predecessor 1, last predecessor 11, priority predecessor 0

Lane C-1: first predecessor 1, last predecessor 11, priority predecessor 0

At a toll plaza, one can more or less change from any source lane to any target lane so there is no priority predecessor lane defined.

#### 10.10.1.8 Example: Turning Lane



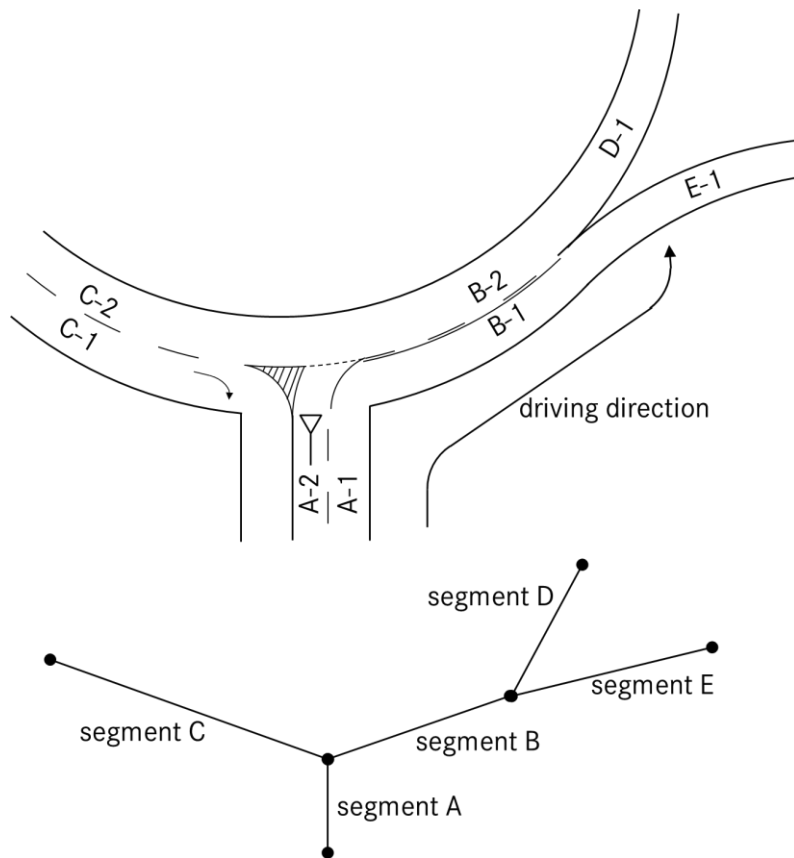
Lane B-2: first predecessor 2, last predecessor 2, priority predecessor 2

Lane B-1: first predecessor 0, last predecessor 0, priority predecessor 0

A-1 is not connected to B-1 as it is a turning lane. Therefore, lane B-1 has no predecessor along the path.



#### 10.10.1.9 Example: Roundabout



Lane B-2: first predecessor 2, last predecessor 2, priority predecessor 2, yield flag set

Lane B-1: first predecessor 1, last predecessor 1, priority predecessor 1, inner line may be crossed only from inside to outside

Lane D-1: first predecessor 2, last predecessor 2, priority predecessor 2

Lane E-1: first predecessor 1, last predecessor 1, priority predecessor 1

Traffic moving from A-1 to B-1 does not have to yield to any traffic in the roundabout. However, traffic from A-2 to B-2 has to yield to traffic going from C-2 to B-2 (yield flag is set), but A-2 is still B-2's priority predecessor along the path. The inner line of lane B-1 can only be crossed from inside to outside and D-1 can only be reached from B-2. As a consequence, traffic going from segment A to B to D has to use lane A-2.

#### 10.10.2 Lane Attributes

<i>Value</i>	<i>Description</i>
<b>0</b>	Virtual shoulder lane (this lane type can be used to transmit the right line marking of the right-most lane)
<b>1</b>	Normal lane
<b>2</b>	Exit lane
<b>3</b>	Entry lane
<b>4</b>	Auxiliary lane (combined exit and entry lane e.g. at highway intersections)
<b>5</b>	Emergency lane
<b>6</b>	Restricted lane we are <u>not</u> allowed to use (e.g. bus lane)
<b>7</b>	Restricted lane we are allowed to use (e.g. passenger car lane to overtake truck holding lane)
<b>8</b>	HOV lane

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>9</b>	Express lane (for faster moving traffic on highways with possibly fewer entries/exits)
<b>10</b>	Reversible lane (i.e. traffic direction may change depending on the time)
<b>11</b>	Slow lane (e.g. truck crawler lane)
<b>12</b>	Drivable shoulder lane (i.e. emergency lane that is used as extra driving lane e.g. for rush hours)
<b>13</b>	Turn lane or suicide lane (i.e. lane may be used from both directions for turning or passing)
<b>14</b>	Reserved
<b>15</b>	unknown or N/A

**Table 38: Lane Type**

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>0</b>	No line marking

<i><b>Value</b></i>	<i><b>Description</b></i>
<b>1</b>	Solid line (single or double)
<b>2</b>	Dashed line (single or double)
<b>3</b>	Left solid and right dashed in the direction of traffic
<b>4</b>	Right solid and left dashed in the direction of traffic
<b>5</b>	Dashed blocks (wider dashes indicating an exit or entry lane)
<b>6</b>	Shaded Area Marking
<b>7</b>	Unknown or N/A

**Table 39: Line Marking**

## 11 References

- [1] [http://www.ertico.com/en/subprojects/adasis\\_forum/objectives\\_approach/](http://www.ertico.com/en/subprojects/adasis_forum/objectives_approach/)
- [2] [http://www.prevent-ip.org/en/prevent\\_subprojects/horizontal\\_activities/maps\\_adas/](http://www.prevent-ip.org/en/prevent_subprojects/horizontal_activities/maps_adas/)
- [3] Mezger K, et al (2004), "Preliminary Interface Requirements, Functional Architecture and Logical Data Model" D12.32, MAPS & ADAS Consortium and ADASIS Forum, Brussels
- [4] Otto H.U, et al (2005), "Data Requirements" D1, MAPS & ADAS Consortium and ADASIS Forum, Brussels
- [5] Angenvoort J, et al (2005), "Draft Specifications", Deliverable D3, MAPS & ADAS Consortium and ADASIS Forum, Brussels
- [6] Angenvoort J, et al (2005), "Interface and Data Entity Specifications (Draft): Part 1 – General Specifications", Deliverable D12.42.1, MAPS & ADAS Consortium and ADASIS Forum, Brussels
- [7] Beuk L, et al (2005), "Interface and Data Entity Specifications (Draft): Part 2 – Data Protocol", Deliverable D12.42.1, MAPS & ADAS Consortium and ADASIS Forum, Brussels
- [8] NAVTEQ's GDF 3.0 Reference Manual v38.0, NAVTEQ 2011
- [9] Tele Atlas Multinet Version 3.5 Format Specification, Data Model, and Data Specification, Tele Atlas NV, 2008.
- [10] CEN ISO 14825:1995, "Geographic Data Files (GDF) 3.0", 1996.
- [11] ISO 14825:2004, "Geographic Data Files (GDF) 4.0", 2004.
- [12] ISO 14825:20xx, "Geographic Data Files (GDF) 5.0", Draft, March 2009.
- [13] Durekovic S, et al (2009), "ADASRP 2009 User's Manual", NAVTEQ 2009.

- [14] ISO 3166-1:2006, "Codes for the representation of names of countries and their subdivisions - Part 1: Country code"
- [15] ISO 3166-2:2007, "Codes for the representation of names of countries and their subdivisions - Part 2: Country subdivision code"
- [16] Convention on Road Signs and Signals, Vienna, 1968  
[http://www.unece.org/trans/conventn/Conv\\_road\\_signs\\_2006v\\_EN.pdf](http://www.unece.org/trans/conventn/Conv_road_signs_2006v_EN.pdf)