ADAS v3协议

说明书：V3.1.0

2018.12.01

保密资料 版权所有

文档控制

文档标题 ADASIS V3协议

电子档名称 ADASIS V3 协议v3.1.0

主要作者与编辑者

当前作者

贡献者

版权声明

本说明说收版权与法律保护，包含了关于ADASIS AISBL的相关专属内容，在没有ADASIS AISBL的书面许可下，不得将此专属内容或此内容的部分章节用于生产，发布，分发，传输，展览，广播或进行他用。读者可能会使用本说明中内容去实现一些功能，并且在其产品中并未移除本产品相关商标、版权声明或其他信息。这并不代表可以进行任何相关产品的生产，公开，内容分发，或者售卖。

版本3.1.0

状态

日期 2018-10-01

作者

复检者

变更概要 初始化、释放

1. 说明见 1.1.1部分 关于协议更新到3.2.0
2. 修改呢变更来自ADASIS AISBL 变更来自ADASIS会议与法务。

概念

ADASIS AISBL致力于将数字地图与ADAS的接口协议标准化。这些接口将给ADAS应用提供

基于车辆位置的车前道路特征信息（比如，大/小路的集合信息，斜坡，限速标识，路线上开放信息标识等等）这种能力称为ADAS Horizon。

当车辆路线或目的地变更时，ADAS Horizon会灵活适配变化并具传输动态数据（比如，信号灯状态）。此外，ADAS Hozrizon能够整合一些辅助数据（比如，交通信息）并且支持传感器数据融合，从而提供多种交互数据方案。

目录

1. ADAS的概念与组成
2. 真实世界中的抽象意义
3. 丁字路口的抽象点
4. 高精地图中丁字路口的抽象意义
5. 可选路线
6. 电子地图中道路网格显示与车辆已选路线显示
7. 基于道路网络的车辆行进路线的简要特征
8. 道路上地图特征范例
9. 道路网络
10. 车辆附近的link网络范例
11. ADAS HORIZON 中相关的道路与link
12. ADAS HORIZON-路线特征
13. ADAS V3 HORIZON -优化路线特征
14. ADAS V3 HORIZON -应用界面
15. ADAS V3 HORIZON -拓补网络
16. 不同类型的交叉路口的区别
17. 车辆相同位移下，多个特征值举例
18. 道路特征数据变化举例
19. 车辆位置信息的时间戳应用
20. 左侧当前路线和关联路线都是P1.右侧：当前路线是P1,其关联路线是P3,MPP上P1-P3路线上父子关系的展示
21. ADASIS V3系统的基础结构
22. ADASIS V2中路线移除（车后范围100m）
23. ADASIS V3中路线移除（路线数据生命周期规定为车辆经过后100m将被移除，路线生命周期则车辆进过后500m将被移除）
24. 数据更新概述
25. 怎样更新掉无用数据
26. 更新终点的偏移量
27. 范围更新
28. 范围无效
29. EHR数据反馈
30. ADASIS V2和V3 Horizon的数据兼容
31. 单个AHP对ADASIS V2和V3的支持
32. 基于核心数据域辅助数据的ADASIS V3系统图示
33. ADASIS V3 Horizontal基于多远数据的概念说明
34. 下游传感器数据融合概念
35. 上游传感器数据融合概念
36. 常规ADASIS V3 数据流程
37. 用于新增信息的循环计数器相关概念
38. 消息丢失异常场景之一-----循环计数器的消息序列中断
39. 消息序列记录获取
40. 路线数据包包含了基于起点和终点的有效性数据。
41. 路线数据范围内允许的误差说明
42. 丁字路口使用场场景

名词解释

ADAS：高级驾驶辅助系统

ADASIS：ADAS的接口说明文档

ADASIS AISBL：布鲁塞尔条约下的一合法非盈利组织

Av2：v2版本的ADASI协议定义说明（为了区分和ADASISV1,此版本已在预装地图和ADAS项目上进行了开发与测试）

Av3:v3版本的ADASIS协议定义说明（与v2版本有区别）

CAN（局域网控制器）：在自动驾驶应用中，用于设备间消息通信的协议

CAN-FD（数据可变的CAN）：CAN协议的扩展协议，单个消息最大容量为64 bytes

ADAS Horizon（AH）：获取车辆前方地图数据

Horizon Privider（AV3HP）：视野数据提供者，ADAS软件的一部分，用于生成ADAS Horizon数据，并提供给其他应用

Horizon Reconstructor（Av3HR）：重够了Av3的结构与接口，并命名为Av3HP

Network of Links：各个路段集合，他们之间通过起点与终点链接，构成了道路网。

Path：在道路网络上，车辆有可能选择的行驶路线

Profile：当前行驶路线上的路线属性与特征描述。

Profile Entry：根据现有经纬度确定的当前行驶路线上的路线属性与特征描述。

Profile Type：Profile的类型，比如当前路段的斜坡曲率。

Segment：Path上的每一段路段的属性是是 ADAS应用中最重要的部分，这部分没有太大改动。

Link：来自于地图数据库，范围是道路中连接两个岔路口中间的部分

Most Prefered Path（MPP,formerly Most probable Path）：某条道路上车辆的偏好路线，这些路线可能是其他路线组合而来。

Root Path：根Path

Sub Path：父Path上的分支Path

Vehicle Path：当前车所在的Path

Transmitted Path：数据提供者通过网络传输Path数据到各个客户端

Path Identifier：Path的唯一标识码。

Offset：在Path上的各个位置

Unknown Value：来自Horizon数据提供者提供的额外数据，目前对于当前Horizon暂无他用

Signal not Available Value(N/A):同上

Spline:通过多个方法并基于一套特定控制点而生成的曲线

Clothoid：根据距离而产生的增加或减少的曲线变化，并且能够体现出道路上曲线变化的最优点。

Message Type：ADASIS V3协议中的消息类型

Path Control Message：路线控制消息

Profile Control Message：Profile相关控制消息

目录表：

1. 概要
   1. ADASIS V3说明文档
      1. 协议更新与兼容说明
   2. Franca IDL 相关说明
      1. 导入Franca
      2. Franca数据转化
      3. Franca的部署说明
      4. 高级定制
2. ADASIS V3的概念
   1. 综述
   2. 相关模型
   3. ADAS Horizon中Path的含义
      1. ADAS Horizon中相关Link的含义
      2. ADAS Horizon 中Simple Path的含义
      3. ADAS Horizon中ADASIS V3 优化路线的含义
   4. ADASIS V3 Horzion中路线分支的深度
3. ADASIS V3 相关基本概念
   1. Path
      1. 相关定义
      2. 路线的可变树状展开
      3. Horizon拓补图
      4. Horzion 界面展示
   2. profile
      1. 定义
      2. 插入类型
      3. 路线特征的相关级别
      4. 全局数据
   3. 位置信息
4. ADASIS V3的管理
   1. ADAS V3 Hozrizon中发送者与接收这的数据同步
      1. PATH的同步
      2. Profile的同步
      3. Path生命周期与Path-data的生命周期分离
   2. 更新和清除机制
      1. 数据更新
      2. 更新范围
   3. 预估数据-内存管理
   4. 预估数据-AHR 反馈
   5. 兼容ADASIS v2
   6. 预估数据-辅助数据提供者和传感器数据融合
   7. 通信方案
      1. 广播
      2. 预估-组播
      3. 单播
      4. 请求/相应
      5. 发布者/订阅者
5. ADASIS V3协议使用入门
   1. ADASIS V3 中相关消息
   2. 路网特征 （profile）
      1. 路网特征（profile）自定义
6. 指导方案与相关推荐
   1. 带宽要求
   2. 实现方案推荐
      1. Path的数量
      2. 相同位置下，相同Profile消息类型
7. 异常数据检测与恢复
   1. 异常检测
      1. 消息丢失与消息重排
      2. 误差空挡检测
   2. 异常恢复
8. 使用案例
   1. 小路段的联通
      1. 大距离下小路段的组成机构的常规案例
      2. 特殊条件下小路段的快速生成
      3. 特殊条件下小路段的分裂展开

8.2 不同缩放级别下的丁字口的详情信息

9．其他相关参考资料

1. 概要

预测电子地图数一般是应用在导航系统中，这也是车机接入地图相关应用的主要目。目前越来越多的车辆都可以有效利用道路信息了。相对于摄像头与雷达系统，电子地图能够提供更多、范围更广、更有预测性的数据。在不久的将来，电子地图也将会完善来自于雷达，摄像头，超声波或激光雷达传感器的数据。

* 1. ADASIS v3 文档说明

ADASIS（高级驾驶辅助系统接口文档）定义了“ADAS 视野”的相关概念，“ADAS 视野”可以通过电子地图来传输当前车辆所在路线的车前数据。为了实现这一点，“ADAS 视野”通过当前车位与当前道路上车辆接下来可能会行进的路线来构建视野数据。这一结果会在道路网络上将会以不同级别来呈现。除此之外，道路的几何纹理也会被抽象化，其相关的一些属性也会被添加到车辆的预测行驶路线数据中。“ADAS 视野”可通过车辆通信网络来传输这些数据。

ADASIS AISBL是ADASIS的标准交互协议，基于之前的版本进行了改进，同时增加了对应用详细解决方案的支持，这些相关优势已经在ADASIS V3 版本中实现了。

ADASIS V3标准协议旨在促进ADASIS V3 Horizon数据从Horizon数据提供者到各个接收端的数据分发。为了实现这个过程，ADASIS V3标准定义了专门的数据结构包含这些数据，并使用Franca IDL 来作为正式开发语言。名为“ADASISV3Messages.fidl”（基于Franca编写）的描述文件就是ADASIS v3标准的一部分，除了常规信息背景描述之外，Franca IDL在数据结构上还有一严格的语法要求。

因此，ADASIS V3 是基于OSI 7 层网络模型，并在逻辑结构上与Horizon 契合

在未来的某个时间点，ADASIS还会标准化从数据提供者到接收端的数据格式与编码格式。然而在ADASIS V3 创建之初，这些构想仅仅是一些非正式的提议。

ADASIS V2 是基与CAN协议为了支持公交车上数据通信而设计的，不过单条消息的数据大小限制为8byte，ADASIS v3 在通信上将使用更高的带宽。单个消息支持更丰富的数据，并且不比关心数据的编码格式。ADASIS V3版本中，单个消息有更大的数据承载量以便于数据实体封装，序列化，或加入更复杂的实体。

ADASIS V3的主要优化点如下：

数据分辨粒度从1m提升至1cm

Profile Entity提供了详细的位置信息来描述数据的合法性，也包含一些拓展数据，或者下一推荐Profile Entity的数据

多个Profile Entity数据可以封装到一条信息内

新增路线控制，道路属性控制机制

新增多个动态行为

车辆位置信息目前包含潜在位置与预估位置并将它们封装到一条信息中，而不是之前的独立消息

乡间小道和小路的详情与几何构造信息

其他额外数据（比如持续变化的车速数据）

对于详细数据的支持（比如来自HD map的支持）

使用ADAS Horizon时，常规结构组成如下：

Horizon数据提供者

Horzion 重构结构

ADASIS V3 协议

ADAS 相关应用

* + 1. 协议更新与兼容性

ADASIS V3文档说明发布后，ADASIS V3的后续开发也在紧张的进行着，下一个版本3.2.0将会加入以下新功能：

路线连接：使用融合目标点位来完成路线树中相邻分支路线的关联，从而实现全量的路网模型。

全新的道路属性信息中将会包含道路上的一些环境信息，比如道路上的“路障”，Path 级别的转向限行等

其他一些扩展枚举数据，比如“信号灯”的相关概念，将会遵循最新的国内或轨迹标准。

所有的概念以及相关道路属性信息将会根据兼容规则[2]向后兼容到3.1.0

v3.2.0的协议中也包含了一些v3.1.0不支持的特殊功能，这些功能之所以不考虑支持的原因是为了确保系统的一致性。以下是所有3.1.0中不兼容的功能列表：

道路属性相关

FRANCA相关

除了所有的扩展功能和变更以外，ADASIS V3.2的文档中也会包含大量的文字说明，所有的placeHolder和gaps都附带详细的说明与案例。

* 1. Franca IDL文档说明

ADASIS V3将会基于Franca IDL并以最全量的交互为目标来实现不同供应商设备上的数据通信。为了实现这一点，非常有必要定义清楚协议中数据的颗粒度细化到bit或字节。这些与协议的数据格式和语法要求都是紧密相关的。

协议的数据含义描述应使用通俗易懂的语言描述，这一点文档中也有提到。至于协议数据结构则需要使用一种描述清晰、机器设备可识别的正式语言作为数据结构。

ADASIS 使用Franca IDL作为ADASISI V3协议的正式定义语言工具

Franca作为一种高级别元语言可机器转译成其他接口定义，也可以直接用来编码。Franca是在逻辑上描述数据结构与相关接口，并不会强制规定数据在二进制中的表现形式。如需要一个全量的二进制数据结构则需要给给到Franca一些列的定义转译规。仅仅是数据不足以构成通信协议的，除此之外，至少需要定义数据在媒介中的传输规则，而对于传输媒体，则需要知道通信双方怎样获取对方、如何建立通信等等。

因此，一个ADASIS V3 协议的详细二进制交互说明实现方案由以下几部分组成：

Franca IDL文件定义了ADASIS V3的协议中消息的数据结构。

ADASISI V3协议中说明了消息体中数据的含义，同时指出了域ADASIS V3 Horizon中的传输过程

数据翻译规则方面，Franca中定义了转译规则来描述怎样将接口转译成为ADASIS V3

数据传输方面，则定义了ADASIS V3 协议中怎样使用下层协议进行传输（比如ip协议）

目前，只有ADASIS V3 协议说明与Franca IDL是正式支持。ADASIS有意向提供正式的转译与通信标准。也希望能够提供一系列转译规则与通信方案来满足不同用户ADASIS 的使用需求。如果是这种场景下，非常遗憾，不同的ADASIS V3环境下的二进制数据不能兼容，但是在数据传输上，可以进行大部分数据重用。

* + 1. 导入Franca

事实上，Franca是非常适合ADASIS V3的正式语言。Franca为一种设置型工具。基于Eclipse 可以用于编辑与验证说明，这些都是用Franca来写的。使用此工具能够非常轻松的创建转译体来讲Franca转译成另一种语言。这种转译类似protocal Buffers（谷歌推出的序列化格式-特点：小快灵）。如果是直接使用Franca进行编码的话，此种语言风格类似C语言。

更多Franca环境搭建详情，请查看Franca Wiki 中的Quick install Guide ，目前来说不必使用Eclipse Mars版本来搭建，当前Franca版本（0.11.1）插件已在Eclipse Oxygen上支持了并进行了ADASIS v3的接口测试。当然了也会对后面的eclipse版本进行支持。在任何情况下，在搭建环境时候，推荐使用Eclipse的DSL工具包。Franca 就是一个DSL，在Eclipse标准版中就加入DSL之后，大量DSL相关工具将自动导入。

在Franca这部分只需要加入运行环境和Ui组件即可

* + 1. Franca 转译模块

在Franca上下文中，目前仅仅支持开放，图形化界面，并且可编辑ADSISI V3 中Franca相关的文档。在未来，也许需要使用Franca的转译模块转译Franca为其他语言。ADASIS AISBL 在自身部分提供了转译请求与案例，将Franca转译成为Protocol Buffer和C++。此案例仅适用于ADASIS的特定场景开发使用，并不一定适用于其他特殊场景或生产环境。

* + 1. Franca的部署描述

在实际开发中，ADASIS V3的子系统中，Franca消息文档与目标转译器都不包含目标数据的特有属性。Franca中所谓的部署描述说明指的是它可以提供一种可选的属性配置。比如内存使用上限配置等，不过这些仅仅是Franca中消息模块的一部分并不属于是ADASIS V3。不过这些也是兼容的，也属于标准管理范畴内。

其实在ADASIS\_spec.fdel文件已经有了配置部署说明，并提供了一套标准的可配置参数。在版本上也有一个试样版本可用参考。后面也也会根据OEM生产商在他们的汽车内的ADASIS V3部署状态来推出更多车辆定制的 ADASIS v3 软件。

可以看到，软件供应商对车辆上相关软件的支持与重构编译是依赖OEM生产商在运行Franca转译后的结果。因此，在车辆上有关ADASIS V3 的所有软件都应使用相同的OEM参数来减少冗余开发操作。

因此，强烈建议以这种方式来进行Franca转译的开发与接入，这样一来就可以保证OEM的部署说明与ADASIS V3标准部署定义的一致性。

* + 1. 后期的定制

除了ADASIS V3 在车辆上一些预定义部署之外，后期OEM也能够继承ADASIS V3 并一些特有数据，特别是一些特有的道路属性数据，这也特性也可以参考Franca的能里，并在ADASIS 中轻松实现。

ADASIS AISBL 提供了图解样例来实现这定制：

ADASISv3MyCustomized.fidl文件包含了Franca的用户定制外观、定制消息类型

ADASISv3MyCustomized.fdep文件包含了整合标准ADASIS V3数据结构的部署描述。（非常遗憾，由于技术原因，必须要copy ADASISv3MyCustomized.fidl 的全部信息到ADASISv3MyCustomized.fdepl 中）

2 ADASIS V3 概念

2.1综述

在电子地图数据库中，真实世界中的各种事物将会按照不同级别抽象成为各种信息。

最简单模型就道路模型了，真实道路在电子地图中会被视为一条线，岔路口会被视为一个连接多条线的节点。节点和线条（也称作边线或连线）在路网中以拓补图的形式表现出阿里，后续将会以地理经纬度来表现。

真实世界的道路在电子地图中的如图2所示，蓝色线条连接着各种各样的节点，红色点代表着道路的形状，紫色的线代表link分支。

在这个简单的丁字路口模型中，蓝线与蓝色的圆圈和展示了一个简单的路口模型，蓝圈代表电子地图中的节点，蓝线则连接这这些节点。红色的点与线条则描述了更为详细与真实的道路模型。

这种方式，可以将更为复杂的环境简化成为此种抽象点，这个例子的俯视图也可以用一个抽象的图表示如图3

以后应用中可能需要更为详细的道路信息，此种现实抽象的方法也可以进行复用，只不过需要更详细的网状节点。这些更详细的网状节点代表了更详细的道路信息

图5则展示了丁字路口的通行信息，这些信息来自于HD高精地图，HD地图中含有更多详细信息或者说是优化掉了很多冗余空白区域。通过HD高精地图更够在丁字口上给到更多的通行信息，这样一来，可用车辆选择的路线就更丰富了

2.2 引用模型

最简单给一些应用获取地图数据的方法是传输部分道路路网信息。路网信息是一协议消息序列中颇为复杂的结构，它们或多或少会与图像强相关。这些消息中能够区分出links 和 nodes 。只要车辆在运行中，图像传输变化就会一直进行中，links 与 nodes 也会一直进行增减。

ADASIS V2 与V3方法中切换地图中心为车辆中心的方案如下：

车辆在道路上行驶时，无法在同一时刻驶向路口的四面八方。换句话说，车辆在道路上行驶的时候，即便是通过路口，车辆也只可能按照一条路线行驶，这就是我们所谓的 PATH，Path信息在传递时包含了现实生活中的一些物理意义，比如基础建设之类的。因此，Path代表的 ADASISI 提供给车辆的一条建议路线。

根据这个假设，那么车辆运行时，我们没有必要传递整个路网信息而应该给到的是一个裁剪后的地图卢新片段，而且也有很多方法可以访问到地图数据。数据传输只不过是车辆运行在特定道路上才进行的。（图6显示了地图上车辆运行在一条指定路线的案例）

通过使用行驶路线，所有潜在路线，这些路网信息与当前车位可以计算出车辆行驶的起点。见图7

Path的数据源自电子地图中的路网信。这些Path是由一些有序的街道link与丁字口组成的。根据路线的概念，路线上的各种建筑属性与地理位置都能获取到图7展示的就是一个现实道路中的抽象意义。（包括了节点和分叉点）

在Path的线性结构中，自然经纬度在此处代表的是一个距离标记代表Path的起点或者是零点。这个标记举例成为指定path上的偏移量。假如路线的偏移量是0m那么通常情况下，这个起点位置在一个路口上。可以能是上一个路口，或者是靠近车辆起点的位置的一些物理环境。（同样是图7，可以看到Path1的起点是靠近车辆位置的，这是第一次匹配。而path2的起点则是分叉点。）

当前车辆位置在粗略看来属于一个逻辑点，它可用来测量位置偏移量。但是这还不足以作为ADASIS中的目标数据。大多地图数据都会被处理为一个修正值，因此，获取到每条数据的变化量以及修正值是非常有必要的。车辆在运行中需要包含地图经纬度系统与路线偏移量经纬系统。他们会提供给车辆可变的修正值。车辆位置偏移量可以按照修正频率进行修正（比如2s一次，每分秒一次）或者当车辆运行时候改变。亦或者两者混合。在两个地图数据之间或是车辆与地图之间计算出来的举例数据都是不一样的。

由于车辆并不会主动在很早之前就确定路线，因此我们并不能够确定车辆所行驶的唯一路线。所以我们必须要说明清楚推荐路线的细节与差异

一条路线上的偏离点主要出现在岔路口或分支点上。预测路线上的每一个偏离点都可以作为车辆在路网上行驶时额外附加路线。这些附加路线会被当做是路网信息的一部分被提取出来提前传输到车辆。这个结果可以单路线的数据拓展，也可以是多路线的数据拓展，如果是相同的车辆方案是完全可使用。

2.3 ADAS Horizon中Path的含义

抽象并高效的路网特征数据需要通过ADAS Hozrizon的数据提供者与数据整合器来完成传输。Horizon数据提供者能够通过路网信息来决定并给一套潜在的路线信息，其中包父子节点路线关系，一下的次级标题将会对此进行详细描述：

2.3.1 ADASIS Horizon 中相关联的link的含义

在电子地图数据库中，路网（如图9）代表了各种路段的集合，各种节点则代表了这些路段的关联关系（如图10） 。

对于一些支持地图并且高度集成ADAS的应用中。会在车辆运行时会在适当时刻将车前一些关联的兴趣点给到车辆。

ADAS Horizon也可以理解为电子地图的一部分，但是只含有车前方的道路信息。通过对比图10中车辆周边的道路与图11中相关连的ADAS Horizon路段信息可以观察到 links 95 100 和 105 在对于ADAS Horizon来说是不重要的信息，这些将不会列入到ADAS 信息的范畴。此外，并不是所有的道路舒心信息都会被ADAS Horizon提供或使用，比如街道名称或建筑物门牌号等，ADAS应用很少使用到。

换句话说，ADAS Horizon为ADAS 应用提供了一个优化的界面以及基于path的界面道路环境。为了更加高效的进行，这些信息将在下面的副标题中进行说明。

在ADAS Horizon架构中包含各种算法，不过这部分并不在本文档的范畴内。

2.3.2 ADAS Horizon中最基本的Path说明。

ADAS Horizon基于电子地图中的路网与路段来抽象出来一些道路的信息。如图11所示，对于ADAS 应用来说，这部分可能颇为复杂些，因为如要获取车前路段的特征信息与地理属性就必须要获取到整条路网信息。举个例子，一个交通信号灯的兴趣点在235link上，那么ADAS应用就必须判断并分析出所有到抵达此link的方案 比如：200-210-230-215-235 或 200-205-220-235

不过 对于ADAS 应用来说只关心一些预测的路线，因此在路线的处理上就显得方便多了。在Horzion数据提供者中，所有的path的构建都是来自地图数据库以及其周边关联模块，但对于ADAS 应用来说，每条path都视为单一实体

回到我们最初的例子中，通过查询路线特征，应用可以很容易分辨出path3和path5上的特有信号灯。不过这里不一定能确定这两条路上交通信号是否一致。对于大多数应用来说，这种信息对于车辆来说其实是不重要的，因为只有车前的信号与距离车的距离才是车辆所关心的。

在图12中显示了纯Path的信息，其中包含了大量冗余信息。举个例子，5条路线都有相同的起始点。假如传输通道或带宽过低的话，那么ADAS Horizon就非常没有必要传输这些信息了。如要传输，则需要提供比数据要求上更高的带宽。

优化的Path在ADASIS v3中有所减少。

2.3.3 ADAS Horizon中优化Path的特征说明

ADASIS V3的概念描绘了ADAS Horizon 是怎样的使用优化路线。此方法减少了大量冗余数据，同时也提供了很多优良的路线（图13）

当优化路线只有1条的时候，那么这就是我们所谓的根路径。

车辆运行时，可能会从根path转到副path，如图13，path1，3,4 都属于根路径的第一级的子Path。

对于一级子path也可能会转到更高级别的子path（path4转到path5这种的）

ADAS Horizon 的架构算法会自动选择root path这样以便于车辆继续行驶时，随时切换。而第一级别的附属path 车辆很少行驶。

大多数ADAS 应用只会在一条path上进行操作。最简单的案例就是当前车辆所在的path。不过根据当前路线获取的信息很有可能帮助车辆转弯到副路径上。在这种情况下，一般应用将会从当前路径上车辆位置作为起点开始，之后行驶到副path上。这就是所谓的潜在路径。这并不是传统意义上Horzion中的一条路径分支，不过将多条path链在所组成的路径树状结构就是当前车辆最有可能行驶的路线。

在分配的环境中，尤其是存在ADASIS V3 Horzion 数据提供者与ADASIS V3 应用时，所发送的次级路线的层次度取决于 ADASIS 应用对Horzion数据的容错能力。当车辆从一个MPP到另一个新的MPP时，如果几秒钟的数据盲点是在接受范围内的话，那么这一时刻仅能传输一条path数据。当车辆离开这条path的时候，ADAS Horizon需要再去创建MPP并发送，这些都必须在短时间内完成。

如果应用一直需要全量的Horizon数据的话。那么全量的ADAS Horizon数据（包含了所有潜在路线）也将会被传输。非常确定的一点是，如果车辆离开了一条path那么client端将会自动在目标位置给到另一条路。

自动ADASIS V3 实现了ADAS Horizon对最优路线使用。我们称之为 ADASIS V3 Horzion。

在ADAS HORZON中，交通要道，道路属性甚至于几何地理特征都将作为路线的特征之一。因此，在应用所检索的有效信息中，path的实体对象才是最重要的。

从应用的角度（如图14）。车辆在一条path上或者是在第二条path上（作为车辆的可变位置的path）兴趣点数据要么是当前这条path要的就是ADAS 视野中的一条副path

2.4 ADASIS V3 Horizon的路线深度

ADASIS V3 Horizon路线分支被限制为多个等级，在这种树状路线分支中，叶子节点的信息往往会比跟根节点信息更丰富。路线分支级别主要是以下几种：

单一路线：不存在其他分叉路线，并且除了当前路线以外，其他路线都不会提供。

可选路线：当前车辆未来会经过的潜在路线会被提供

全部路线：其他额外路线信息将会提供给车辆作为潜在轨迹参考。该级别的信息会随碰撞点而增加

除此之外，ADASIS V3 Horizon中所有路线和分支都是有数量上限的。

Horizon提供的path信息中仅会包含当前所展示的路线信息。因此，单个path能够展示出所有关联子path所在的节点，但不会提供子path的详情

3.ADASIS V3的基础实体

正如我们所看到的，ADASIS V3 Horizon由多种path组成，每条path都以id进行区分。

ADASIS V3中Path信息涵盖了根Path 父Path 子Path 等。这些path的起点取决于父path信息以及父path的起点偏移量

Path的固有属性来自于各种Profiles.而各种Profiles则是来自于ADASIS V3 Horizon在当前路线上给到的数据信息。

当前车辆状态会根据其位置，并结集合ADASIS V3 Horizon与其他车辆相关信息（比如车速，车头朝向等）所定义。

3.1 path

ADAS Horizon通常来说是一个展现车辆前方周边信息的预测树状图，此图描述了不同的path与导航地图的相关数据。这种预测树状图由各种相连接的path组成，每条路径属于道路的一部分，并且这些路径通过交叉口相连接（这里的交叉口映射的是真是路口）只要车辆移动，位置有了变化，那么预测数也会有变化（经过的path会被移除，新预测的path会添加）

一条path的特征主要来自于一个场景数据的一个结合体，这些数据描绘了车边环境、街道环境、大量的小路、物理几何信息、起点偏移量、路径父子关系等。

偏移量数据属于一种距离标识与定位说明属于path的真实数据。尤其是在定义绝对距离的时候。一条path与偏移量数据是密不可分的，path的源点与偏移量数据连用从而确定偏移点与path源点间的距离。偏移量数据将会增加path的声明周期。如果一条path是个rootPath，偏移量数据为0，此值将会与有效点相关联从而与车辆的起始位置相匹配。

在实现这部分功能时，系统工程师将定义Horizon长度上限；在特定场景下，Horizon数据是不会超过上限的。Horizon长度的上下限范围如下：

Horizon\_length = greatest offset – smallest offset

数据提供者在构建path时，可根据下限和上限自由构建。

在特性的Horizon TREE 构建时，不会存在差异与缝隙。每条path都与其他path相连接（除了rootPath），每条path至少有个父Path，一个父path也能有0个或多个子path（或者叫副path）。如果是在HD map可用或者是处于岔路口时，此时将会从父path上的一些路口分叉点开启一条path

3.1.1定义

ADASIS V3 介绍了一些path的定义与概念，部分说明已在V2中已进行叙述，其他的部分则是一些新引入的概念与不同点。

Root Path

Vehicle Path

Most Preferred Path (MPP)

Root Path:一条根路径是路径的源头，因此是没有父路径的。对于一个特定的Horizon tree来说只会存在一条Root Path，其他类型的path将会相互连接，并最终也能够连接到Root Path。

Vehicle Path: 其表现为一条高亮的path也是车辆即将行驶的path，取决于不同地图所匹配的位置点

MPP:其表现为高亮的一条路线，车辆也是最有肯能跟随的一条路线。开始于当当前车辆，结束于更优path的message。在Vehicle path 与 mpp之间构成的完成path链条中可以被看做是子path-父path的关系图，定义了从当前path到目的地的path之间的关系。这也是车辆即将行驶的。这个概念替代了ADASIS V2 中 Most Probable Path的概念。

3.1.2 可变path树状结构

在ADASIS V3,并不是所有的path都来自于一个path tree、一个root path。Path tree可以是多种的。每个pathTree 都有自己的root，因此在ADASIS v3中就形成了一种多元path森林。

一个path森林包含了车辆在道路上不同位置的相关结果信息。他们彼此之间互不联系。因此，在这种情况下，我们会有一个车辆主位置来作为一个主树的数据，这个主位置来自于车辆位置列表中第一条数据。（通常情况下，车辆位置是最优先的，但不是ADASIS V3 所必须的）其他（可变）车辆位置要么也是在主树上，要么就在其他的额外的无联系的树中，我们称之为可变数。

除此之外，也会有path和树间接关联车辆位位置的场景（成为：微弱行驶意向数）

3.1.3 Horzion 几何结构。

为了构建Horizon，在车辆的行驶方向上会按照父-子关系组合连接起来。

如图15，path1是Path2的子path；path5是path4的子path path5和path4都是path2的子path

3.1.4 Hozrizon的视角

视角或上下文的定义依赖于以下几点：

静态视角：系统将车辆当前行驶的path作为最有意向路线（MMP），当到了岔路口的时候，path就会高概率延展路线。如果导航系统支持，那么在岔路口时，path路线延伸也会响应系统。相关的路口延伸路线将会作为其他path的起点。当到达一个新地点时，MMP是不会改变的这与新地点的信息与信息来源无关

动态视角：（路线/目的地改变的情况）:MMP变化（在MMP的结果中不同于当前车辆path）

导航系统修改了导航路线

用户修改了目的地。

3.2 Profile

Profile以常规的方式提供了在Horizon的视角下当前道路上的属性信息。Profile含有类型的区别一遍存储区分。

例如，限速profile就提供了在path上当前点位上的限速数据。

3.2.1 定义

一个path上可以有任意个Profile,它们之间以类型进行区分，一个Profile其实描述了当前道路上的一些路网属。每个独立的profile中的属于与当前位置也是相关的。Profile实体集合中共有一个属性类型，这些profile实体集合组成了profile。

ADASIS V3会根据一些目标位置(如关键兴趣点位置)发送指定类型的Profile。如果这些位置含有默认数据（比如隧道处会发送“隧道”的profile 隧道外就不发送）的话，ADASIS V3 也会主动过滤掉一些。在此种场景下，数据接收端就没有必要确认数据是否接收完毕。

另一方面，对于某些profile类型（比如交通信号灯类型）在同一位置也可能存在多个profile实体的。一些profile类型代表了道路的特殊属性。当车辆再次处于相同位置的时，ADASIS V3也会透出相同类型的数据。

Profile数据按照其名称存储。根据Profile类型的不同，profile也会有特殊的类型数据。

在path上位置的profile取决于当前道路位置。位置的定义取决于与path起点的距离，这部分也在2.2中进行了描述。在存储的location相关的profile数据都是取决于插入的profile类型。基于插入类型，profile数据可能会被定义成一个包含一些必要数据域平滑曲线的向量用来描述相关信息。

一个profile类型都具备以下特征：

任何情况下，只有Profile中数据变化时才会被传输，除了位置点篡改。见下文

在path上的一些位置中，可能会丢失相关Profile数据，这种场景下，profile则则提供一个NotAvaliable 标记用以表示无效数据。

一些profile数据依赖于驾驶方向（或依赖于限速、或大量的道路）。这种情况下就需要对每一个方向进行数据分离（正向，反向等）。Profile中含有相同的type用来处理不同方向的数据（比如完成度的检验，更新等）

对于与path的方向相关的说明有“path正向”与“path反向”。车辆在path上的自然行驶与path方向是相关的。这种定义在不可达path上同样适用。如果一条子path提供的一条路是在与其父path的交汇点之前，那么我们将认为当前是“反向path”。如果子path起点是在岔路口很远的地方；“正向path”的场景就不必要了。

Profile中位置上的path反向需要进行path的方向测量，即结束点的profile实体要远大于逻辑起点（逻辑起点属于当前path中的某一个点）

3.2.2 数据插入类型

插入数据是为了计算两个Profile之间的中间值而产生的，取决于profile的类型。它描述了profile在path中位置的变化量。插入数据有Profile类型定义，也因此会被一个profile所修正。

以下是ADASIS V3所支持的一些插入类型。

1. 点类型插入：

基准点的插入只有在确定了具体位置后的路线上属性上才会生效。一般读取的中间属性数据都是一个估算值。连续不断基准点数据很可能都是一样的。此种情况下具体的位置才是影响属性变化的变量。

比如：交通信号灯可被定义为基准点，因为每一个信号灯总是在一条路上固定的位置因此中间量在这方面就不那么有用了。

对于一些特殊的路线信息属性类型可以定义为相同的位置说明，比如交通信号灯

1. 步长插入

插入点的步长也与道路属性信息有关，一个插入点步长的有效范围是当前与下一个位置所在道路属性之间。

比如，一个常规的限速常量会在一个确定的距离范围内有效，直到下一个限速改变点。

一段路线信息在同一位置可以持有多个值，但是这些值不能覆盖。在路线信息中，任何数据改变后，说有数据都需要重新创建，在道路信息属性值中，任何一个位置变量改变时，此位置上所有数据都会及时响应到。

图17中展示了路线为200m的图示。全程线束80，除此之外，还有2个限速区域分别是位置为50的坐标和100的坐标处。他们分别是：

位置0: 限速80

位置50：限速80 如果雨天此处限速60

位置100：限速80 雾天限速50 同时雨天限速60结束

位置150：限速80 且雾天限速50结束

1. 线性插入

线性插入道路路线数据适合于一些位置非常确定的点位，在两个点位之间的数据必须是线性的。这种方案比较适合于连续的道路信息变量的场景。

在一些位置上，线性插入的值可能会出现断层，比如目标点的逻辑左值与逻辑右值不一样的情况下。为了表现这种数据上的断层，将会使用以下方案。在道路属性数据中，左值的endOffset= 0 而同一位置右值中 endOffset > offset。

数据断层现象仅仅在同一位置出现不同数据的线性场景下

1. 高次序的插入

高次序的插入特征（方程、多项式等）特指了一些有相关属性的定位说明。这种插入算法常用来计算两个给定的点位中 profile特指的一些属性。

1. 特殊插入

对于斜坡或者弯曲路段，ADASIS V3 会将多组定位数据封装到一个profile实体中用以节约带宽。斜坡和弯道数据也会加入进来。但当profile 还不是标量值时候，这俩是不会打包进来的，只会将他们将会封装到整体的数据结构中。对于数据插入，客户端需要知道数据结构详情，也需要知道处理数据插入的依据。

对于“特有插入”只不过是在我们对特有场景的认知下的一种可能存在的场景说明。而且其他的一些profile都会将会有各自的插入类型与处理方式。

3.2.3 profile层级

在高精地图中，道路地图数据也进了扩充，这样就会导致数据数据传输时的数据暴增。这种情况下，发送者和接受者之间要需要时使用更高的带宽。然而profile的级别的特性就能减少这种场景。这种数据层级允许重建者挑选有用数据，从而达到数据提供者与带宽之间的优化平衡。

Profile层级在ADAS Horizon显示上已经完成。第一层同样也是基础层的数据为path的相关属性数据。下一个级别的数据则来自于更多的道路详情，岔路小径信息等。最后一个级别就是地理几何信息了。

假如数据构建这仅需要基础path数据，那么数据提供者就应该避免发送其它高级别的数据。这主要依赖开发者在使用api时所设置的数据获取界别。

3.2.4 全局数据

全局数据属于ADASIS V2中的常规数据，这些数据在传输时不包含特有的路线数据全局数据在传输时，使用GlobalDataMessage类来完成

以下全局数据的使用场景：

提供当前位置数据（比如国家码、区域码、速度单位等）

提供状态信息（比如版本信息或引导状态等）

提供相关事件 （比如当前距终点距离变化的事件或天气更新事件）

简单来说，全局数据和horizon路线数据用于同一个Profile结构中。Profile类型也可以定义全局数据，路线数据或者两者兼之。一个GlobalDataMessage会持有大量的profile实体全局数据经常在数据变化发送或者在一个指定的周期内发送。全局数据被分为不同的GlobalDataMessage

实例。每个时间点的实例对象都是不同的。

通常来说，GlobalDataMessage所在的事件中可以包含任意类型的数据，但并不推荐这么做，因为全局数据本应该仅和地图数据相关，并且是用于提供状态，或是ADASIS 协议版本等

3.3 位置

我们的路线系统将会提供一整套路线并且包含了路网信息同时在预测路线上也是有体现的、位置信息则来自于在路线树上的各个点位。位置信息通常用于描述当前运行车辆的位置信息，并且可以通知相关app当前车辆的位置状态。

车辆位置是与path进而offset相关的。由于一些未知因素，车辆点位会出现在多个path上。这种情况下，我们将会使用位置数组来作为当前位置的附加数据，在数组中，位置误差将介于车辆绝对位置与地图匹配位置（特有方案实现的地图抓路匹配）。如果当前车辆位置不匹配，则数组为空。以下是位置信息附带位置数组的场景：

车辆在路线之外

车辆在道路上。（地图上可以显示车辆在道路上）

车辆在不同的path上

车辆在路线之外，但是靠近一个path，同时地图匹配也是在这个path上

车辆正在靠近或者驶离一个path

额外信息包含了一些地图信息，车辆绝对位置信息。不过不含计算信息与系统状态。此类信息将会体现在全局数据中

时间戳是位置信息中的重要部分。这个时间戳指的是传感器收集定位数据的时间

（这些定位数据用于计算车辆位置）并不是路线被算出的时间，也不是车辆位置被计算的时间。也是不是位置信息传输时的时间。这个时间是来自于GNSS定位或是航位推算。时间戳一个全局的并且接收者和发送者共享。如果共享时间不存在，则位置时间是完全不同于传感器时间戳的。当位置信息被传递时，全局时间是需要重新整合计算的。

误差也是位置信息中的重要部分，这个误差的距离是在车辆的绝对位置与地图抓路匹配的位置之间如果当前车辆在野外地图或者正在靠近或驶离某个路段，那么这个误差将会作为参考从而决定是否应用于horizon Data。具体的误差细节见【2】

推荐路线也是位置信息的重要部分。这部分数据基于当前路线，并给到一个推荐路线。当然了，并一定是在一条单一的路线上 也可以是在副路线上。推荐路线将会提供很多大概率会使用的推荐信息。通常

情况下，ADASIS Horizon数据提供者将会假定车辆会使用推荐路线，数据提供者也会基于这个来提供扩展数据。 然而，在其他引导或者算系统中，这些信息将对MPP进行变更，除此之外，当前位置依然有效，这样一来就非常需要将当前路线与推荐路线，统一处理在路线列表中。

完整车辆信息由时间戳，位置数组组成。当前也有可能有以下场景：

无实体数据信息，位置数组也是null，如果没有可用的车辆绝对位置，没有地图抓路匹配数据或者不存在其他数据，则此位置信息将被忽略

有一个实体位置用于标记时在道路之外，车辆绝对位置已知，并且地图数据存在，但无法地图抓路配置车辆位置。

有一个实体对象状态为道路外，还有些是在道路上。系统在path上可以识别到车辆的绝对位置，但不满足使用的场景下。那么在道路上的那部分误差数据将作为地图抓路匹配的计算参考。

有一个实体位置状态为道路上。系统可以识别道路上车辆的绝对位置。在别的path上也没有备用数据

有多个实体位置状态为道路上，系统使用地图抓路匹配识别到多个车辆位置信息。这种情况下，第一个实体对象将视为推荐位置用于ADASIS horizon，并提供给client接收端使用。 这个位置不一定是最接近的位置。如果有两个位置他们都很接近，地图会进行对比计算，之后得到最接近的那个，但是数据提供这还是会要求接收端确定一个位置作为列表的开头。即便是数据误差较大也要定义好这个列表的开头其他的额外实体数据则是一些平行路段或者十字路口等信息。在位置实体类中参考数据，准确数据，误差数据都将作为导航精准度的质量与数据融合的衡量标准

4.ADASIS V3 管理

基础的ADASIS v3由horizon数据提供者组成并支持1对多的的的数据传输发送，如图21

这些应用可能是通过车内网连接horizon provider 也可能是来自于horizon provider 内部，并通过本地特有机制连接的。

4.1 ADAS V3 horzion 中发送方与接收方的数据同步

在ADASIS V3中，provider是用于管理活动的horizon内容。在数据提供这和接收者中有两 个不同的概念会动态改变目标信息，path管理和profile管理。Path管理着负责horizon中的 相关几何信息。Profile管理者负责path相关信息，这两个概念详情见4.1.1 4.1.2

新机制的好处：

信息的唯一标识将被废弃

生命周期中的动态行为

只有数据提供者侧才会实现。

将path和 profiles的声明周期分离

数据提供者这会对horizon全面掌控，不仅仅是传输数据到接受方，同样也会决定接受方什么时候删除数据。这样provider就知道receive的内存情况了，也就能够决定额外数据的量了。 如此一来，提供者发送数据时，接受方就不会内存不足了。发送方在发送全量数据时 path生命周期才可见

4.1.1 路线同步

在发送方与接受方的路线同步应使用专有的路线控制信息。此路线控制等信息定义了全量的路线信息

他们都存在于ADASIS Horizon中

当路线不存在时，路线控制信息会在接受方进行本地创建，目的是为了便于路线管理。如果路线出现在

接受方却没有出现在路线控制信息中，这种情况接受方会进行路线删除

可以看到发送方会规律性发送路线控制信息到接受方，以确保接受方可以同步路线信息（或容错同步偶然

丢失的路线信息）。发送方会发送额外的路线信息来通知接受方，或者通知接受方是否删除冗余数据，来为

新的业务路线数据创造更多的内存空间。（在车辆环境中会对循环发送额外路线信息时会有额外限制）

除此之外，在现有的path信息中包含了父路线与子路线的信息。其中也包括了父子路线分叉点的位置信息。

在这个信息中，包含了完成的几何树状结构。数据传输完成后。接受方将会有大量的去存储（特别是Profile Data）、

并且关联horzion tree中的必要定位信息。这样可以帮助接受者在发送方启动后快速启动。并且能够快速容错

请注意，一个空的路线信息需要接受的清除所有的horzon信息（除了全局信息）这也是在ADASIS V3中允许存在的

数据重置机制

1.数据提供者说明

任何时候，只要数据提供者给ADASIS Horizon中添加path，那么每个path必然会有一个id，这些信息都会包含在

path控制消息中

如果path有父级path，那么父path也会有个id（0代表根path的id）数据提供者使用parent pathId时，此path必须是存在

的，并且接收者已知。只有一种情况下存在父pathId改变，那就是父pathId 由null->0的情况，此情况下

必然是父类path被移除的场景。对于数据提供者来说也可以发送指定的父级别id去删除path

推荐做法是 数据提供者在发送额外数据到新的path之前要么等待路线控制信息传输完成，要么发送一个额外

路线信息并包含这个path。

在发送控制消息之前，允许发送一些path的属性数据与额外信息。就属性信息而言，这将会导致接收者在未

接收到控制消息时，就本地创建了一个path，这种path就称为隐式path创建（这也是路线控制信息中的标准

创建之一）provider必须要考虑receiver是否需要存隐式创建的path，如果receiver存储空间不足，那么provider

不应该触发receiver的存储操作。path 的起点来自于path中的 分割点，而这个分割点则来自于路段的分割点

或者是岔路口点。

provider将会决定是否要删除无用的path。当然了在ADASIS V3中，也并没有明确的定义要求车辆位置变化后

就要删除掉之前的path。

需要注意的是，如果车辆是在MMP上行驶时，那么这个path对象将会一直存在，当然了这也是推荐的做法，因为这样的故障率最低

2.接收者说明

当本地不存在path信息时，receiver将会在路线控制信息列表中创建path信息。（内存不足除外，如果内存不足

这种场景下将不会创建路线对象）

receiver也会删除在路线信息中不存在的path，与此path相关的数据也会随之删除 如果receiver收到的profile Data（含路线控制信息）中某个path在receiver本地不存在，那么reciver也会 自动创建此path（内存不足除外）

当receiver收到的数据中存在pathId在本地不存在的情况，receiver将会把这个id作为一个"灵活引用"。当

需要使用此pathId时，receiver将会检查本地是否存在。这个id会影响到 位置数据，父path数据，路线控制数据子pathid数据，道路id等。

4.1.2 Profile数据同步

profile数据同步是在provider和reconstructor之间，这说明profile数据中每个path数据都受控于horizon，

profile数据的管理与存储发生在provider侧，profile数据同步是在特定范围的位置下完成的。此范围来自于

profile位置控制器

profile控制位置指的是provider侧存储和使用profile实体时最小的为位置信息。在profile实体中，其他位置

信息都会比控制位置信息要大。在此范围内的额外的profile信息也是如此。此信息的使用场景是起点在范围之外

同时终点又在范围内。

所有范围呃逆的profile实体受控与provider并且会实时更新。因此，所有此范围内profile实体的使用和存储都

在reconstructor中完成。

规则：每个path中的profile control位都要通过profile 消息控制器来传输

provider发送Profile控制消息来更新profile控制位置

每当provider新增一个位置，就会通过Profile控制消息发送此位置到horizon数据重构器。之后，数据重构器就会

自动删除前一个位置。与其相关的path数据将会删除。

检测path数据的删除时机：

path数据通过Profile 控制信息来删除。此信用于更新受控path中的profile控制位置。只要发送了Profile控制信息

，如果有任意一个path新增了位置，那么provider发送此信息到reconstructor来更新位置。当接收数据信息

之后，reconstructor将会删除前一位置关联的所有path信息

必要时，数据重构器也会在特定path上不进行删除操作，当然了，这也必须是在有效的范围内（即范围终点必须是在

path控制位置之后）。此种场景下，重构器会保存更多数据。这是重构器的职能之一

4.1.3 path声明周期与path data生命周期分离点

在ADASIS v2中，系统会根据经过路段（滞后长度）的长路来选择进行部分数据移除。这种处理会体现在receiver中，

并作用于path列表中。当一个路口点处于滞后长度之后时，这个路口点所在的parent path将会被删除。

如图22，当path2上当前车位经过了岔路口100m后，岔路口所在的parentpath 将会被删除。

在ADASIS V3中，在provider侧，拥有更多详细的处理机制。这部分也是与receiver隔离的。

由于path数据受控于Path Control Message 而profile data属于path中的数据，因此path与path-data的生命周期

定义是在provider内部进行的。以下是一个路口上的两条岔路模型。并演示车辆在这两个平行路段行驶时，数据变化。

当车辆驶离岔路口时，provider会发送Profile Control offset 来删除path-data并且会通过Path Control Message

发送path item 来延长parent path的声明周期。如图所示，绿色的方框展示了在path1 path2上 Profile Control Offset相关信息

在这些绿色的区域内path是存在的。在蓝色的区域，则是要被删除的部分。如图所示，即便是child path（path2）的蓝色区域

被删除时,path1仍然存在。新的预览数据将会加入到path1与path2上直到provider进行其他操作。

有一种场景下会删除parentPath。当地图匹配过慢时。这种情况下，provider在发送Path Control Message时

不必引用parent pathId，这种情况下会直接触发所有Receiver在此path上的删除逻辑。

在为了便于理解上述案例中给到了个典型的父子路段的路网模型。但是实际路网结构中的机制会更加复杂。

注意：

发送Profile Control Offset 意味着通知Receiver删除当前path上指定位置所关联的相关path-data

发送Path Control item 到一个path上意味着要在Receiver侧延长某个path的生命周期

Path Control Message的数据结构的定义是为了便于传输和管理path列表与path之前的数据传递

4.1.4 动态生命周期计算。

在4.1.3章节中，我们看到了不同的生命周期定义可以对数据和逻辑数据对象"path"进行支持。因为这些

逻辑都是在provider侧进行定义的并且易于扩展。这些定义不必写死。通过实现一个算法动态生成数据的

生命周期这些也是非常有必要的

4.2.1 数据更新

profile实体类通过使用profile距离id来区分。在AHP第一次给AHR或应用传输profile数据时就会提供。此id便于AHR进行数据验证。只要profile数据变化，就说明接AHR接收到了实时交通信息。

1.数据更新与移除的场景：

1.传输ProfileMessage来创建一个新的profile数据

2.另一个ProfileMessage来更新数据

3.从profile中移除ProfileMessage数据。这样会在proflie中产生gap隔断，这个隔断是不受检查验证的如果需要进行验证。则数据将会被设为不可用

以上案例适用于spot、step 和linean profile数据

2.更新结束点和结束profile数据

在provider的数据供给时，path数据增长而profile数据不变时，更新end offset就非常有必要了。

先前的end Offset可以用于更新新的path。

当endOffsetFinal flag设置为true时，表明不会再接收到endOffset数据了。

或者endOffset和endOffsetFinal可以放在一个message中更新

4.2.2 数据范围更新

范围的更新取决于整个位置的范围。单个数据不涉及范围更新。因此只有含有特定距离id的数据进行

更新验证时，才涉及到范围更新。

在一个路线结构中range是与起点位置和终点位置相关的。范围更新是间隔进行的。包含起点不含终点，

范围的变更曲剧与以下几点：

1.在AHR中间隔执行时，数据不能为null，这种情况会到值操作不完整，使得数据与profile中的不一致。

因此profile 数据更新时应该对所有涉及起点和终点的数据都要作用到。

2.在ProfileMessage中范围更新的执行顺序也同样重要

a.删除先执行

b.更新操作在删除之后

3.为了避免profile数据断层，删除和创建操作应在同一个ProfileMessae中完成

交通信号profile 可进行范围更新

1.举例：在一个ProfileMessage中进行全量更新

更新一个range，所有的数据都要被删除。新数据的创建使用相同的ProfileMessage时，将会覆盖这个range

2.举例：Range失效

range数据移除时有2个方法可以完成此操作，以下案例展示了完成版的移除range和range更新为不可用场景

三个操作分别分在了3个ProfileMessage中

第一个表示range删除的是2个被移除的profile和一个空的profile

第二个表示range删除的是2个被移除的profile和一个不可用范围

第三个表示一次性移除所有与profile，proflie为空值

这个例子展示了从（1,2）移除时，可能存在的2种场景，一个Provider会根据profile类型来进行选择性移除

4.3[预测]内存管理

数据的边界在发送方和使用方都是提前定义好的。在reconstructor中会创建内存池来保存数据。当某个类型的

profile数据到达类型上限时，provider将不会发送。这就意味这reconstructor侧的数据受限于当前的存储的

数据与下一个massage数据。作为provider发送数据时，不应超过这个限制。而对于重构侧，数据上限则要包含

下一个mssage数据。

在标准ADASIS V3环境中。provier控制reconstructor侧的数据内存。provider也会通过Proflie Controller Message

和Path control 来进行数据的删除。基于这种机制，provider可以知道reconstructor是否已存储当前数据，但目前

Provider还并不能够判断reconstructor中剩余的空间是否足够下一个data使用。provider也需要知晓reconstructor

中总共有多少可用内存。

在ADASIS V3中，使用Franca开发文档可以给provier、reconstructor提供一个确切的内存大小说明。不过我们

仍需要解释下这些大小规格。

为了满足MISRA要求，我们使用类型安全的数据分派模型。每一种类型的数据都会被自动分配。我们使用一个

内存池并且池中的数据真实可用。数据项在池中分类到其数据类型，并返回池中。池中的每一个item数据类型

都是来自Franca开发文档中的描述定义

这里重点说明一下"动态分配"。在ADASIS v3中并不是所有的数据都需要动态分配。以下场景请注意：

1.当数据已指明大小时，数据不必进行动态分配

2.对数组来说，除了路线数组，其他数组都需要动态分配

3.对于多态的数据在实例化时，也需要再内存池中动态分配空间。

请注意内存池碎片化问题，France中存储一些数据集合。

在内存池中，所有动态分配的数据可通过数据类型进行区分

一个特殊场景，小数组将会被内联创建，这就意味着在闭合的数据结构中可以尽可能的最大限度分配数组空间

不是通过内存池中动态分配。当然了每个数组的长度上限是需要开发者根据Franca开发文档定的。在这之上，

数组会被定义为内联数组。

现在，根据这种算法，我们可以对同一时刻多个structure data实例进行限制。数据的分配规则定义在

ADASIS V3的 Franca标准文档中。而实际的数据可通过其实现类的指定。

到目前为止，对于无结构类型的数据的自动分配还没有一个对等的处理策略。由于大型的管理处理在

分配数据时是自动置顶的，因此那些无结构类型数据将会被设为内联处理。否则，将会很容易穷举出构建类型

Franca定义的全局数据类型很有必要使用。

在拥有了2中方式限制实际数据时，我们最终定义了限制标准：只有在ADASIS V3 horzion中出现的数据，

我们才进行计数定义。假设所有的Path Control messahge 已删除过期的path数据。所有的Profile Control

message 也完成了过期数据的清理。此种场景下，对于全局数据中每一个profile类型，我们会包含最近的

一个profile数据与最近的一个位置信息。在这些数据则是基于ADASIS V3 Horizon Provider创建的message

对象。这里必须要提一下内存池的大小。换句话说，对于已经存在的horzion数据与下一个将要被处理的

horizon数据都必须要满足reconstructor中的要求。-这个规则可用，但不强求。

请注意这里的描述，在实现的时候并不一定要按照说明的字面意思实现。实现类可以以任何方式管理这些

数据。这只是reconstructor存储 处理数据的能力，并保证数据正常。prodiver的数据发送能力基于描述中的说明。

4.4[预测]AHR 数据反馈

AHR（adas应用方）具备数据传递到AHP的能力，使用profile instanceId 可作为其标识。与AHP 到 AHR通讯流程类似，AHR 回传的feedback message与AHP中先关的数据是兼容的。因此，AHR也可以在数据变化时候，使用profileData中数据变化属性，并以feedback来进行AHR到AHP的数据传递。比如：AHR可以提供合法有效的profileData 比如限速数据。而且AHR也有可能去请求一个真实的profileData，并且给AHP提供一个已经观察到数据。而且AHR 肯能会请求AHP 进行数据proflie的添加或删除操作

4.5 与ADASIS V2 共存

基于车辆架构的不断更新，ADASIS V2 和V3 可能会存在于同一辆车上 ，因此就可能需要v2 和v3的provider ADASIS V2 和 ADASIS V3是可以共同独立存在的。ADASIS V2默认的传输协议是Can 总线而v3则使用的是其他通信媒介。不过即使ADASIS V2 是通过非Can媒介传输的，那么数据也会进行打包区分处理，用来区分此数据是来自v2 还是v3 ，通过这个手法来完成两种协议的独立传输。

这样做是有好处的，不必完全分离provider的实现而是保留一些horzion解雇，并且在其内部仅构建一次。这个Provider将会从ADASIS Horizon接口实现两个接口分别是V2的message 和V3的messahe，除此之外，还能为ADASIS Horzion节约算力，这个在V2 V3数据同步时候是很管用的。比如MPP 中的同步。

另一方面。很难做到仅有ADASIS V3的provider，并且要将数据从V3转为V2.这种转化通常不会再同一个级别的messag中完成而是要使用一个 ADASIS V3的 reconstructor和一个全量的ADASIS V2的 Horzion provider。在这种条件下也许可以转化成功。然而更好的方案则是上述所说的 策略模式生成V2 和V3 的provider。如果V2的传输未通过Can进行的，那么解包发送时，则需要用到CAN.

4.6[预测] 额外数据与传感器融合。

在ADASIS V3中，不必让一个horzion provider 提供所有的ADASIS V3数据。如图32 ADASIS V3 provider提供的

是路网集合数据和道路属性数据。除此之外，额外的horizon provider 提供额外的profile data。创建规则，系统

创建path时候使用规则一致。

在ADAS Horzion中不同的provider都会为其提供数据，这是传输巨量数据的一个重要指标。ADASIS V2中，核心

provider发送来自地图数据库中的道路稽核数据。但是在ADASIS V3中其他provider仅提供数据骨架。这些数据

都来自传感器或传感器单元。

现在，我们可以看到汽车的自动驾驶能力越来越完善，不就的将来车辆都将会具备无人驾驶的能力。基于此，

车辆就非常需要知道更为精准的突发环境信息，以及不同范围的前方道路预测信息。融合传感器的能力就是为

了提供这些信息。传感器融合可以聚拢来字传感器网络的互补或冗余信息，从而增强系统能力。传感器质量和

覆盖范围将作为预测算法的重要依据，从而能够处理更广范围内车辆自主驾驶情景。传感器融合的是个热题，

全球各大论坛、协会、工厂都在在讨论。传感器架构、级别、取决于设备制造商。

传感器融合是一门学科，本文档是远远覆盖不完的。然而ADASIS V3必须具备支持传感器融合的能力从而足够

应对各种融合结构。对此，ADASIS V3 不仅能够提供来来自数据库的静态数据，同时能够提供来自车辆传感器

的动态数据。

注- ADASIS 不会直接处理传感器数据，而是通过其灵活的架构能力能力支持传感器的融合、多种ADASIS V3 provider概念模型 如图33，此概念模型表明传感器融合数据在系统内来自多种数据源。（比如限速信息就来mapDb和摄像头侦测）每一种数据源都可以使用ADASIS V3的协议发送。因此，每种数据都会监听core provider发送的 horizon并且会加入一些特有的数据来丰富horizon。每种provider都可根据预测数据而类型或特有数据类型来自主定义自身。在ADASIS V3中，只有一限制性的条款是不可变的。对于未融合的数据需要使用自定义proflie并且与标准profile 对齐，这样一来，这种数据将会按照融合数据处理

可以想象多亏了ADASIS V3 架构的能力，我们可以兼容不同的传感器。比如，应用方可以获得多源数据，并处理自身的融合数据。或者由中控处理融合数据。

如上所述，这两相关的算法控制依赖于传感器的输出。然而ADASIS 中提供了2种类极为不同的处理方案。下行处理和上行处理。以下插图说明了这两个概念。

下端融合

这是一个比较保守的处理方案，使用方式V3与v2一致，地图就是一种特殊的传感器。可视范围内的Horizon就像广播一样穿过车辆。不仅仅是map数据，ADASIS V3也可以广播出来自其他传感器的信息。具体的传感器融合则按在client侧完成。因此成为下端融合

上端融合

第二种方法旨在细化和精确化车辆周边的环境数据。非常适合于自主车辆运行时的算法控制。因此，传感器融合处理

将会在provider侧进行处理。此种方案成为上端融合。传感器的输出数据常用于填充AH