Daniel Burgstahler, Martin Moebus, Tobias Meuser, Doreen Boehnstedt and Ralf Steinmetz: A Categorization Scheme for Information Demands of Future Connected ADAS. In Proc. Automotive meets Electronics (AmE), IEEE, 2017. ISBN: 978-3-8007-4369-8

A Categorization Scheme for Information Demands of Future Connected ADAS

Daniel Burgstahler, Martin Moebus, Tobias Meuser, Doreen Boehnstedt and Ralf Steinmetz
Multimedia Communications Lab (KOM), Technische Universität Darmstadt, Darmstadt, Germany
Email: firstname.lastname@KOM.tu-darmstadt.de

***Abstract*—Modern Advanced Driver Assistance Systems (ADAS) make use of a very high amount of sensors in the vehicle to detect the vehicle state and the vehicle's surrounding. The general goal of all ADAS is to improve or enhance safety, driving comfort and economy. In terms of connected ADAS and together with highly precise sensor information, sharing of the detected information becomes an important feature in modern vehicles. Therefore other vehicles have a long range predictive view available to their assistance systems, also moving forward to automated driving. In our solution we present a categorization scheme for such information demands with all important restrictions concerning the distance to an upcoming event. The categorization scheme is divided in three zones: Information, Awareness and Safety. Especially in the categories safety and awareness low latencies are required for information distribution. The category relates on distances of more than 30 seconds away from an upcoming event and can therefore be used with current mobile networks. For this category we provide a fully generic and efficient data structure for information exchange, which works to the best of our knowledge with every scenario.**

I. Introduction

Advanced Driver Assistance Systems (ADASs) play an intrinsic role for today's automotive development. Vehicles are equipped with all kinds of sensors, such as Radio Detection and Ranging (RADAR), Light Detection and Ranging (LIDAR), Global Positioning System (GPS) or cameras. The sensors are used for various safety or comfort features, such as an emergency brake assistant or Adaptive Cruise Control (ACC). In today's automotive application systems connectivity has been added as an important feature to the vehicles. Original Equipment Manufacturers (OEMs) like BMW or Opel introduced systems called BMW Connected drive [1] and Opel OnStar [2], which already allow access to a limited amount of sensor data remotely. The U.S Electric Vehicle (EV) manufacturer Tesla even allows Over-The-Air Software updates for their vehicles [3]. However, in the future V2V and especially V2X applications will become a core feature in modern vehicles, moving forward towards autonomous driving. For the communication technology the IEEE 802.11p standard has been introduced in 2010, which is a vehicular specified Wireless Local Area Network (WLAN) technology. However, today it is common that the LTE network is used for vehicular communication applications, since this does not require additional infrastructure hardware to be set up first.

Especially cellular networks like LTE are varying extremely in their available bandwidth depending on the current location and the amount of users logged in at the local Radio Frequency (RF) tower. Moreover at some locations their might even be no service available. Due to these varying conditions of connectivity it is important to clearly define how and when sensor information is transmitted and shared with other vehicles or infrastructural servers.

Connecting ADAS to the internet or ad-hoc with vehicles in the near surrounding improves and enhances every assistance feature by allowing a long range predictive driving. For example a just crashed vehicle could inform the directly behind driving vehicle about the crash, which is then able to avoid a collision. Additionally other vehicles behind the accident need to be informed as well, so that they can reduce their speed and are prepared. If the accident now leads to a traffic jam, also vehicles which are far away are supposed to be informed, so that they can exactly determine when to reduce speed before reaching the tail end of the traffic jam (c.f. figure 1).

With this scenario and especially the defined use-case scenarios in the ETSI standards (c.f. [1]–[4]), we have categorized three zones of information demands. The zone called "Safety" mainly handles time-critical communications for latencies between 2 and 5 seconds. As in the example mentioned before this would relate to the crashed vehicle and the vehicle(s) directly behind. The zone "Awareness" allows distances of up to 30 seconds to the event. In the zone called "Information", all communication purposes allowing latencies of more than 30 seconds are covered.

For the realization of a connected ADAS feature different communication technologies are available. The use-case definitions from the ETSI standards (c.f. [1]–[4]) therefore as an example have been categorized in the previous mentioned scheme. Additionally for each of the three zones a technology realization is presented, which relies on existent standardizations.

For the zone "Information", which accepts latencies of more than 30 seconds and does not have as much communication technology constraints as the other zones, a generic and message-size-efficient data structure has been developed. The structure is not limited to certain applications and works with all of our tested scenarios. Just as the well-known data structure of "Here" (c.f. [5]), it is implemented in Google Protobuf. The structure is categorized in data detection, data requests, data provision and data consolidation with an electronic horizon (eHorizon). With this generic division of data, new applications can be implemented without requiring any changes in the

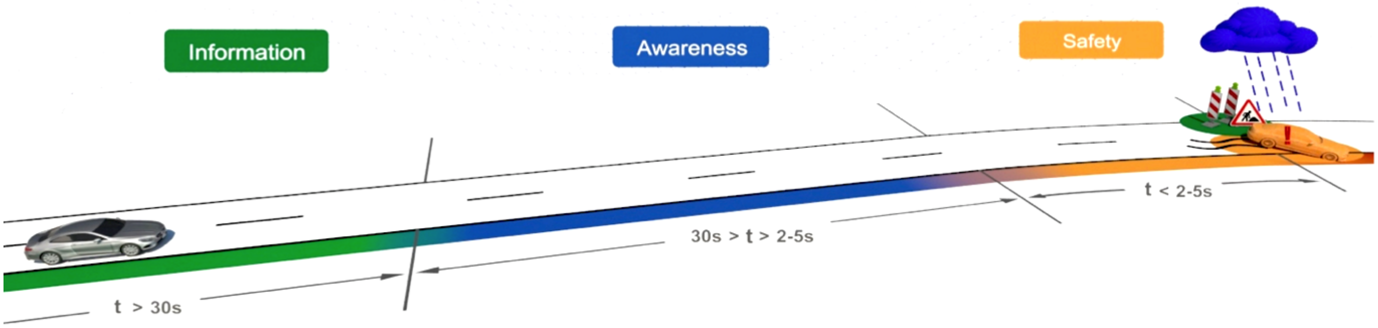

Figure 1: Visualization of our three derived zones of information demand for communication based ADAS. The classification is related to the maximum allowed latency from the earliest detection of information till the usage in the assistance feature.

structure definition, whereas "Here" is limited to a predefined data-set.

With our approach we contribute a detailed classification scheme for connected ADAS. For each of the three zones for information demands, different use-case scenarios and technology realizations are presented. Moreover for the communication between vehicles and backend-infrastructure a generic data structure is provided as Open Source.

In the next Section II we give a more detailed view on the three zones of information demand. Following this, we present our proposed data structure for data exchange with regard to the *information zone* in Section III.

II. Concept of Categorization

The concept of Vehicle-2-Vehicle (V2V) introduces many different technical demands, depending on the actual use case. We took all use cases from the European Telecommunications Standards Institute (ETSI), standards of ITS vehicular communication and Vehicle-to-Everything (V2X) applications. , and also relevant research publications [1]–[4], [6]–[8]. Furthermore we worked on a scheme to categorize these use cases. We separated them into three groups, namely *safety zone*, *awareness zone* and *information zone*. The decision criterion to separate between these zones is based on the tolerated latency between the appearance of an event and the moment the information has to be processed. Object dynamics and property are also considered in the categorization. Nevertheless the criterion is not sharp, which means an event can be relevant in more than one of the before mentioned zones, but information distribution might be different. Following this, we give a description of each zone.

A. Safety Zone

In the *safety zone* we need to handle highly dynamic information and high latency demands. For example use cases regarding collision prevention or collision risk warning. For the decision criterion we take over the ETSI time definition of 2-5 s as a maximal tolerated latency from the appearance of an event till the start of the counteraction. For the communication method it is preferable to use an ad-hoc communication with broadcast of messages and no connection establishment. This is due to the before mentioned high requirements in the *safety zone* and that cellular network availability can not be guaranteed. A message from the *safety zone* contains mainly the object position, velocity, acceleration, heading and steering angel of vehicles. Due to the fact that the remaining time till the event location is 2-5 s, it is not possible to send a warning signal to the driver. Therefore the system has to act autonomously to the given use case.

B. Awareness Zone

In contrast to the *safety zone*, we have to deal in the *awareness zone* with dynamic information in not safety relevant situations. But most essential information content is still vehicle position and dynamics, however, with lower demands. These use cases need attention right away or reaction of the driver. For that reason we take over the ETSI time definition again and get a lower limit of 2-5 s and an upper limit of 30 s. Example use cases are cooperative driving maneuvers and cooperative adaptive cruise control. These use cases have gotten even more attention recently, since the program *New Vehicle and System Technologies* of the German Federal Ministry for Economic Affairs and Energy[1] indicates that there is a need for communication technologies which make it possible to have a fast, save and reliable cooperation between vehicles. Also the project *IMAGinE*[2] (Intelligente Manöver Automatisierung - kooperative Gefahrenvermeidung in Echtzeit) showcases how important these technologies are for future implementation of autonomous driving. In these use cases we need to know dynamic information in a relatively long-distance range. Therefore we can use multi-hop ad-hoc or cellular communication. However ad-hoc communication is to prefer because of the availability of cellular network, that can not be guaranteed and the need of a low latency for the dynamic information. Typical information that has to be send

[1] Research and innovation framework *New Vehicle and System Technologies*, Federal Ministry for Economic Affairs and Energy (BMWi), http://www.tuvpt.de/fileadmin/downloads/bmwi_Neue_Fahrzeug-_und_Systemtechnologien_2015_s06.pdf, last accessed and validated in January 2017

[2] Press release, 'BMWi startet Förderung des Großforschungsprojekts IMAGinE für kooperatives Fahren in der Zukunft', http://www.bmwi.de/DE/Themen/technologie,did=779040.html, last accessed and validated in January 2017

is position, velocity, acceleration, heading, etc., which comes from all objects in range. Due to the fact that the upper time limit is now 30 s, the ego vehicle can't communicate directly with the vehicle of interest, so that it has to interact with vehicles in between. Trough this multi hop communication the intermediate vehicles combine their own information with the information of interest.

C. Information Zone

Apart from the other zones, the *information zone* contains no dynamic information. We only use static or semi static information that is not safety relevant. These use cases only inform the driver and do not need any direct response from him. Usually this zone contains use cases that affect the route, e.g., traffic sign information. For the limit we take over the ETSI time definition, so that we have a lower limit of 30 s. Often an objective for these use cases is to decrease the energy consumption and to raise comfort. The requirements in latency in the *information zone* are low, compared to both other zones, so that we can use multi-hop ad-hoc or cellular communication. Because of the fact that between the point of information and the actual vehicle are long distances, the cellular communication is preferable. The tenor of the message is static object information, such as traffic signs and their range of validity. The fact that cellular communication is costly leads to minimizing the amount of transmitted data.

III. Data Structure for Information Demands

For the long range information zone, a data structure for an efficient communication between vehicles and a backend infrastructure has been developed in Google's Protocol Buffers (protobuf). The structure is depicted in Figure 2. It is divided into 5 main components of which a message can consist. "RequestTags" are used to request either sensor information or events from a vehicle. Messages with RequestTags as message content are usually sent from backend servers to a vehicle or a group of vehicles. (c.f. section III-B) "Tags" are usually the response of a vehicle to "RequestTags". Tags include SensorValues or also Events, which a vehicle has determined. (c.f. section III-C) Since the server backend usually also provides information to vehicles, this can be transmitted through the message content "Information" (c.f. section III-E). For requesting such information form a backend server or any other entity, the message content "RequestInformation" is used (c.f. section III-D). Additionally within a message it is also possible to transmit an electronic horizon (eHorizon) within a message. This can be useful if highly precise map material is available on a server and is provided to a vehicle. This allows to transmit attribute enriched path information. (c.f. section III-F)

A. Message Header

In general, the complete data structure is designed with optional values. This means, that every field and every message content is optional within every message. This allows to use the data structure to the best of our knowledge with every use case in the long range information demand zone. The header of every message can consist of an unique identification (ID), a timestamp, a priority level and a sender identification. The sender identification can either be also an randomly generated universal unique identifier (UUID) or in case the sender is a vehicle, the vehicle identification number (VIN) can be used as sender identification. Like mentioned before, the message itself can then consist of "RequestTags", "Tags", "RequestInformation" fields, "Information" fields and eHorizons.

B. Request Tags

RequestTags are used to request sensor values or event information from a vehicle. Every RequestTag can consist of an unique identifier (ID). This can be used for later responses to a previously received request. Therefore, when later sending a tag, which is sent because of a previously received request, the ID in the tag can be set to the RequestTag ID, which will simplify the assignment on the receiver side. Additionally again a timestamp and a priority can be set. As mentioned before, it is possible to request sensor values and event information. An approach for efficient gathering vehicular sensed data by the use of probabilistic transmission is presented in [9].

1) Event Request: When event information is requested, the event is identified by a previously defined event identification number. Additionally, limits for how long the request is valid (duration) and/or how often the event needs to occur until the request is invalid.

2) Sensor Request: Also when sensor values are requested, each sensor is identified by a previously defined sensor identification number. Here, it is also possible to define an interval in which sensor values should be transmitted, besides a duration and maximum amount.

C. Tags

Tags are used to either respond with the desired sensor values or triggered events after a previously received request or to simply transmit such information without a request, depending on the application use case. In the header of a tag, it is possible to set a response identification. This field should be used to clearly assign a tag to a previously received RequestTag, so that the tag is treated as a direct response. Additionally, it is again possible to add a time stamp and a prioritization to the tag. Like mentioned before, a tag can consist of an event or sensor values. If it consists of sensor values and and event, this could be treated as an event, which is connected to sensor values. For example, if a speed limit has been detected (event) and the sensor values hold the location where it has been detected. The events need to be previously defined as event type identification numbers. Since sensor values can consist of either integer values, floating point values or even text based values, an optional field is available for each type, since it would be inefficient transmitting an integer number through a floating point data field. The unit of the sensor value is set by a previously defined unit identification number and also the sensor type is set by a previously defined sensor type identification number. If sensor values are somehow location related (e.g. latitude, longitude or heading from a GPS), a flag called locationRelevant can be set. Additionally it is possible to add accuracy information to each sensor value. Here, the accuracy can also be set as an integer, floating point or even text based value and the same unit identification number reference system can be used.

D. Request Information

If a vehicle or a backend server wants to request already processed derived information from an entity, the message content "RequestInformation" is used. In the header of every "RequestInformation" field it is again possible to set a request identification number, so that the receiving entity can directly respond to a request. Additionally, a time stamp and a prioritization can be set. The most important field of the "RequestInformation" header is the information type ID. With this previously defined identification number the desired information is declared, so that an entity knows which information it should send. Sometimes it is also useful, if the information request consists of parameters. For example if the outside temperature at an certain location is requested. Therefore a location parameter can be added to an information request. In another case it might be useful that a sensor value or a set of sensor values is added to an information request. For example when requesting an eHorizon, it might be useful to add sensor values such as the current speed and also the location (c.f. section III-F). The sensor value parameter is implemented identically to the sensor value parameter of a tag (c.f. section III-C).

1) Location: The location parameter can be used in many different ways. If the application scenario requires an unique identification in the transmitted locations, this can be set with the location ID field. Additionally it is also possible to set a reference to map material. This can be done through the fields map ID. Here, an integer field and text based field are provided, which are meant to be used in a way that it fits to the reference map material. If the reference in the location is mapped to a road segment, the field segment offset can be used to define the exact position on this segment. If a location parameter consists of a simple point, the fields latitude, longitude and optionally also the heading can be used to define the location. For an even more precise location, also the lane of road segments and the altitude of a point can be set. If the location parameter should define an whole area, the radius field can be used. Sometimes it might also be desired to transmit a polygon of location points. This can be done by adding recursively child locations to each location point. This also allows to define areas, which are not circular, by setting the child of the last location point to the location point of the first location. This results in a closed polygon line. Polygon lines can also be generated with a radius for each point, which also allows to define areas. Since it might be also useful for some application scenarios to know the preciseness of a given location, an accuracy can be set as well to each location point.

E. Information

For transmitting any already processed data between entities, the message content "Information" is used. In the header of an Information field it is again possible to set a response ID, which correlates to the request ID of a previously received information request. As with most other message content fields, it is also possible to add a time stamp and prioritization to the information transmitted. For defining of which information the message consists, an information type identification number can be set. This needs to be defined ahead of time, so that sender and receiver know how to process the information. Optionally each information field can also have connected information values. For example if an information type is speed limit, the value could be 100 km/h. All values can again be stored as integer, floating point or text based values and an accuracy and unit ID can be set by using previously defined unit identification numbers. Since one information can hold multiple values, it is also possible to add an identification number to each value. It is also possible to use the in section III-D described location field for connecting a location structure to an information field. For some use cases it also might be of interest to connect information field with each other or to have a hierarchical structure in the information. For example if traffic signs are transmitted they can be connected to child information, such as speed limits, stop signs, yield, etc. Therefore the message content "Information" provides the data field connected information, which allows to connect multiple information fields with itself.

F. Electronic Horizon

The last possible message content is the eHorizon. The eHorizon can include a road network, which is usually determined in front of the vehicle. Usually a server calculates the most probable path (MPP) to the location of vehicle is currently driving on. The transmitted road network then includes up to a certain length, based on various parameters, such as speed and road type, the MPP and also road segments which are intersecting with the MPP. Therefore the eHorizon message content allows the implementation of a server based eHorizon provider, as presented in [10]. Each eHorizon can have an unique identifier and since multiple implementations and types of eHorizons for different network connectivity types exist, an eHorizon type can also be specified in the message header. For most use cases it is of interest to enrich the eHorizon path with attributes, such as speed limits or danger zones. Therefore the complete eHorizon can be connected with Information fields. This is usually global information, which holds true for the whole eHorizon. The eHorizon can be generated as a hierarchical tree structure, where the nodes represent road segments between intersections in the map material. Therefore each eHorizon is connected to a set of nodes (road segments), which build the tree. Each TreeNode (path segment) gets its identification number within the tree. For setting up the hierarchical tree structure, within each tree node the identification number of the parent tree node (parent road segment) is stored, which again correlates to the tree node ID of this segment. Each node (road segment) in the tree can also be enriched with attributes, by connecting a node with a set of information fields. This sis also used to define the start location and end location of a road segment or to define its reference in the map material. Of course it is also possible to use the information field with the location parameter for describing the course of the road segment with a polygon line. A more efficient way to transmit the course of the eHorizon path is the use of B-Spline approximation. Therefore each tree node can have the B-Spline parameters (knot vector and degree) set in the header. Since the MPP is usually also predicted, the probability for each (node) road segment can also be given within tree node structure. Also it is defined whether a node (road segment) is part of the MPP or not. Many implementations of eHorizon providers are possible through this structure. One efficient and working solution is presented in [10] and the source code is available under Apache 2.0 open source license [11].

IV. Conclusion and Outlook

Within this work we have presented our use case categorization approach, that harmonizes use cases presented in literature and standardization with regard to communication. The mentioned three zones, namely *safety zone*, *awareness zone* and *information zone*, have different demands on information propagation, in particular with respect to the maximum tolerated latency between information occurrence and processing in the respective ADAS feature. Whereas communication mechanisms for the *safety zone* are already discussed in literature and standardization, mechanisms for the other zones are open issues. Within the *awareness zone* an extended perception on neighboring moving objects is needed. Here an efficient multi-hop propagation mechanism would be required. For realization the propagation of position beacons and tracking information has to be much more efficient compared to the propagation of the *Cooperative Awareness Message (CAM)* or *Basic Safety Message (BSM)* [**?**], [12]. This will be a requirement to realize a respective tracking information forwarding. Use cases with regard to the *information zone* can be realized via cellular communication because of a much higher tolerated information propagation latency. A logically centrally (not necessarily physically central) server or cloud can gather and propagate respective information. An approach which can also deal with sparse traffic situations. For the respective information exchange we introduced our data structure that is already used within our eHorizon prototype [10], [11].

Acknowledgments

The work presented in this paper was partly funded by the German Federal Ministry of Education and Research (BMBF) under grant no. 01IS12054.

References

[1] "ETSI TR 102 638 V1.1.1 (2009-06) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions."
[2] "ETSI TS 101 539-1 V1.1.1 (2013-08) Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification."
[3] "ETSI TS 101 539-3 V1.1.1 (2013-11) Intelligent Transport Systems (ITS); V2X Applications; Part 1: Longitudinal Collision Risk Warning (LCRW) application requirements specification."
[4] "ETSI EN 302 895 V1.1.1 (2014-09) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Local Dynamic Map (LDM)."
[5] HERE, "Vehicle Sensor Data Cloud Ingestion Interface Specification v2.0.2," 2015.
[6] F. Bai, T. ElBatt, G. Holland, H. Krishnan, and V. Sadekar, "Towards Characterizing and Classifying Communication-based Automotive Applications from a Wireless Networking Perspective," in *Proceedings of the first IEEE Workshop on Automotive Networking and Applications (AutoNet)*, Dec 2006, pp. 1–25.
[7] C. Weiss, "V2X communication in Europe - From research projects towards standardization and field testing of vehicle communication technology," *Computer Networks*, vol. 55, no. 14, pp. 3103–3119, 2011.
[8] C. Campolo, A. Molinaro, and R. Scopigno, Eds., *Vehicular ad hoc Networks: Standards, Solutions, and Research*. Springer, 2015.
[9] D. Burgstahler, T. Meuser, U. Lampe, D. Böhnstedt, and R. Steinmetz, "ProbSense.KOM: A Probabilistic Sensing Approach for Gathering Vehicular Sensed Data," in *Proceedings of the 3rd International Conference on Mobile Services (MS 2016)*. IEEE, Jun 2016, pp. 9–16.
[10] D. Burgstahler, A. Xhoga, C. Peusens, M. Möbus, D. Böhnstedt, and R. Steinmetz, "Remotehorizon. kom: Dynamic cloud-based ehorizon," in *GMM-Fachbericht-AmE 2016–Automotive meets Electronics*. VDE VERLAG GmbH, Mar 2016, pp. 46–51.
[11] "Horizon.KOM and eHorizon.KOM – Open Vehicular Horizon Provider." [Online]. Available: http://www.kom.tu-darmstadt.de/research-results/software-downloads/software/horizonkom/
[12] "IEEE 1609 Family of Standards for Wireless Access in Vehicular Environments (WAVE)."
[13] "ETSI EN 302 665 V1.1.1 (2010-09) Intelligent Transport Systems (ITS); Communications Architecture."

All online references in this paper were last accessed and validated in January 2017.

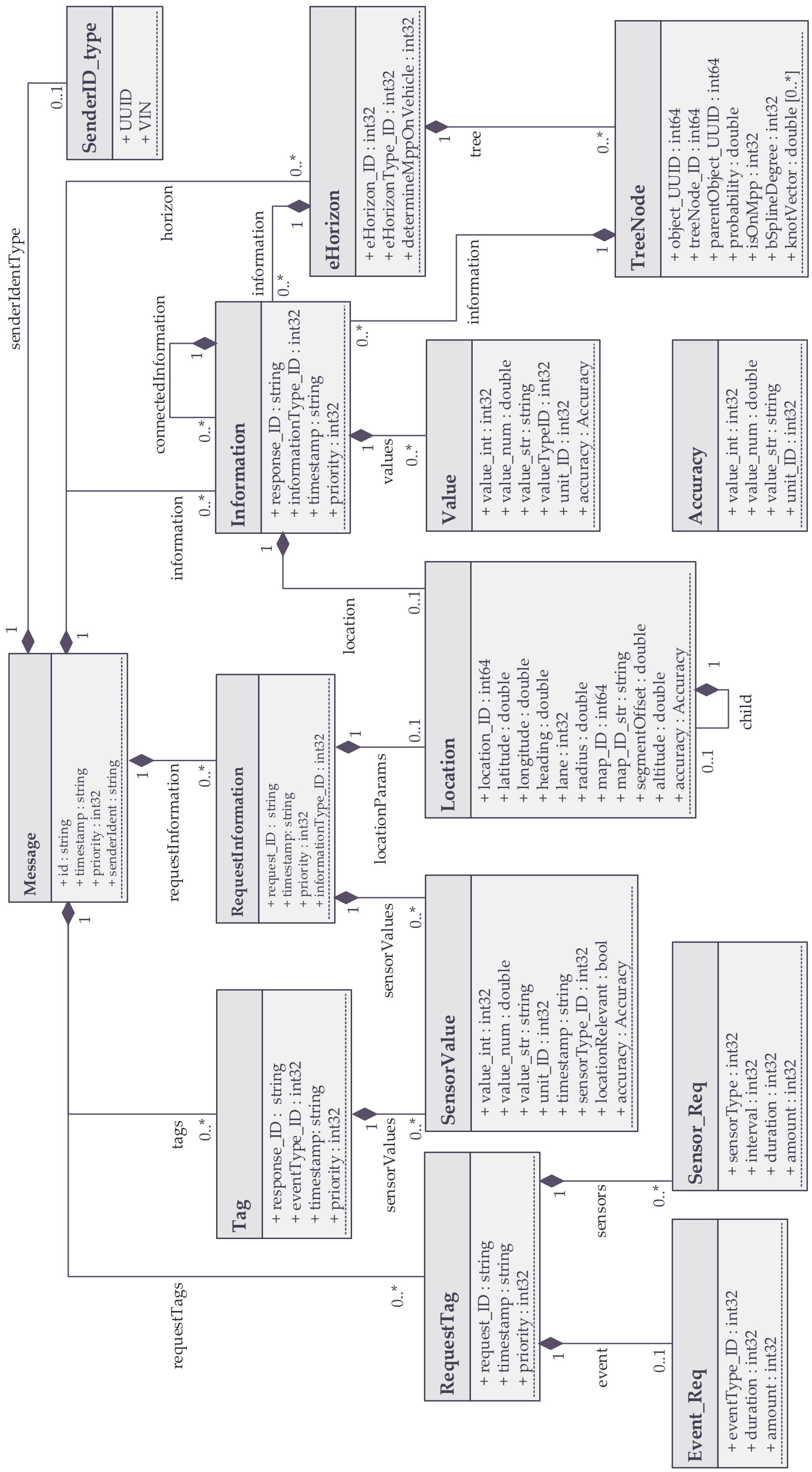

Figure 2: UML diagram of our proposed data structure.