HAFAS Real-time Exchange (HRX)

Interface Specification

Version 2.4.8

Date February 8th 2019

Rainer Arz, Marc Föll, Wolfram Fiekert
HaCon Ingenieurgesellschaft mbH
Lister Straße 15
30163 Hannover
Germany

Contents

Introduction

HAFAS Real-time Exchange (HRX) is a real-time information interface and format developed and maintained by HaCon Ingenieurgesellschaft mbH for the purpose of mutual exchange of real-time information between HAFAS systems and other parties.

The scope of the HRX interface is

- Real-time information on trips
- Real-time information on connections
- Real-time information on connection protection
- Real-time information on trip occupancy and train formations
- Real-time information on vehicles

1 Basic Technical Definitions

The HAFAS Real-time Exchange (HRX) interface is a push oriented server to server interface using XML/SOAP and HTTP for message exchange.

1.1 Server to Server

The communication between the “connected” or peered systems is such that each peer only knows the endpoints of the other peer and vice versa.

For example consider the communication between a HAFAS Smart ITCS system and a HAFAS journey planner system.

HAFAS Smart ITCS would be represented by the endpoint http://smartitcs.hafas.de/hrx/, the journey planner system could be represented by the URL http://planner.hafas.de/hrx/

All communication between the two peers is done exclusively via the respective endpoints via HTTP.

For certain types of information (real-time on trips, vehicle data) further sub paths for the endpoints can be defined on a per project or product base.

1.2 Push Oriented

The peer systems only push those pieces of information that they judge as relevant for their peer to the peer’s respective endpoint. There is no active requesting of information from either side.

Which type of information (trip information, vehicle information, connections, etc.) is exchanged and how the information is filtered (e.g. by operator) has to be settled in per project or product peer agreements.

For the above example the HAFAS Smart ITCS would push information like trip route and prognoses to the journey planner system. But the HAFAS Smart ITCS peer would only do so if a relevant change has happened – in this case for example if the prognoses for a trip have changed.

1.3 XML/SOAP

The messages exchanged between to HRX peers use the XML formats defined in the accompanying XML schema definition and WSDL files.

The XML examples given in the following omit XML namespace declarations and XML namespaces for the sake of readability.

1.4 Complete Information Set

For each HRX message a response is sent in the form of a RealtimeResponse element. The response contains the attribute serviceStartTimestamp which denotes the time of the start of the responding service.

By tracking the start time of its peers a peer can detect restarts of the underlying services at a peer.

For each pair of HRX peers it has to be defined if the peers must detect a restart at the other peer and then send a complete set of current information to the peer.

1.5 Filters and Mappings

1.5.1 Mappings

Both sides of a HRX communication might use different identifiers to reference stations, product classes, line numbers, etc.

In addition both sides of a HRX communication might use a different granularity in their identifiers. For example:

- One side might have detailed knowledge about station poles and use different identifiers for each pole where the other side only has one identifier for the whole station.
- One side might know individual trip numbers for trips on the same line while the other side only knows line numbers and start times of trip.

For each pair of HRX peers it has to be defined on a per project or product base which identifiers have to be mapped by which peer.

1.5.2 Filters

Both sides of a HRX communication can apply filters for the information they send out to the peer. Most common filters will be operator or category based filters which restrict the scope of real-time information sent to the peer to certain operators or categories.

For each pair of HRX peers it has to be defined on a per project or product base which filters will be used.

2 Use cases for HRX

A general overview of the different use cases for HRX.

2.1 Real-time information for a journey planning system

In this use case a system which is capable of generating, collecting or consolidating real-time information from public transport (e.g. a vehicle management system) will provide the information to a journey planning system.

The relevant information for a journey planner is sent trip wise in RealTrip elements, s. Real-time Information on Trips.

2.2 Vehicle Management System

2.2.1 Basic Description

This describes the use case where one HRX peer in the role of a vehicle management system manages trips and vehicles which are not directly connected (e.g. through some proprietary protocol) to it, but to a 3rd party backend system.

HAFAS Smart ITCS is an example of a vehicle management system that is capable of using a HRX interface with 3rd party backend systems.

The HRX interface and the 3rd party system take the role of a mediator to e.g. the proprietary interface used between the 3rd party backend system and the vehicles.

For the following description the abbreviation VMS is used for the HRX peer in the role of the vehicle management system. The 3rd party system is simply called the other system.

The primary source of information about the vehicle (current trip, current position, etc.) is HRX RealTrip elements.

2.2.2 Multiple Tenants, Real-time Data Identifier

The vehicle management system may be used to monitor vehicles from different “tenants” (different clients / operators / contractors / sub-contractors) represented by different 3rd party systems and HRX peers.

The vehicle management system (e.g. HAFAS Smart VMS) should then be able to cleanly separate data from different tenants, e.g. when presenting vehicle related information to a user.

In this use case the HRX peers for the vehicle management system and the 3rd party systems have to mutually agree on a set of “real-time data identifiers”. These real-time identifiers have to be used consistently in all communications between HRX peers; they are represented in the sender attribute of the HRX RealtimeInfo root element.

2.2.3 Identifying Vehicles

A 3rd party system must consistently use the identifiers for the primary on-board units in the vehicles for which it provides and receives data. The identifiers for the primary on board unit have to be distinguished from a vehicle identifier. The reason her is that multiple devices may be present in a vehicle, the vehicle identifier serves to logically link the different devices in one vehicle.

Examples for an identifier of an on board unit are:

- International Mobile Equipment Identity (IMEI)
- Media Access Control (MAC) address

Generally an identifier that does not change during the lifetime of an on board unit.

The primary on board unit identifier must always be given in the Real-Trip.TripRef.TripID.UniqueID element for trip related information, whereas the vehicle identifier (if known) has to be passed in the RealTrip.VehicleID element.

The vehicle identifier must be used consistently in data collected from the vehicles and delivered to the vehicle management system by 3rd parties.

2.2.4 Types of Information

The VMS will usually receive the following types of information from the other system:

- geo positions of vehicles
- stop arrivals and departures of vehicles
- wait requests and wait confirmations
- vehicle data like occupancy data, telemetry data, door events

The VMS will usually send the following types of information to the other system:

- planned / schedule information on a trip, including short term deviations from the planned trip route
- real-time information on trips, especially the vehicle's progress on the trip
- wait recommendations and wait reports

2.2.5 Start and end of a trip

The other system must inform the VMS when a certain vehicle is starting or ending a trip.

The start of a trip is signalled by sending a RealTrip element with TripRef sub element filled as agreed on for the project or product.

The end of a trip is signalled by sending a RealTrip element with TripRef sub element filled such that it does not reference a trip. Example:

```
<RealtimeInfo version="2.4.3" timestamp="2016-11-30T15:48:44+01:00">
    <RealTrip>
        <TripRef>
          <TripID>
```

```
            <TripName></TripName>
            <OperatingDay>2016-11-30</OperatingDay>
            <UniqueID>5642</UniqueID>
          </TripID>
        </TripRef>
    </RealTrip>
</RealtimeInfo>
```

2.2.6 Tracking Vehicles

The vehicle management use case allows for tracking vehicles which are currently not conducting a planned trip. By simply adding a RealPosition sub element to a RealTrip element without referencing to a planned trip the other system can inform the VMS about the last known geo position of the vehicle. Example:

```
<RealtimeInfo version="2.4.3" timestamp="2016-11-30T15:48:44+01:00"
xmlns="urn:hrx">
    <RealTrip>
        <TripRef>
          <TripID>
            <TripName></TripName>
            <OperatingDay>2016-11-30</OperatingDay>
            <UniqueID>5642</UniqueID>
          </TripID>
        </TripRef>
        <RealPosition>
          <Xcoordinate>2.439788</Xcoordinate>
          <Ycoordinate>50.785646</Ycoordinate>
          <RealDeparturePrediction>2016-11-
30T15:48:43+01:00</RealDeparturePrediction>
        </RealPosition>
    </RealTrip>
</RealtimeInfo>
```

Starting with version 2.4.5 HRX allows using a GeoPosition sub element in a RealTrip element. GeoPosition additionally allows for speed and bearing of a vehicle, information which usually is present in date from GNSS receivers. Example:

```
<RealtimeInfo version="2.4.5" timestamp="2018-01-09T15:48:44+01:00"
xmlns="urn:hrx">
    <RealTrip>
        <TripRef>
          <TripID>
            <TripName></TripName>
            <OperatingDay>2018-01-09</OperatingDay>
            <UniqueID>5642</UniqueID>
          </TripID>
        </TripRef>
        <GeoPosition>
          <Xcoordinate>2.439788</Xcoordinate>
          <Ycoordinate>50.785646</Ycoordinate>
          <Timestamp>2018-01-09T15:48:43+01:00</Timestamp>
          <Speed>50327</Speed>
          <Bearing>217</Bearing>
        </GeoPosition>
    </RealTrip>
</RealtimeInfo>
```

HRX allows mixing RealPosition and GeoPosition elements but that usually be avoided for consistency.

2.2.7 Collecting Vehicle Data

A typical use case for vehicle management systems is to collect various types of data from the managed vehicles:

- Occupancy data, e.g. the number of boarding and alighting passenger at a certain trip stop.
- Door events like opening or closing a door.
- General telemetry data like current odometer value or fuel level.

In HRX these types of data are modelled in the OccupancyData, DoorEvent and Telemetry-Data elements.

2.3 Connection Protection

This describes the exchange of data / messages between HRX peers for the case where fetcher trips managed by one HRX peer are expected to perform connection protection for feeder trips managed by the other HRX peer.

HRX peers can establish connection protection between them by exchanging messages with

- wait requests
- wait recommendations
- wait confirmations

For each pair of HRX peers it has to be agreed per project or per product which peer is responsible for which fetcher trips and which peer is responsible for which feeder trips. Typically this is based on selecting feeders and fetchers by transport operators.

2.3.1 Wait Requests

Wait requests are sent by a HRX peer in order to explicitly state the need that a specific fetcher trip should waiting for a specific feeder trip.

Examples use cases are:

- One HRX peer manages a vehicle acting as feeder on a certain trip. When at some point of the trip a passenger explicitly states the wish to use a certain connection a wait request for this connection can be sent to the other HRX peer.
- One HRX peer in a journey planner system can issue wait requests to the other HRX peer in case a passenger plans a journey or buys a ticket for a certain journey.

Optionally along with the wait request a HRX peer can send the number of transfer passengers that are planning to use the connection.

2.3.2 Wait Recommendations

Wait recommendations are issued by a HRX peer managing one or more feeder trips that have a connection to a certain fetcher trip.

2.3.3 Wait Confirmations

Wait confirmations are sent by a HRX peer managing a fetcher and explicitly state if or if not a fetcher is waiting for specific feeders in connections.

2.4 Real-time Information for Traffic Light Control Systems

In this use case real-time information is sent to a system capable of controlling / influencing traffic lights based on real-time information.

The HRX peer producing the real-time information will probably be in the role of a vehicle management system managing trips and tracking the geo positions of the vehicles it manages.

The HRX peer in the role of a vehicle management system will send HRX messages containing RealTrip elements with RealPosition sub elements containing a reference to nearby traffic lights.

3 Basic XML Structure

3.1 Root Elements

All messages exchanged between HRX peers follow the same basic scheme: The sender sends a <RealtimeInfo> element with sub elements modelling the various types of information HRX supports.

RealtimeInfo		XML attributes
timestamp	mandatory	Generation time of this message.
Version	mandatory	HRX XML schema version
sender	optional	The sender attribute is mandatory in cases where the receiving peer has to support simultaneous communication with more than one HRX peer. When used each of these peers has to always supply the same unique identifier in this attribute. When used the concrete values for the sender attribute have to be agreed on per project and installation.
resetRTData	optional	If the resetRTData attribute is given and set to true, the receiving peer must delete all accumulated real-time data previously received from the sending peer.
fullRTDeliveryStart	optional	In cases where a complete delivery of real-time information to a HRX peer is spread over separate HRX message and where the processing of the complete delivery should not start before the last part was received by the peer this attributes denotes the first part of a complete delivery.
fullRTDeliveryEnd	optional	This attributes denotes the corresponding end of a complete delivery. The sending peer must ensure that all parts are sent and received before this part is transmitted.
RealtimeInfo		XML sub elements

	RealTrip	optional, multiple	
	OccupancyData	optional, multiple	
	TelemetryData	optional, multiple	
	DoorEvent	optional, multiple	

The receiving system responds with a <RealtimeResponse> element.

RealtimeResponse			XML attributes
	timestamp	mandatory	Time of reception.
	Version	mandatory	HRX XML schema version
	serviceStartTimestamp	optional	Timestamp of last restart of the receiving system.
RealtimeResponse			XML sub elements
	ErrorNumber	optional	Error code, to be defined on a per project basis
	Errortext	optional	Textual description of error

3.2 Common XML Elements

3.2.1 TripRef

The TripRef structure as a whole is used to identify a timetable trip instance on a specific operating day. The actual content is project or product dependant and has to be agreed on between HRX peers.

TripRef			XML sub elements
	TripID	mandatory	Trip identifier.

	TripStartEnd	optional	First stop with (scheduled) departure time, last stop with (scheduled) arrival time; information is used in receiving system to identify (“match”) the real-time trip with planned schedule.

TripID			XML sub elements
	TripName	mandatory	A number identifying the trip (for the given day of operation), typically a train number or a trip number. In case a line number is used the TripStartEnd element must be given.
	OperatingDay	optional	Day of operation – typically coincides with date of departure time at first station.
	UniqueID	optional	UniqueID is used in cases where the sending system can give a stable identifier for the referenced trip. Example use cases are: 1. The unique ID is a per day identifier of the actual trip the real-time information is sent for. The unique ID may be known in the timetable data of the sending and the receiving system, but this is not necessary. 2. The unique ID uniquely identifies the primary onboard device in the vehicle that conducts the trip the real-time information is sent for.

TripStartEnd			XML sub elements
	StartStopID	optional	Identifier of start station. Hint: can be mapped by a station mapping table. This does not necessarily have to be the very first station of the route. At least one of the fields “StartStopID” or “EndStopID” has to be specified.
	StartTime	optional	Departure time at start station (if specified, then “StartStopID” is mandatory).
	EndStopID	optional	Identifier of destination station. Hint: can be mapped by a station mapping table. This does not necessarily have to be the very last station of the route.

			At least one of the fields “StartStopID” or “EndStopID” has to be specified.
	EndTime	optional	Arrival time at destination station (if specified, then “EndStopID” is mandatory).

3.2.2 LocalizedStringType

The LocalizedString structure describes an optionally localized string value.

LocalizedStringType			XML sub elements
	Language	optional	XSD language code for the localized string.
	String	Mandatory	Actual value.

3.2.3 StopRef

StopRef references a stop in a trip.

StopRef			XML sub elements
	StopID	mandatory	Identifier of the stop.
	Arrival	optional	Planned arrival time at the stop.
	Departure	optional	Planned departure time at the stop.

3.2.4 GeoCoordinates

GeoPosition			XML sub elements
	Xcoordinate	mandatory	Longitude; x-coordinate (WGS84-system) of current position in decimal degree notation.
	Ycoordinate	mandatory	Latitude; y-coordinate (WGS84-system) of current position in decimal degree notation.
	Zcoordinate	optional	Altitude; z-coordinate of current position in meters above sea level.

3.2.5 GeoPosition

This is an extension of GeoCoordinates.

GeoPosition		XML sub elements
Xcoordinate	mandatory	Longitude; x-coordinate (WGS84-system) of current position in decimal degree notation.
Ycoordinate	mandatory	Latitude; y-coordinate (WGS84-system) of current position in decimal degree notation.
Zcoordinate	optional	Altitude; z-coordinate of current position in meters above sea level.
Timestamp	mandatory	Timestamp of when the geo position was acquired. In case of GPS like position determination it is strongly recommended to use the original GPS timestamp as provided by the GPS / GNSS device.
Speed	optional	The current speed in meters per second.
Bearing	optional	The current bearing in degrees (0 to 360).
Quality	optional	The quality of the geo position. To be defined on project or product baiss, for example the HDOP (“horizontal dilution of precision”.
Source	optional	Source of the geo position, e.g. GNSS for satellite navihation.

4 Real-time Information on Trips

Real-time information on trips is delivered via RealTrip elements.

RealTrip		XML sub elements
LineID	optional	The LineID can be used to match the with the long-term schedule line in the receiving system.
DirectionID	optional, multiple	Typically identifies the outward or return direction of a line. Hint: If more than one DirectionID is specified, the validity ranges of DirectionIDs – specified by attributes “fromStopID” and “toStopID” – are NOT allowed to overlap (see DirectionID below).

VehicleID	optional	Vehicle identifier
CompleteTrip	optional	Flag indicating whether this real-trip contains the complete sequence of stops as currently known to the delivering system. Typically “true” for additional or replacement trips. Complete trips should be sent once to receiving systems which expect the complete schedule information.
LineText	optional	Line text (e.g. “Bus 123”), sent when different from published schedule and to systems which expect the complete schedule information.
ProductID	optional	Product id (e.g. 0 for high speed trains) is sent when different from published schedule and to systems which expect the complete schedule information.
DirectionText	optional	Text describing the direction of the trip, sent when different from the published schedule and to systems which expect the complete schedule information. Hint: If more than one DirectionText is specified, the validity ranges – specified by attributes “fromStopID” and “toStopID” – are NOT allowed to overlap (see DirectionText below).
AdministrationID	optional	Identifier of the administration of the trip.
InfoText	optional, multiple	General information on the trip, sent when different from the published schedule and to systems which expect the complete schedule information.
DelayReason	optional, multiple	Specific reason for delays on this trip.
TrainName	optional	New train name, e.g. “West Coast Express”. No specification means that there is no change with regard to planned schedule.
TransportModeText	optional	Name of the mode of transport (e.g. “Tram”), sent when different from the published schedule and to systems which expect the complete schedule information.
PredictionPossible	optional	Indicates if the delivering system is currently capable to deliver estimated times (=predictions for stations not yet reached) for this trip.

	PredictionInaccurate	optional	Quality status of the prediction. (“QueueFlag”; vehicle stopped in traffic jam, delay will most probably increase further). If this field is set, the receiving system assumes that delay times are uncertain. Please note that the values of this field are predefined in the XSD.
	ExtraTrip	optional	Indicates that this trip is an addition to the planned timetable (value=”true”).
	Deleted	optional	If “true”, this indicates that the whole trip has been cancelled (the list of RealStops can also be empty in this case).
	ResetRTData	optional	If this flag is set to “true”, the receiving system will discard all accumulated real-time changes for the RealTrip.
	SectionDeactivation	optional, multiple	Information on a deactivated trip section
	SectionActivation	optional, multiple	Information on an activated trip section (normally used to revert a section deactivation)
	CyclesPermitted	optional	“true”, if it is possible to take cycles on this trip.
	VehicleTypeID	optional	Transport vehicle type; this information will be useful for disabled people.
	PassengerLoad	optional	Possible values: “Light load”, “Heavy load”, “Overloaded”. Information can be useful for fetcher 16onfigution in a feeder/fetcher scenario.
	TripRef	mandatory	This uniquely identifies the “RealTrip” (based on TripID and optionally StartStop/EndStop of the Trip).
	NextTrip	optional, multiple	Reference to identifier of follow-up-trip (important for chained trips or “ring lines”).
	RealStop	optional, multiple	Elements with details on the stops in the route.
	RealPosition	optional, multiple	Element with geographical position of vehicle (not necessarily coinciding with regular stops); can be useful if external system does not keep track of all possible stops.
	GeoPosition	optional, multiple	Element with actual geo position of a vehicle, but without relation to stops or POIs.

	TripProgress	optional, multiple	Element with information about the trips progress in relation to its stops or to realgraph nodes.
	Occupancy	optional, multiple	Element with occupancy information for the trip or a section of the trip.

NextTrip ***(FolgeFahrt)***			XML sub elements
	TripRef	mandatory	see “TripRef” definition above

RealStop ***(IstHalt)***			XML sub elements
	Deleted	optional	If “true”, this indicates that this station has been removed from the original sequence of stations (which are part of the trip).
	StopID	mandatory	Station identifier. Hint: can be mapped by a station mapping table. Hint 2: In systems working with a stop area/stop point differentiation you have to supply in this field the stop point ID of the actual (=”current”) stop according to the given realtime situation.
	DepartureTime	optional	Planned departure time, omitted at the end stop. At least one of the following fields should be specified: “DepartureTime”, “ArrivalTime”, “RealDepartureTime” or “RealArrivalTime”.
	ArrivalTime	optional	Planned arrival time, omitted at the start stop. At least one of the following fields should be specified: “DepartureTime”, “ArrivalTime”, “RealDeparturePrediction” or “RealArrivalPrediction”.
	RealDeparturePrediction	optional, deprecated	Prognosis for departure time. Use RealDepartureTime with status=”Prognosis” instead.
	RealArrivalPrediction	optional, deprecated	Prognosis for arrival time. Use RealArrivalTime with status=”Prognosis” instead.

	RealDeparture	optional	Real departure time (in sub element Time), reported or predicted according to status.
	RealArrival	optional	Real departure time (in sub element Time), reported or predicted according to status.
	PredictionInaccurate	optional	Quality status of the prediction. (“QueueFlag”; vehicle stopped in traffic jam, delay will most probably increase further). If this field is set, the receiving system can assume that delay times are uncertain. Please note that the values of this field are predefined in the XSD.
	ScheduledDeparture	optional	Scheduled Departure bay/platform name – or station point/pole identifier (depending on 18onfiguretion) Hint: This field is only relevant for systems working with a stop area/stop point differentiation. If this is the case, this field must be filled in all applicable cases.
	ScheduledArrival	optional	Scheduled Arrival bay/platform name – or station point/pole identifier (depending on configuration) Hint: This field is only relevant for systems working with a stop area/stop point differentiation. If this is the case, this field must be filled in all applicable cases.
	DeparturePlatformText	optional	Departure bay or platform name (to be specified only if there are changes compared to schedule).
	ArrivalPlatformText	optional	Arrival bay or platform name (to be specified only if there are changes compared to schedule).
	NoBoarding	optional	Vehicle only stops for alighting. No specification means there is no change with regard to planned schedule.
	NoAlighting	optional	Vehicle only stops for boarding. No specification means there is no change with regard to planned schedule.
	PassThru	optional	Vehicle does not stop here, travels straight through. No specification means there is no change with regard to planned schedule.
	PassengerLoad	optional	see explanations on similar field for “RealTrip”; scope here is restricted to a station.

	ExtraStop	optional	This stop is additional and unplanned (if “true”). No specification means there is no change with regard to planned schedule.
	Coordinates	optional	Geo coordinates of the stop.
	Name	optional, multiple	Name or names of the stop, optionally localized.
	ShortName	optional, multiple	Short names of the stop, optionally localized.
	AreaID	optional	
	AreaName	optional	
	AreaCode	optional	

RealDeparture			XML sub elements
	Time	mandatory	Real departure time.
	Status	mandatory	Status of the departure time, either “Prognosis” or “Reported”.

RealArrival			XML sub elements
	Time	mandatory	Real arrival time.
	Status	mandatory	Status of the arrival time, either “Prognosis” or “Reported”.

RealPosition			XML sub elements
	Xcoordinate	optional	Longitude; x-coordinate (WGS84-system) of current position in decimal degree notation (6 digits after dot are significant). One of the following has to be specified: pair of Xcoordinate/Ycoordinate, LastStop or at least one of DepartureTime or ArrivalTime.

	Ycoordinate	optional	Lattitude; y-coordinate (WGS84-system) of current position in decimal degree notation (6 digits after dot are significant). One of the following has to be specified: pair of Xcoordinate/Ycoordinate, LastStop or at least one of DepartureTime or ArrivalTime.
	Zcoordinate	optional	Altitude; z-coordinate of current position in meters above sea level
	DepartureTime	optional	Scheduled departure time (if available) One of the following has to be specified: pair of Xcoordinate/Ycoordinate, LastStop or at least one of DepartureTime or ArrivalTime.
	ArrivalTime	optional	Scheduled arrival time (if available) One of the following has to be specified: pair of Xcoordinate/Ycoordinate, LastStop or at least one of DepartureTime or ArrivalTime
	RealDeparturePrediction	optional	Actual departure time at current position One of RealDeparturePrediction or RealArrivalPrediction has to be specified.
	RealArrivalPrediction	optional	Actual arrival time at current position One of RealDeparturePrediction or RealArrivalPrediction has to be specified.
	LastStop	optional	Identifier of the last station passed through Hint: can be mapped by a station mapping table. One of the following has to be specified: pair of Xcoordinate/Ycoordinate, LastStop or at least one of DepartureTime or ArrivalTime.
	DistanceToLastStop	optional	Distance to last station passed through in meter
	CirculatesOnDetour	optional	Flag, if “true” the vehicle circulates on a detour route.
	NearbyPOI	optional	Zero or more points of interest (e.g. traffic lights) nearby the current position. Given as sub elements of type POIType.

NearbyPOI		XML sub elements

	ID	mandatory	ID of the POI.
	Name	optional	Name of the POI.
	Type	mandatory	Type of the POI. Currently defined types are: 0 for traffic lights 1 for fuel stations
	Xcoordinate	optional	POI longitude in WGS 84 decimal degrees.
	Ycoordinate	optional	POI latitude in WGS 84 decimal degrees.
	Zcoordinate	optional	POI elevation in meters.
	Attribute	optional, multiple	POI attributes, to be defined per project.

SectionDeactivation			XML sub elements
	TripSection	mandatory	Section of trip affected by cancellation/deactivation.

SectionActivation			XML sub elements
	TripSection	mandatory	Section of trip affected by activation.

TripSection			XML sub elements
	FromStopID	optional	Stop identification number (can be remapped by conversion table) of the first stop in the affected section; FromStopID can be omitted if it denotes the first stop of the trip – but then ToStopID has to be specified.
	DepTime	optional	Departure time at stop identified by FromStopID; field will be ignored if FromStopID is not specified.
	ToStopID	optional	Stop identification number (can be remapped by conversion table) of the last stop in the affected section; ToStopID can be omitted if it denotes the last stop of the trip – but then FromStopID has to be specified.
	ArrTime	optional	Arrival time at stop identified by ToStopID; field will be ignored if ToStopID is not specified.

DirectionID		The direction identifier value Optionally qualified by the following XML attributes:
fromStopID	optional	Stop identification number (can be remapped by conversion table) of the first stop to which the direction ID applies.
toStopID	optional	Stop identification number (can be remapped by conversion table) of the last stop to which the direction ID applies.
fromStopDepTime	optional	Departure time at stop identified by fromStopID; field will be ignored if fromStopID is not specified.
toStopArrTime	optional	Arrival time at stop identified by toStopID; field will be ignored if toStopID is not specified.
hafasDirFlag	optional	Single character – which represents in HAFAS a group of directions (e.g. “0” for outbound, “1” for inbound)

DirectionText		The direction text value Optionally qualified by the following XML attributes:
fromStopID	optional	Stop identification number (can be remapped by conversion table) of the first stop to which the direction ID applies.
toStopID	optional	Stop identification number (can be remapped by conversion table) of the last stop to which the direction ID applies.
fromStopDepTime	optional	Departure time at stop identified by fromStopID; field will be ignored if fromStopID is not specified.
toStopArrTime	optional	Arrival time at stop identified by toStopID; field will be ignored if toStopID is not specified.

DelayReason		XML attributes
ID	optional	Identifier for the delay reason, must be valid for this specific trip only.
Delete	optional	When true the delay reason is removed from this trip. When true, this also requires that the ID is given.

DelayReason			XML sub elements
	Text	optional, multiple	Localized texts describing the delay reason. The language codes used in the localized texts have to be agreed on per project. It is advisable to use ISO-639 codes for specifying languages.
	Code	optional	Optional code for a delay reason. Delay reason codes have to be specified on a per project base.
	Scope	optional, multiple	Optional specification for where along the trip the delay reason or delay reasons are valid.

GeoPosition			
	tbd		

TripProgress			
	tbd		

Occupancy			
	prognosis	mandatory attribute	Defines if the given occupancy information is a prognosis or an actual information.
	fromStop	optional	Optional stop from which the occupancy information is valid.
	toStop	optional	Optional stop to which the occupancy information is valid.
	Capacity	mandatory	The total capacity of the trip the information is given for. The capacity might be - an actual capacity of the vehicle conducting the trip. - a base value for a relative capacity. For example using 100 for Capacity allows to give a percentage value in the Level. For example using 3 in Capacity directly allows to give a Level value directly usable to steer graphical display with indicators for “empty”, “light”, “heavy”, “full”.

			The values used for Capacity and Level should be agreed on per project or per product.
	Level	mandatory	The occupancy in relation to Capacity.

5 Data from Vehicles

Several types of information acquired from vehicles such as occupancy data and telemetry data can be sent to HRX peers.

5.1 Occupancy Data

Occupancy data acquired by counting devices in the vehicle can either contain relative changes of passenger count or an absolute count. In addition a trip reference, a stop reference and a geo position can be given in order to reference the acquired data more exactly.

For each pair of HRX peers the actual amount of data sent has to be defined on a per project or product base. Especially the question if it is acceptable that one peer only sends relative counts has to be agreed on, since sending relative counts only has a risk of cumulating errors.

OccupancyData		XML sub elements
Timestamp	mandatory	Generation time of this message.
VehicleID	mandatory	Identifier of the vehicle on which the data was acquired, must be known to both peers.
MaxPassengers	optional	Maximum number of passengers allowed in the vehicle, when known to the sender.
PassengerCount	optional	The actual number of passengers on the vehicle.
Boarding	optional	Total number of boarding passengers.
Alighting	optional	Total number of alighting passengers.
DoorCount	optional, multiple	
TripRef	optional	
StopRef	optional	

	GeoPosition	optional	

DoorCount			XML sub elements
	DoorId	mandatory	Identifier of the door the count was done at, necessary when the vehicle has more than one door.
	Boarding	mandatory	Number of passengers that boarded the vehicle.
	Alighting	mandatory	Number of passengers that alighted from the vehicle.
	Location	optional	Location name of the door or sensor.

5.2 Door Events

DoorEvent			XML sub elements
	Timestamp	mandatory	Generation time of this message.
	VehicleID	mandatory	Identifier of the vehicle on which the data was acquired, must be known to both peers.
	Type	mandatory	Type of door event, one of: - Open - Close - Lock - Unlock
	PassengerCount	optional	The actual number of passengers on the vehicle.
	TripRef	optional	
	StopRef	optional	
	GeoPosition	optional	

5.3 Telemetry Data

Telemetry data acquired on a vehicle can be sent to a HRX peer as a sequence of key-value pairs annotated with a timestamp.

TelemetryData		XML sub elements
VehicleID	mandatory	Identifier of the vehicle on which the data was acquired, must be known to both peers.
TripRef	optional	Optional identifier of the vehicle's current trip.
TelemetryValue	mandatory, multiple	A single recorded telemetry value.

TelemetryValue		XML sub elements
Timestamp	mandatory	Time at which the value was acquired.
Key	mandatory	Key for the telemetry value. Several keys are pre-defined, additional keys may be defined on a per product or project base. The predefined keys are: ML: mileage as read from odometer in meters FL: level of remaining fuel in percent TS: tachograph state, allowed values are: - Unknown - Inactive - Working - Driving - Resting
Value	mandatory	Actual value.
GeoPosition	optional	Geo position where the value was acquired.

6 Real-time Information on Connections

To be defined.

7 Real-time Information on Occupancy and Train Formation

To be defined.

This describes the data / messages sent by a HRX peer that manages occupancy and train formation details.

8 Version Information

Version	Date	Author	Changes
2.4.1	2016-11-30	Marc Föll	Rewrite of HRX documentation. Additional information types OccupancyData and TelemetryData. Multiple changes to RealTrip element, dropping unused properties, but backwards compatible.
2.4.2	2017-01-27	Marc Föll	New attribute RealtimeInfo.resetRTData.
2.4.3	2017-03-22	Marc Föll	Clarification of RealStop.RealDeparture and RealStop.RealArrival types. Small changes on WSDL for easier code generation. Additional element type for sending door open and close events for vehicles. Additional element type for sending vehicle / trip position with nearby POI. Clarifications and additional properties for occupancy and telemetry data. New attributes RealtimeInfo.fullRTDeliveryStart and RealtimeInfo.fullRTDeliveryEnd.
2.4.4	2017-10-26	Marc Föll	Clarifications of patterns for UniqueIDType and VehicleIDType. New optional elements Name and Coordinates in element RealStop. GeoPosition element as extension of GeoCoordinates element.
2.4.5	2017-12-18	Marc Föll	Allow to use sub element GeoPosition in RealTrip. New sub element DelayReason in RealTrip.
2.4.5	2018-01-09	Marc Föll	Additional example for RealTrip with GeoPosition.
2.4.6	2018-03-05	Marc Föll	XSD change only, refined schema for DoorCount
2.4.7.	2018-07-02	Marc Föll	New optional sub element RealTrip.TripProgress. New XML elements for connection protection.

2.4.8	2019-02-08	Marc Föll	TripProgess now contains full stop information. Refined description of tenant-/client-distinction. RealTrip.Trip.OperatingDay is an optional attribute now. Simplified VehicleIDType and UniqueIDType in XSD. New sub elements AreaID, AreaName and AreaCode in RealStop. New sub element Source in GeoPosition. New sub element Location in DoorCount. New door event types Lock and Unlock.