



# GS1 EPCIS for Rail Vehicle Visibility Application Standard

GS1 Transport & Logistics standard that explains how to  
implement EPCIS rail vehicle visibility

*Release 1.0.1, Ratified, July 2015*

---

## Document Summary

Document Item	Current Value
Document Name	GS1 EPCIS for Rail Vehicle Visibility Application Standard
Document Date	July 2015
Document Version	1.0
Document Issue	1
Document Status	Ratified
Document Description	GS1 Transport & Logistics standard that explains how to implement EPCIS rail vehicle visibility

## Contributors

Name	Organisation
Lennart Andersson	Trafikverket
Niclas Andreasson	Havs-och vattenmyndigheten
Dave Burbridge	Network Rail
Jonas Buskenfried	GS1 Sweden
Christian Daniel	SNCF Rolling Stock Department
Daniel Dünnebacke	GS1 Germany
Peter Feldmann	HARTING IT System Integration GmbH & Co.
Marc Flederud	FermRFID
Alan Gormley	GS1 Ireland
Hein Gorter de Vries	GS1 Netherlands
Heinz Graf	GS1 Switzerland
Pertti Hakala	GS1 Finland
Mikael Hansson	Trafikverket
Gunnar Ivansson	Learningwell AB
Theo Kruse	ProRail
Philip Leslie	Coriel Ltd
Roar Lorvik	GS1 Norway
Terje Menkerud	GS1 Norway
Alice Mukaru	GS1 Sweden
Mark Read	Coriel Ltd
Hubert Rheinheimer	Schweizerische Bundesbahnen SBB
Egil Røyseth	Jernbaneverket
Frank Siebenmorgen	SupplyOn AG
Holger Strietholt	Schweizerische Bundesbahnen SBB
Pär Söderström	SJ AB
Vesa-Pekka Tapper	Vilant Systems Oy
Petter Thune-Larsen	GS1 Norway



Name	Organisation
Geir Vevle	HRAFN AS
David Weatherby	GS1 UK
Karl Åkerlund	Trafikverket

## Log of Changes

Release	Date of Change	Changed By	Summary of Change
1.0	Apr 2015	Michael Sarachman Craig Alan Repec Coen Janssen	Initial publication
1.0.1	July 2015	David Buckley	Errata – fix the list numbering to start at 1

## Disclaimer

GS1®, under its IP Policy, seeks to avoid uncertainty regarding intellectual property claims by requiring the participants in the Work Group that developed this **GS1 EPCIS for Rail Vehicle Visibility Application Standard** to agree to grant to GS1 members a royalty-free licence or a RAND licence to Necessary Claims, as that term is defined in the GS1 IP Policy. Furthermore, attention is drawn to the possibility that an implementation of one or more features of this Specification may be the subject of a patent or other intellectual property right that does not involve a Necessary Claim. Any such patent or other intellectual property right is not subject to the licencing obligations of GS1. Moreover, the agreement to grant licences provided under the GS1 IP Policy does not include IP rights and any claims of third parties who were not participants in the Work Group.

Accordingly, GS1 recommends that any organization developing an implementation designed to be in conformance with this Specification should determine whether there are any patents that may encompass a specific implementation that the organisation is developing in compliance with the Specification and whether a licence under a patent or other intellectual property right is needed. Such a determination of a need for licencing should be made in view of the details of the specific system designed by the organisation in consultation with their own patent counsel.

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF THIS SPECIFICATION. GS1 disclaims all liability for any damages arising from use or misuse of this Standard, whether special, indirect, consequential, or compensatory damages, and including liability for infringement of any intellectual property rights, relating to use of information in or reliance upon this document.

GS1 retains the right to make changes to this document at any time, without notice. GS1 makes no warranty for the use of this document and assumes no responsibility for any errors which may appear in the document, nor does it make a commitment to update the information contained herein.

GS1 and the GS1 logo are registered trademarks of GS1 AISBL.



# Table of Contents

- 1 Introduction ..... 6**
- 2 References..... 6**
- 3 Terms and definitions ..... 7**
  - 3.1 Rail visibility terms ..... 7
    - 3.1.1 Equipment ..... 7
    - 3.1.2 Infrastructure..... 7
  - 3.2 GS1 Terms ..... 9
    - 3.2.1 GS1 Keys..... 9
- 4 Rail vehicle visibility ..... 10**
  - 4.1 Business need ..... 10
  - 4.2 Asset tracking applications ..... 10
  - 4.3 Wayside Train Monitoring System (WTMS) applications ..... 10
  - 4.4 RFID enabled Automatic Vehicle Identification (AVI) systems ..... 11
  - 4.5 Train Management System (TMS) ..... 11
- 5 Vehicle identification ..... 12**
  - 5.1 Vehicle identification with “master” GIAI ..... 12
  - 5.2 AIDC device identification with “proxy” GIAI ..... 12
- 6 Read Point and Business Location Identification ..... 14**
  - 6.1 Read Points ..... 14
    - 6.1.1 Identifying a track using EPC/RFID..... 14
  - 6.2 Business Location..... 15
- 7 Determining vehicle and train visibility data..... 17**
  - 7.1 Determining the train direction ..... 17
  - 7.2 Determining the orientation of the rail vehicle..... 17
  - 7.3 Determining Source and Destination..... 19
  - 7.4 Determining a train passage..... 19
- 8 Sharing vehicle and train visibility data with EPCIS..... 20**
  - 8.1 Vehicle and train visibility – critical tracking events ..... 20
  - 8.2 EPCIS event data ..... 21
    - 8.2.1 What ..... 21
    - 8.2.2 When ..... 21
    - 8.2.3 Where ..... 22
    - 8.2.4 Why ..... 22
    - 8.2.5 Extension elements..... 24
    - 8.2.6 Transaction event elements ..... 25
- 9 Examples of Rail Visibility Events ..... 26**
  - 9.1 Rail vehicle observations..... 26
  - 9.2 Rail vehicle changing direction ..... 29
  - 9.3 Train passage event ..... 33



- 10 EPCIS Query examples for rail vehicle visibility ..... 35**
  - 10.1 On-demand queries via EPCIS Query Control Interface ..... 35
    - 10.1.1 Observations of a specified vehicle since a specified date/time ..... 35
    - 10.1.2 Observations of all vehicles at a specified read point in a specified window of time..... 35
    - 10.1.3 Events for a given passage ..... 35
    - 10.1.4 Passage-level queries ..... 36
  - 10.2 Standing queries (Subscriptions) via EPCIS Query Call-back Interface ..... 36
    - 10.2.1 Notification whenever a specified vehicle is observed at any read point. .... 36
    - 10.2.2 Notification whenever any uniquely identified vehicle is observed at a specified read point. .... 36
- 11 Outside the scope of EPCIS: Determining rail anomalies and what HAS NOT been seen . 37**
- 12 Events and Queries – Examples (non-normative)..... 38**
  - 12.1 Example 1 – Rail vehicle observations..... 38
  - 12.2 Example 2 – Rail vehicle changing direction..... 40
  - 12.3 Example 3 – Train passage Transaction event..... 43
  - 12.4 Example 4 – Query for observations of a specified vehicle since a specified date/time..... 45
  - 12.5 Example 5 – Query for observations of all vehicles at a specified read point in a specified window of time 46
  - 12.6 Example 6 – Query for all events for a given passage ..... 47
  - 12.7 Example 7 – Passage-level queries ..... 48
  - 12.8 Example 8 – Standing query (subscription) for notification whenever a specified vehicle is observed at any read point ..... 49
  - 12.9 Example 9 – Standing query (subscription) for notification whenever any uniquely identified vehicle is observed at a specified read point. .... 50
- 13 Abbreviations ..... 51**

## 1 Introduction

This document explains how to implement EPCIS rail vehicle visibility. Contributors have expressed their commitment to implement the approaches described in this standard to ensure interoperability of shared data among rail stakeholders and other related sectors.

This standard will periodically be updated, reflecting the learnings of initial implementations. Please contact the GS1 Transport & Logistics team ([GS1tlc@gs1.org](mailto:GS1tlc@gs1.org)) for more information about projects in rail and developments on this topic.

## 2 References

Document	Author / Year
GS1 General Specifications, version 14	GS1, 2014
GS1 Tag Data Standard (TDS), version 1.9	GS1, 2014
EPCIS Standard, version 1.1	GS1, 2014
Core Business Vocabulary (CBV), version 1.1	GS1, 2014
GS1 in Europe Rail Identification Guideline	GS1 in Europe, 2012
GS1 Application Level Events (ALE) Standard, version 1.1.1	GS1, 13 March 2009

## 3 Terms and definitions

### 3.1 Rail visibility terms

#### 3.1.1 Equipment

##### **Train**

The railway terminology that is used to describe a 'train' varies between countries.

One or more railway vehicles hauled by one or more locomotives or railcars, or one railcar alone, running under a given number or specific designation from an initial fixed point to a terminal fixed point.

[\[OECD Glossary of Statistical Terms\]](#)

In the United Kingdom Section 83(1) of the Railways Act 1993 defines "train" as follows:

- a. two or more items of rolling stock coupled together, at least one of which is a locomotive; or
- b. a locomotive not coupled to any other rolling stock.

##### **Railway vehicle**

Mobile equipment running exclusively on rails, moving either under its own power (locomotives and railcars) or hauled by another vehicle (coaches, railcar trailers, vans and wagons).

[\[OECD glossary of statistical terms\]](#)

Synonyms: Rolling stock, Rail vehicle

##### **Wagon**

Railway vehicle normally intended for the transport of goods.

[\[OECD glossary of statistical terms\]](#)

##### **Locomotive**

Railway vehicle equipped with prime mover and motor or with motor only used for hauling railway vehicles.

[\[OECD glossary of statistical terms\]](#)

Synonyms: Traction, Equipment

##### **Wheelset / axle / bogie / truck**

A wheelset is the wheel-axle assembly of a railway vehicle. The frame assembly beneath each end of a railway vehicle that holds the wheelsets is called the bogie (or truck in North America).

##### **Train passage**

In this standard a train passage is defined as a collection of one or more rail vehicles passing a read point at the same time, and therefore assumed to be connected to each other.

#### 3.1.2 Infrastructure

##### **Track**

A pair of rails over which railway vehicles can run.

[\[OECD glossary of statistical terms\]](#)

The track on a railway or railroad, also known as the permanent way, is the structure consisting of the rails, fasteners, railroad ties (sleepers, British English) and ballast (or slab track), plus the

underlying subgrade. For clarity, it is also referred to as railway or running track (British English and UIC terminology) or railroad track (predominantly in the United States).

[[OECD Glossary of Statistical Terms](#)]

### **Railroad direction**

Most railroads use nominal cardinal directions for the directions of their lines which often differ from actual compass directions. These directions are often referred to as "railroad" north, south, east or west to remove ambiguity with the same compass directions.

Typically an entire railroad system (the lines of a railroad or a related group of railroads) will describe each of its lines by only two directions, either east and west or north and south. This greatly reduces the possibility of misunderstanding the direction in which a train is travelling as it traverses lines they sometimes twist and turn or even reverse direction for a distance. These directions also have meaning in conflicts between trains running in opposite directions.

### **Axle counter**

An axle counter, also known as a track vacancy detection system, is a device (*usually using wheel sensors*) on a railway that detects the passing of a train between two points on a track. A counting head (or 'detection point') is installed at each end of the section, and as each train axle passes the counting head at the start of the section, a counter increments. A detection point comprises two independent sensors, therefore the device can detect the direction and speed of a train by the order and time in which the sensors are passed. As the train passes a similar counting head at the end of the section, the counter decrements. If the net count is evaluated as zero, the section is presumed to be clear for a second train.

### **Wheel Sensor**

A wheel sensor is a device attached to a rail to detect the presence of a wheel. Wheel diameter can often also be determined from the output, depending on the installation and type. The device usually relies on one or two inductive coils. When using a device with two coils, or two single coil devices, other information such as speed and direction of travel can also be determined from the output.

### **Railway station**

A train station, a stopping point for trains, usually with passenger access.

### **Depot**

Facility where [locomotives](#) and other railway vehicles are housed when not being used, repaired, or maintained.

### **Freight terminal**

A facility where freight is loaded and off-loaded, typically located at ports, including airports, seaports, railroad terminals, and trucking terminals.

### **Classification, Marshalling or Shunting Yards**

Classification (American and Canadian English), marshalling (British, Indian English and Canadian English) or shunting yards are railroad yards where individual railway vehicles are assembled into trains.

### **Wayside Train Monitoring System (WTMS)**

Wayside Train Monitoring Systems (WTMS) are integrated systems of measurement devices that monitor the condition of rolling stock travelling on the railway tracks.



## 3.2 GS1 Terms

GS1 manages a global system that allows companies all around the world to globally and uniquely identify objects such as trade items (products & services), assets, logistic units, shipments, physical locations and parties.

### 3.2.1 GS1 Keys

The following GS1 keys and attributes are applied in the EPCIS Rail implementation.

#### 3.2.1.1 Global Location Number (GLN)

The GS1 Identification Key used to identify physical locations or parties. The key comprises a GS1 Company Prefix, Location Reference, and Check Digit.

##### GLN Extension Component

The GLN extension component may be used to identify internal physical locations within a location that is identified with a GLN, such as specific areas within a building or on a site.

##### SGLN

The SGLN EPC scheme is used to populate an EPCIS event with a GLN (either with or without the optional extension component), which uniquely identifies a physical location

The SGLN EPC scheme's general syntax is:

```
urn:epc:id:sgln:CompanyPrefix.LocationReference.Extension
```

If a GLN Extension Component is applied, it must be included in the Extension field of the SGLN. If no GLN Extension Component is applied, the Extension field in the SGLN must be filled with a single zero digit, indicating that the SGLN represents a GLN, without a GLN Extension Component.

- Example of SGLN without extension component: urn:epc:id:sgln:4012345.89012.0
- Example of SGLN with extension component: urn:epc:id:sgln:4012345.89012.254

#### 3.2.1.2 Global Individual Asset Identifier (GIAI)

The GS1 Identification Key used to identify an Individual Asset. The key comprises a GS1 Company Prefix and Individual Asset Reference.

- Example of GIAI: urn:epc:id:giai:4012345.98765432198765432

## 4 Rail vehicle visibility

### 4.1 Business need

The needs for information sharing are mainly found to be two-fold: (i) there is a need for tracking of vehicles as they travel within countries and across different countries (“asset tracking”), and (ii) there is a need to associate the vehicle data with the Wayside Train Monitoring System (WTMS) data about vehicles and vehicle components to enhance preventive maintenance.

It is assumed that more applications for vehicle and train visibility information will emerge as the use of RFID for railway vehicles becomes more and more popular. However, these two applications are the basis for the initial EPCIS application in rail.

UHF Gen2 RFID has been implemented for railway vehicle tracking in some countries in Europe, including the following examples.

- Shunting yard management system using UHF Gen2 handhelds since 2010 used by VR Transpoint, the freight operator in Finland,
- Fixed RFID reader solution by the Infrastructure manager, the Finnish Transport Agency, as all the railway vehicles in Finland have been tagged since 2010.
- In addition, similar system is in building phase in Sweden, and there are several systems in piloting or testing phase, e.g., in UK, Switzerland and other European countries.

### 4.2 Asset tracking applications

The need for vehicle *asset tracking* relates to normal operative functions where the location of specific vehicles needs to be known. Asset tracking allows operations to be better planned and can improve vehicle utilisation.

Asset tracking is typically managed at the vehicle level, tracking the location and status of each vehicle as they travel. This is managed independent of the train.

Potential future applications:

- real-time cargo tracking
- estimating the vehicle distance travelled for planning preventive maintenance ,
- Planning vehicle availability

### 4.3 Wayside Train Monitoring System (WTMS) applications

Wayside Train Monitoring Systems (WTMS) are an integrated system of measurement devices that monitor the condition of rolling stock travelling on the railway tracks. These systems have already been in use for many years and are becoming more common, as a means to reduce accidents, enable preventive maintenance and improving rail system reliability. The most typical WTMS device types are:

- Hot Axle-Box Detectors (HABD) or hot-box detectors – measure wheel and axle bearing temperatures to identify those that require maintenance or repairs
- Wheel impact load detectors (WILD) – measure static and dynamic forces between rail and wheels to identify wheel defects and improper loading of the vehicles
- Acoustic Axle Bearing Monitoring (AABM): system that measures the degradation of an axle bearing using acoustic sensors.
- Automatic pantograph monitoring systems (APMS) –high-speed cameras and image processors used to recognise broken Pantograph carbon strips.

RFID is sometimes used to enhance WTMS systems by automatically linking measurement results to each specific vehicle. Without RFID, the measurement results typically are not saved, but only in case of critical alarm, the train is stopped, and the vehicles/axle/wheel causing the alarm is manually sought, identified and repaired. RFID allows for automatic collection of all measurement

results and creating statistical data of measurement results per vehicle, which can be used for, e.g., pre-emptive repair of the developing faults in the vehicles.


Applying standard EPC/RFID protocols will facilitate the exchange of data across WTMS systems and between railway operators.

#### 4.4 RFID enabled Automatic Vehicle Identification (AVI) systems

RFID enabled AVI systems usually comprise fixed readers and wheel sensors. The fixed trackside readers identify the vehicles of the passing train, while the train speed can be up to the full line speed at that track location.

The AVI system is capable of identifying all tagged vehicles and their order in the train, and also is able to detect the presence of vehicles with missing or broken tags and their relative location in the train. The latter is important for the WTMS use case, since it enables the measurement results to be linked to the correct vehicles in a train set.

In addition, the AVI system can determine the travel direction, the orientation, axle count, speed, and length of each vehicle.

- 
**Note:** Knowing the orientation of each vehicle is vital since it enables the data to be associated to individual vehicle components, such as wheel sets.

**Figure 4-1** Fixed trackside RFID configuration



Besides vehicle level information RFID enabled AVI systems also enable train level information exchanges, such as observing a train entering or leaving a yard, and the composition / formation of the train.

#### 4.5 Train Management System (TMS)

A Train Management System (TMS) is a system used to control railway operations. It detects and controls movement of trains on a track.

Used in combination with RFID enabled AVI systems the information from the TMS can be used to generate additional event data. For example, the train entering or leaving an area can be deduced by combining data from a TMS and data from previously read points provided by the AVI system.

## 5 Vehicle identification

### 5.1 Vehicle identification with “master” GIAI

Each rail vehicle is identified with a “master” GIAI, which identifies each rail vehicle as an asset. This master GIAI, encoded as a data carrier-neutral, “pure identity” EPC URI SHALL be used to represent rail cars which are included in EPCIS events.

- example: `urn:epc:id:giai:4012345.98765432198765432`

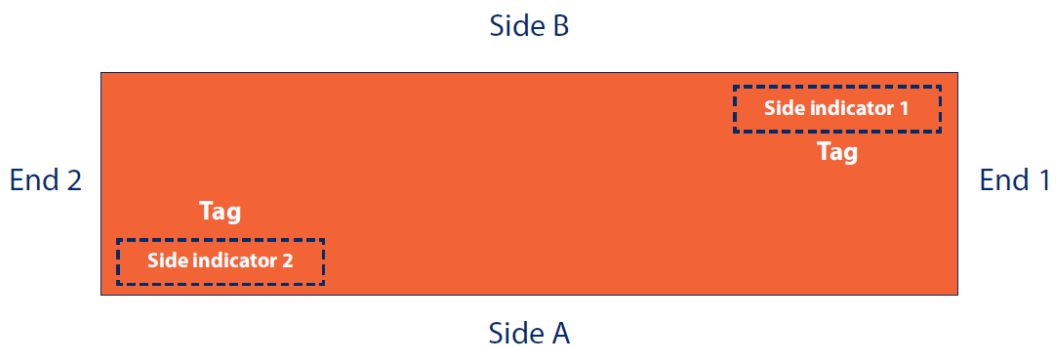
The master GIAI should make it possible to unambiguously determine static information about the rail vehicle, such as the rail vehicle type, the axle count, the vehicle owner, etc. This may be accomplished by communicating this information as master data.

### 5.2 AIDC device identification with “proxy” GIAI

Multiple EPC/RFID tags should be placed on a vehicle in order to enable the automatic identification of the side or end of the wagon. This is often accomplished by placing two tags on opposite sides of the wagon at opposing corners as illustrated in **Figure 5-2** and **Figure 5-3**.

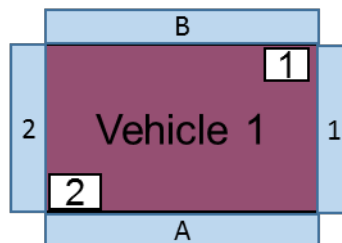
An example is GS1 in Europe’s “RFID in Rail” guideline (*European Guideline for the Identification of Railway Assets using GS1 Standards*) that describes the recommended approach for applying and encoding EPC/RFID tags to European railway vehicles.

**Figure 5-1** Vehicle end/side indicator as per GS1 in Europe’s “RFID in Rail” guideline

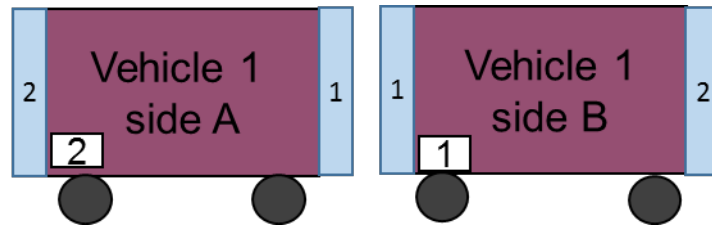


In this global standard the same convention is used when referring to the sides and ends of a rail vehicle. It assumes that vehicle orientation can be inferred from the identity of the tag observed. For further explanation, refer to chapter 5.

**Figure 5-2** Rail vehicle with multiple tags – top view



**Figure 5-3** Rail vehicle with multiple tags – side views



Each of these tags SHALL be identified, by means of a unique GIAI, as fixed assets in their own right. This in turn enables the inference of the vehicle orientation.

- example tag 1 of 2: urn:epc:tag:giai-96:1.4012345.18765432198765432
- example tag 2 of 2: urn:epc:tag:giai-96:1.4012345.28765432198765432

As demonstrated in this example, the tag's filter value should be set to "1" ("Rail Vehicle", per TDS section 10.6).

These "device" GIAIs serve as "proxy" representation in place of the vehicle's "master" GIAI, in case the latter is not physically encoded on an EPC/RFID tag. In this case, these proxy GIAIs should be mapped to the master GIAI in master data. The proxy GIAIs are not replacements for the master GIAI, which SHALL be included in the "WHAT" dimension of EPCIS events pertaining to the rail car in question.

Note that applications receiving EPCIS data shall neither rely on, nor assume the use of, the constructs in the examples above to indicate a proxy/master relationship between GIAIs. External parties consuming data must not be required to make such assumptions, except by explicit bilateral agreement.

## 6 Read Point and Business Location Identification

This section defines how locations can be identified in a rail context. In EPCIS, the Read Point and Business Location are used to provide business context to location types.

**Note:** Read Point and Business Location are distinct from the primitive reader types (Physical Reader ID and Logical Reader ID) as defined in ALE.

### 6.1 Read Points

A Read Point is a location that is meant to identify the most specific place at which an EPCIS event took place.

The read point in an EPCIS event should be identified using the SGLN.

**Example (non-normative):**

```
<readPoint>
  <id>urn:epc:id:sgln:4012345.89012.524</id>
</readPoint>
```

The unique Read Point should make it possible to unambiguously determine the:

- read point’s physical location (e.g., as an geographical address or as geo-coordinates);
- line name / ID and track name / ID

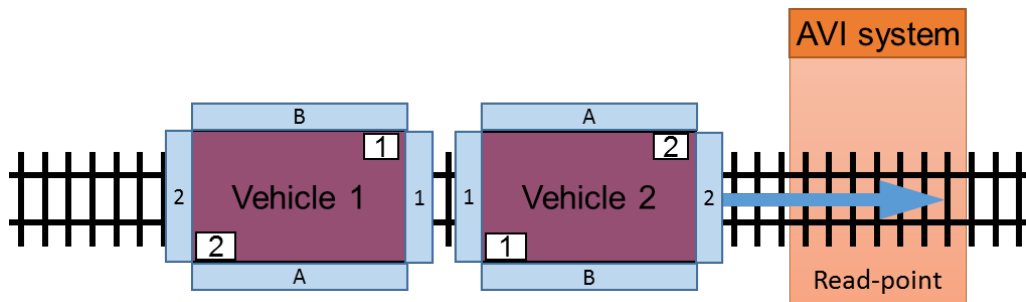
This information should be communicated with relevant stakeholders as master data.

#### 6.1.1 Identifying a track using EPC/RFID

Trackside UHF RFID readers (and their antennas) will be often used to observe the passage of EPC/RFID tagged rail vehicles.

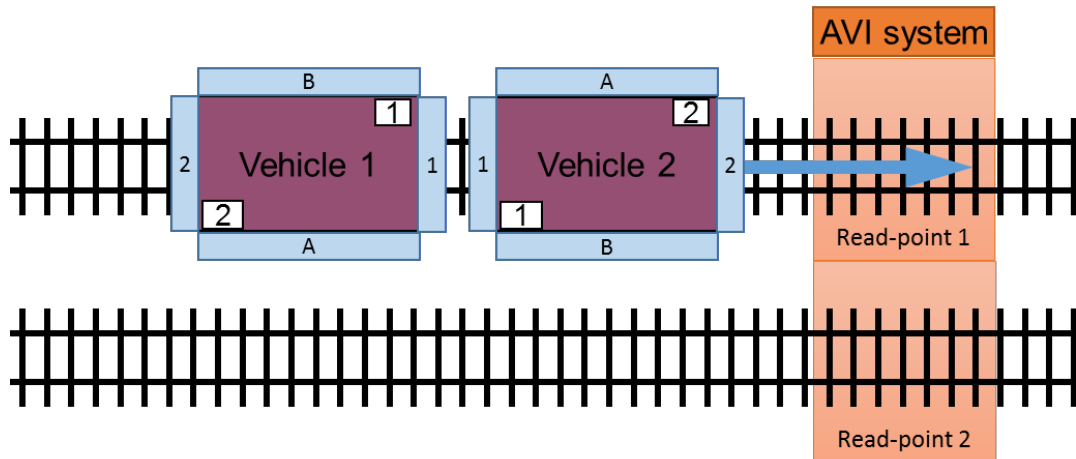
When the reader is configured to monitor a single track there is a one-to-one relation between reader and read point.

**Figure 6-1** AVI system monitoring single tracks



Readers are sometimes configured so that a given interrogation zone intentionally spans multiple tracks (as in the diagram below). In that case it is good practice to apply separate read points with distinct SGLNs for each track, rather than using a single SGLN for the read point which cuts across all sections of track.

**Figure 6-2** Reader monitoring multiple tracks



## 6.2 Business Location

Following its observation at a given read point, the whereabouts of the observed rail vehicle MAY be specified by means of a business location, identified by an EPC URI for locations as per section 8.3.1 of the CBV.

The Business Location is the location where the rail vehicle is assumed to be following the event, until a new event takes place that indicates otherwise. In other words, the Business Location for a given rail vehicle is assumed to be valid until superseded by the business location of a subsequent event pertaining to the rail vehicle.

In railway vehicle asset tracking, the Business Location is used to tell the location where the vehicles or trains are found after the event took place, e.g., track section, station, shunting yard, or specific shunting yard location.

While the Read Point is more directly related to the measurement values received in conjunction with an EPCIS event, the Business Location can be used to serve asset tracking needs, indicating the whereabouts of the asset subsequent to the event, down to the level of granularity/resolution made possible by the operator-determined density of read points

EPC URIs for identification of business locations are specified in section 8.3.1 of the CBV, as well as the TDS. The preferred way to identify a business location is the SGLN.

### Example (non-normative):

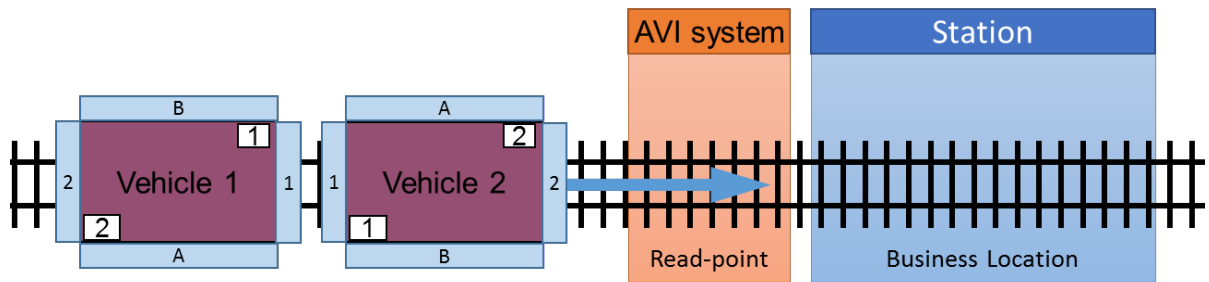
*Following its capture by EPCIS Object event as it enters a station, a rail car is stopped on a given track, identified by business location:*


```
<bizLocation>
  <id>urn:epc:id:sgln:4012345.47119.133</id>
</bizLocation>
```

A given business location can optionally be linked to specific geographical coordinates by means of master data, where required.

**Example**

**Figure 6-3** Business Location is a facility



 **Note:** A business location could also be used to represent a designated area within a rail network.



## 7 Determining vehicle and train visibility data

### 7.1 Determining the train direction

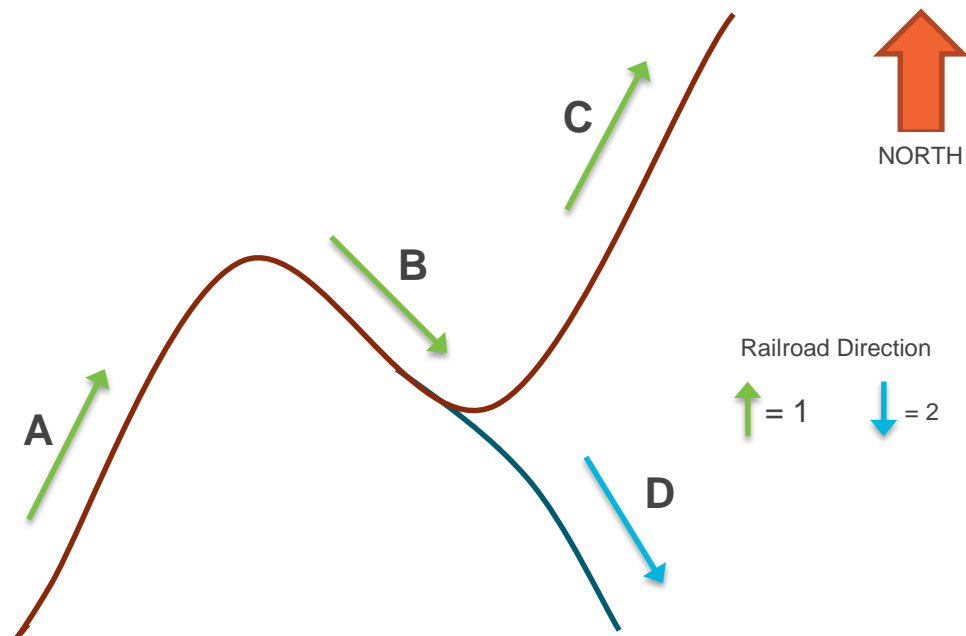
The train direction is assumed to be available from the AVI system.

The train direction should be expressed in terms of the 'railroad direction' of the track, using the **direction indicator** as defined by the infrastructure manager, with the following allowed values:

- 0 = indicates that the direction was not detected,
- 1 = indicates one direction in the rail network,
- 2 = indicates the opposite direction in the rail network.

Besides the direction indicator, the **compass direction** indicating the actual direction of the train at the read point can optionally be provided. The compass direction should be expressed using a cardinal (N, S, W, E) or intermediate cardinal (NW, NE, SE, SW) value.

**Figure 7-1** Determining Train Direction



Position	Railroad Direction	Compass Direction
A	1	NORTH (N)
B	1	SOUTHEAST (SE)
C	1	NORTH (N)
D	2	SOUTH (S)

### 7.2 Determining the orientation of the rail vehicle

Measurements from WTMS equipment need to be assigned to the correct axle on a vehicle. For example, a given vehicle may have four axles, labelled 1, 2, 3 and 4. On its own, the observed tag merely provides information on the (static) orientation of the vehicle, but does not tell the system which end of the vehicle was leading, and therefore it is indeterminate whether the axles passed in

the order 1-2-3-4 or 4-3-2-1. The correct assignment of measurement values therefore also requires the direction of travel of the vehicle.

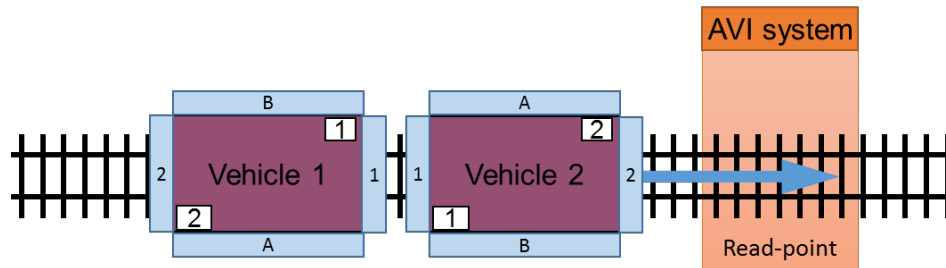
Thus, the orientation of a rail vehicle is determined by combining two pieces of information:

1. the observed tag
2. the train direction

**Vehicle ID and relative sides**

The tag location can be used to provide a first level of information on the ID and relative orientation of a rail vehicle.

**Figure 7-2** Determine train set configuration



Results:

- t1: **Vehicle 2 – tag 2**
- t2: **Vehicle 1 – tag 1**

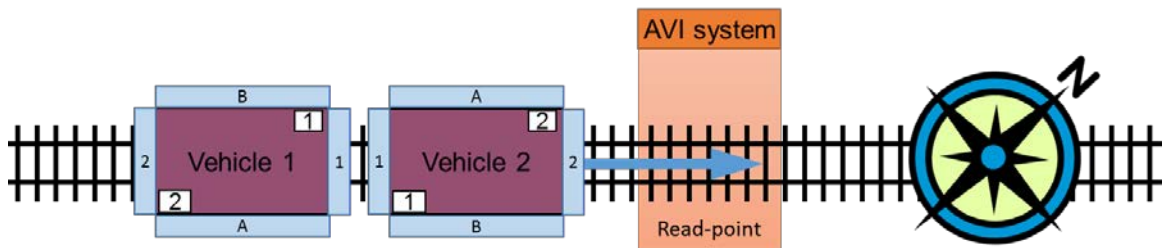
(where t = time)

*Train composition known, observed vehicle tags known*

**Train direction**

In order to specify and share the vehicle orientation the train direction needs to be known as well (see paragraph 7.1).

**Figure 7-3** Determine the train direction



The vehicle orientation can now be derived from a combination of the vehicle's direction and the observed tags.

Results:

- Train direction indicator = **2** (compass direction = **NE**)
- t1: Vehicle 2 – tag 2, **vehicle end 2 passed first**
- t2: Vehicle 1 – tag 1, **vehicle end 1 passed first**

(Where t = time)

*Train composition known, direction known, and orientation known.*



**Note:** If the reader would have been located at the opposite side of the track, for vehicle 2 tag 1 would have been read, and for vehicle 1 tag 2. The orientation would still have been the same:

- t1: Vehicle 2 – **tag 1**, vehicle end 2 passed first
- t2: Vehicle 1 – **tag 2**, vehicle end 1 passed first

### 7.3 Determining Source and Destination

It can be useful for parties to have information on the origin of the train and its ultimate destination. Parties with access to the railroad plan should be able to derive this based on the provided read points and direction information.

To support parties that do not have access to such information it is possible to utilise the Source and Destination elements in EPCIS. Source can be used to house the origin of the train journey, destination can be used to house the final destination of the train journey.

It is recommended to identify source and destination with an SGLN. Use of a UN/LOCODE is also permissible, pending the addition of a standardised UN/LOCODE URI scheme to future updates of GS1's Core Business Vocabulary (CBV).

It should be noted that source and destination information is not necessarily available from the trackside AVI system, and may need additional inputs from, for example, the signalling system. This would need to be combined into the data before it can be shared.

### 7.4 Determining a train passage

The AVI will be able to detect whether observed vehicles are connected in the same train-set.

Upon completion of the passage observation the related event data will be posted. Each vehicle observation will lead to a separate EPCIS event.

In order to specify which vehicles were observed as part of the same passage, a passage identifier should be added to the object events. *The passage identifier should be included as an EPCIS **Business Transaction**.*

The passage ID will also be used in EPCIS transaction events, in order to share aggregate passage level information.

As part of the 'passage' transaction event also the **train number** can be provided. This information can be used to link to information in other train management systems.



**Note:** *The passage identifier should be unique across read points. Two available formats are GDTI (GS1 Global Document Type Identifier) and GUID (Globally Unique Identifier). The GUID is random number which could be generated by a local system, GDTI could have some logic such as sequencing but would then require central administration of allocated IDs.*

Vehicles that are hidden from RFID readers or lacking RFID tags will still be registered by AVI systems.

Use of "dummy" EPCs as proxies for non-identified wagons in EPCIS events should be avoided.

The recommended approach is to include the vehicle level information in the EPCIS **transaction event** for a passage. A **Transaction Event** will be used to list all observed rail vehicles (also the unidentified ones) as well as any relevant totals. The transaction event should reference the passage ID (the same reference that is used in the object events, see above). See chapter 6 for more information.

## 8 Sharing vehicle and train visibility data with EPCIS

It is necessary to note a few special issues when applying EPCIS functions to railway vehicle visibility.

First, it is important to understand the division between EPCIS interface specifications and RFID system specifications. In this EPCIS standard, the EPCIS interface assumes that the RFID system will send the location and the other information in the specified event fields, and does not and should not dictate how that information is created and captured. Still, EPCIS specifications will affect the RFID implementations by telling what information to collect.

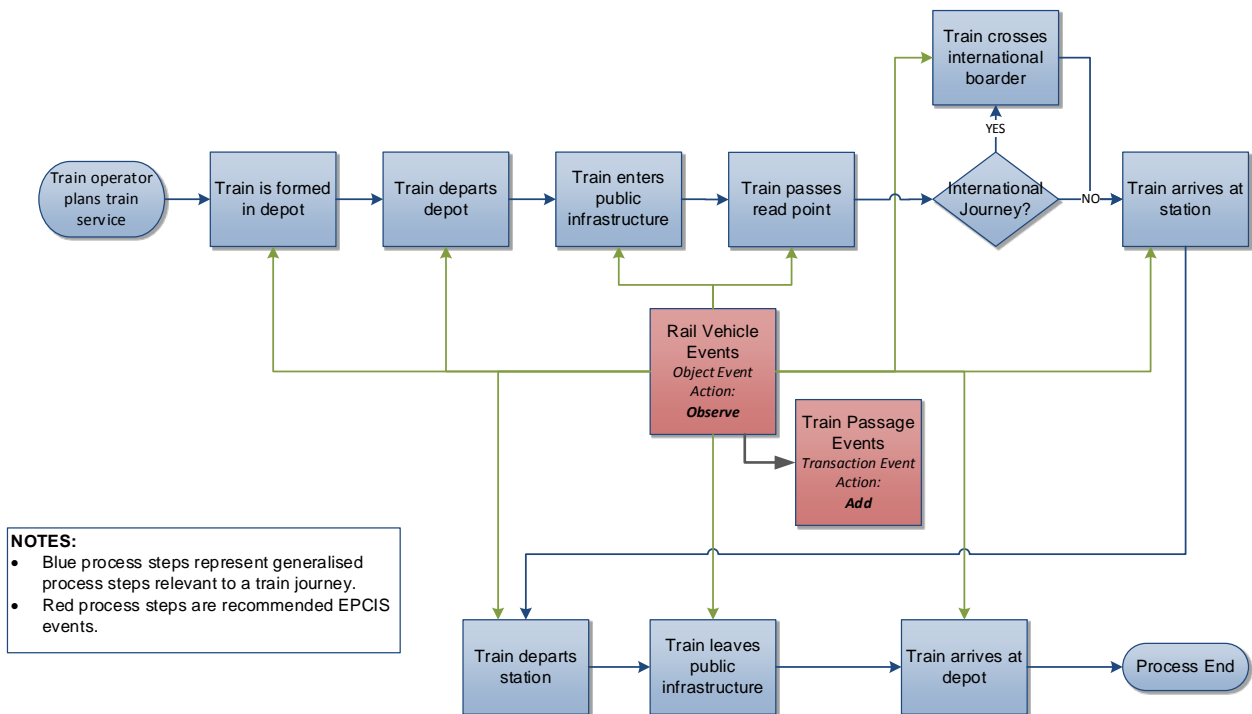
Second, using EPCIS for vehicle visibility requires using EPCIS and CBV standards in a novel way. The EPCIS and CBV standards primarily focus on supply chain operations. However, some asset-tracking-related business steps and dispositions might be required, as new vocabulary elements.

EPCIS also enables information sharing. Consistent EPCIS event structures ensure data can be shared and interpreted by other rail parties.

### 8.1 Vehicle and train visibility – critical tracking events

The diagram below provides a high-level overview of the main train events and associated EPCIS events.

Figure 8-1 Standard Rail Journey Diagram



## 8.2 EPCIS event data

In this rail visibility standard the following EPCIS event types and actions are applied:

- `ObjectEvent` (action `OBSERVE`), which serves as an observation of a uniquely identified rail vehicle in passage along its journey, or upon its arrival at or departure from a terminus;
- `TransactionEvent` (action `ADD`), which serves as a “summary” event following the observation of a passing train’s trailing vehicle, reiterating the proxy GIAIs of positively identified vehicles, as well as relevant totals (e.g., number of wagons ,number of axles) for all – identified and unidentified – vehicles.

### 8.2.1 What

The “WHAT” dimension indicates the objects to which the EPCIS event pertains.

Each observed rail vehicle should be captured in a separate `ObjectEvent`. The `epcList` element should contain only the master GIAI of the observed vehicle.

#### Example:

```
<epcList>
  <epc>urn:epc:id:giai:735005385.9907412345676</epc>
</epcList>
```

A passage SHALL be defined using a `TransactionEvent`. Here in the `epcList` element the master GIAIs of all observed, positively identified rail vehicles SHALL be included.



**Note:** In cases where no EPCs were observed, the value to be provided in the `epcList` element of a `TransactionEvent` will be defined in mutual agreement with the EPCIS data repository owner.

#### Example:

```
<epcList>
  <epc>urn:epc:id:giai:735005385.9907412345676</epc>
  <epc>urn:epc:id:giai:735005385.9907412345607</epc>
</epcList>
```

### 8.2.2 When

The “WHEN” dimension serves as a timestamp for the EPCIS event.

The `eventTime` of `Object` events should reflect the time at which the vehicle was observed.

The `eventTime` of `Transaction` events should reflect the time at which the first (leading) vehicle of a passing trainset was observed.

The `recordTime` reflects the date and time at which this event was recorded by an EPCIS Repository. This field SHALL be ignored when an event is presented to the EPCIS Capture Interface, and SHALL be present when an event is retrieved through the EPCIS Query Interfaces. The `recordTime` does not describe anything about the real-world event, but is rather a bookkeeping mechanism that plays a role in the interpretation of standing queries (subscriptions).

#### Example

```
<eventTime>2014-10-18T12:00:22.000+02:00</eventTime>
<recordTime>2014-10-18T12:01:00.000+02:00</recordTime>
<eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
```

### 8.2.3 Where

The “WHERE” dimension component indicates the location at which the EPCIS event was observed, as well as the whereabouts of the object subsequent to the event.

The `readPoint` – indicating the SGLN corresponding to the event’s location.

The `bizLocation` – indicating the SGLN corresponding to the object’s whereabouts subsequent to the event.

For all object events and transaction events either the `readPoint` or the `bizLocation` or both should be populated.

#### Example

```
<readPoint>
  <id>urn:epc:id:sgln:734005385.011.507</id>
</readPoint>
<bizLocation>
  <id>urn:epc:id:sgln:734005385.011.214</id>
</bizLocation>
```

### 8.2.4 Why

The “WHY” dimension reflects the business context (“Business Step”) of the EPCIS event, as well as the status (“Disposition”) of the object subsequent to the event. The Core Business Vocabulary (CBV) defines standard values for these.

#### 8.2.4.1 Business Step

A Business Step specifies the business process linked to the EPCIS event. Business Step identifiers are specified in section 7.1 of the CBV. The following business step identifiers are applied in this standard:

bizStep value	Description
arriving	The vehicle or train is arriving to a location or seen in location while in transit
departing	The vehicle or train is departing from a location
inspecting	The vehicle or train is stationary and being audited, for example in order to check the train formation.
transporting	The vehicle or train is observed in transit.

#### Example

*An EPCIS event is generated as a train departs the station:*

```
<bizStep>urn:epcglobal:cbv:bizstep:departing</bizStep>
```

#### 8.2.4.2 Disposition

The Disposition denotes the status of an object (i.e., the rail vehicle), subsequent to the EPCIS event. The Disposition is assumed to hold true until a subsequent event indicates a change of disposition. Disposition identifiers are specified in section 7.2 of the CBV. The following disposition identifiers are applied in this standard:

disposition value	description
in_progress	The vehicle is available for use, but not in transit.
in_transit	The vehicle is in use and included in a train composition, reserved for a train composition.

### Example

Following a vehicle's departure, it proceeds along its journey:

```
<disposition>urn:epcglobal:cbv:disp:in_transit</disposition>
```

### 8.2.4.3 Source / Destination

Source and/or Destination can be used to indicate the origin and ultimate destination of the train and rail vehicle.

It is recommended to use the `urn:epcglobal:cbv:sdt:location` source/destination type identifier, with SGLN as the location's identifier, when specifying the source or destination.



**Note:** At the time of publication, there is currently no standardised URI syntax for UN/LOCODE, although it is expected to be included in future versions of the CBV. Therefore, if there is a strong reason to use UN/LOCODE as the Source/Destination identifier, a non-standardised URI encoding would need to be established in bilateral agreement between stakeholders sharing the corresponding EPCIS events.

### Example

```
<sourceList>
  <source type="urn:epcglobal:cbv:sdt:location">
urn:epc:id:sgln:734005385.481.0</source>
</sourceList>
<destinationList>
  <destination type="urn:epcglobal:cbv:sdt:location">
urn:epc:id:sgln:734005385.011.0</destination>
</destinationList>
```

### 8.2.4.4 Business Transactions

Each EPCIS event MAY include one or more `bizTransaction` elements.

*EPCIS Business Transactions are defined using a combination of Business Transaction Type and Business Transaction ID, expressed in accordance with section 8.4 of the CBV*

The following **rail sector-specific** (non-CBV) business transaction types should be used for rail vehicle visibility applications:

business transaction type	description
urn:gs1:epcisapp:rail:btt:passage	A transaction that represents an observation of one or more rail vehicles passing a read point together, and assumed to be connected to each other.
urn:gs1:epcisapp:rail:btt:train_number	A number used by the infrastructure manager or railway operator in order to identify trains traveling on the rail network.

**Rail sector-specific** vehicle visibility applications should use HTTP URLs for business transaction identifiers, as per section 8.5.3 of CBV 1.1.

### Example of HTTP URL for business transaction identifier

`http://transaction.examplerail.com/passage/xyz12345`

To share information about a passage, the `bizTransaction` element can be used in two ways:

1. as a transaction reference in a rail visibility `ObjectEvent`, to indicate which events belong to the same 'passage' (i.e., and therefore the same trainset);
2. as a transaction type in a rail visibility `TransactionEvent`, to specify totals and tag IDs (of positively identified vehicles) for a particular 'passage' (i.e., trainset).

### Example of transaction reference

```
<bizTransactionList>
  <bizTransaction type="urn:gs1:epcisapp:rail:btt:passage">
    http://transaction.examplerail.com/passage/xyz12345</bizTransaction>
</bizTransactionList>
```

## 8.2.5 Extension elements

In this paragraph various extension elements are specified. All of these elements should be specified using the namespace `urn:gs1:epcisapp:rail`



**Note:** User-specific data can be included in EPCIS events in a similar fashion, using user-defined namespaces. Also, see section 5.2 of the EPCIS standard. However, user-specific data SHALL NOT use the `urn:gs1:epcisapp:rail` namespace. Instead, user-specific data shall use a namespace under the control of the user defining the data, such as an HTTP namespace URI using the user's Internet Domain Name. See section 8.1.3 of the CBV for more information on using HTTP URLs as user vocabulary elements.

### 8.2.5.1 Direction elements (for object and transaction events)

In order to express the train direction and vehicle orientation, the following non-CBV attributes SHALL be applied:

- `directionIndicator` is a code with the following allowed values:
  - 0 = indicates that the direction was not detected,
  - 1 = indicates the first direction in the rail network,
  - 2 = indicates the second direction in the rail network.
- `compassDirection` should be expressed using a cardinal (N, S, W, E) or inter-cardinal (NW, NE, SE, SW) compass directions, as appropriate.

### 8.2.5.2 Object event elements

To indicate vehicle orientation, the following additional elements SHALL be included in the respective vehicle's `Object` event:

- `vehicleOrientation` is a code with the following allowed values:
  - 1 = vehicle end one is leading, relative to the direction of travel,
  - 2 = vehicle end two is leading, relative to the direction of travel
  - 0 = leading vehicle end not determined
- `vehiclePosition` is a number identifying the relative position of the rail vehicle within the passage
- `vehicleAxleCount` indicates the number of axles of the vehicle.



- proxyGIAI the GIAI(s) of the observed tag(s), in EPC Pure Identity URI form.

#### Example

```
(namespace is xmlns:rail="urn:gs1:epcisapp:rail" for all elements)
<rail:directionIndicator>2</rail:directionIndicator>
<rail:compassDirection>NE</rail:compassDirection>
<rail:vehicleOrientation>2</rail:vehicleOrientation>
<rail:vehiclePosition>2</rail:vehiclePosition>
<rail:vehicleAxleCount>4</rail:vehicleAxleCount>
<rail:proxyGIAI>urn:epc:id:giai:735005385.2907412345676</rail:proxyGIAI>
```

### 8.2.6 Transaction event elements

In the passage Transaction event, the following optional, non-standard elements MAY be included:

- trainAxleCount: the total number of axles observed for the passage of the entire trainset.
- trainVehicleCount: the total number of vehicles observed for the passage of the **entire trainset**.
- vehicle: information on each of the observed vehicles
  - vehiclePosition: the relative position of the vehicle in the passage
  - vehicleAxleCount: the number of axles of the observed vehicle
  - vehicleUniquelyIdentified: indicates (true/false) whether the ID of the observed rail vehicle was captured
  - vehicleMasterGIAI: optional element specifying the ID of the rail vehicle


#### Example

```
(namespace is xmlns:rail="urn:gs1:epcisapp:rail" for all elements)
<rail:trainAxleCount>40</rail:trainAxleCount>
<rail:trainVehicleCount>10</rail:trainVehicleCount>
<rail:vehicle>
  <vehiclePosition>2</vehiclePosition>
  <vehicleAxleCount>4</vehicleAxleCount>
  <vehicleUniquelyIdentified>true</vehicleUniquelyIdentified>
  <vehicleMasterGIAI>urn:epc:id:giai:735005385.9907412345676
  </vehicleMasterGIAI>
</rail:vehicle>
```

## 9 Examples of Rail Visibility Events

In this chapter examples of event data are provided for the following use cases:

- Rail vehicle observations
- Rail vehicle changing direction
- Train passage transaction event (including untagged vehicle)

 **Note:** The following event examples are depicted in tabular form. Please refer to section 10 for the corresponding xml event examples.

### 9.1 Rail vehicle observations

In this example, two tags were observed. First tag 2 of vehicle **676**, after that tag 1 of vehicle **070**.

- In EPCIS these two events translate into two distinct **Object** events with action **OBSERVE**.
  - The first Object event relates to vehicle **676**. Its orientation is '2', meaning that end 2 is in front.
  - The second object event relates to vehicle **070**. Its orientation is '1', meaning that end 1 is in front.
- The passage ID for both observed vehicles is the same, meaning that the vehicles are assumed to be connected to each other as part of the same train set.
- The source and destination are expressing the origin and ultimate destination, using unique GLNs to represent each of the two termini.

Figure 9-1 Rail vehicle observations - Object event 1

EPCIS event	Event type	ObjectEvent		
	Action	OBSERVE		
WHEN	eventTime	2014-12-08T12:00:00.000+02:00		
WHAT	epcList	urn:epc:id:giai:735005385.9907412345 <b>676</b>		
WHERE	readPoint	urn:epc:id:sgln:734005385.011.511		
	bizLocation	urn:epc:id:sgln:734005385.011.212		
WHY	bizStep	urn:epcglobal:cbv:bizstep:arriving		
	disposition	urn:epcglobal:cbv:disp:in_progress		
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345	
	source	urn:epc:id:sgln:734005385.481.0		
	destination	urn:epc:id:sgln:734005385.011.0		
	<b>Rail extensions</b>	compassDirection	NE	
		directionIndicator	1	
		vehicleOrientation	1	
	vehiclePosition	1		
	vehicleAxleCount	4		
	proxyGIAI	urn:epc:id:giai:735005385.1907412345 <b>676</b>		

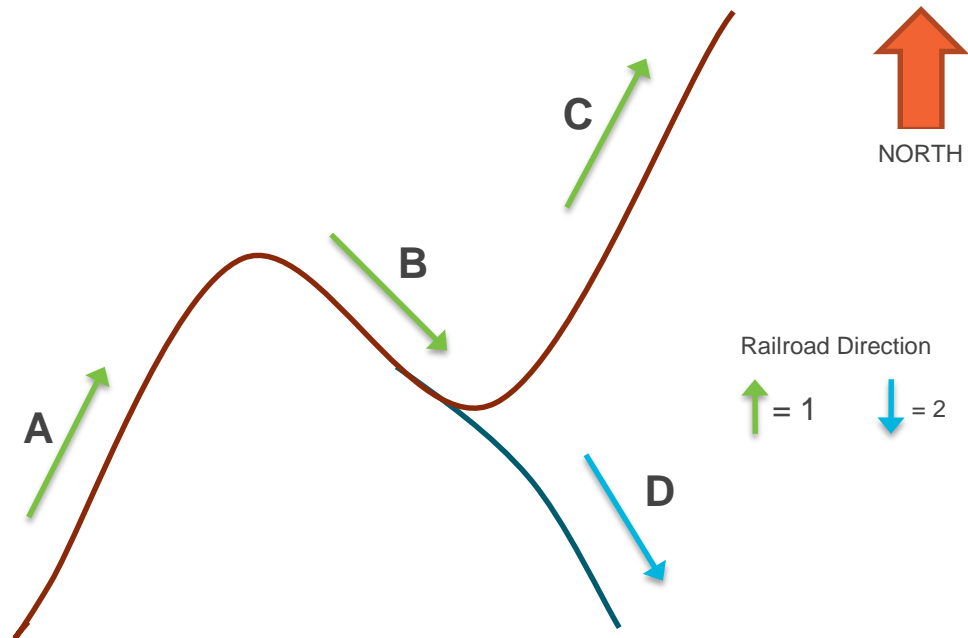
**Figure 9-2** Rail vehicle observations - Object event 2

EPCIS event	Event type	ObjectEvent		
	Action	OBSERVE		
<b>WHEN</b>	eventTime	2014-12-08T12:00:00.300+02:00		
<b>WHAT</b>	epcList	urn:epc:id:gjai:735005385.9907412345 <b>070</b>		
<b>WHERE</b>	readPoint	urn:epc:id:sgln:734005385.011.511		
	bizLocation	urn:epc:id:sgln:734005385.011.212		
<b>WHY</b>	bizStep	urn:epcglobal:cbv:bizstep:arriving		
	disposition	urn:epcglobal:cbv:disp:in_progress		
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345	
	source	urn:epc:id:sgln:734005385.481.0		
	destination	urn:epc:id:sgln:734005385.011.0		
	<i>Rail extensions</i>	compassDirection	NE	
		directionIndicator	1	
		vehicleOrientation	1	
		vehiclePosition	2	
		vehicleAxleCount	4	
	proxyGIAI	urn:epc:id:gjai:735005385.1907412345 <b>070</b>		

## 9.2 Rail vehicle changing direction

In this example we show a rail vehicle that passes three subsequent read points. It is based on the figure introduced in paragraph 7.1 and it is assuming the train passes points A, B and D.

**Figure 9-3** Illustration of direction change



Position	Railroad Direction	Compass Direction
A	1	NORTH (N)
B	1	SOUTHEAST (SE)
D	2	SOUTH (S)

**Figure 9-4** Illustration of direction change – object event 1

EPCIS event	Event type	ObjectEvent		
	Action	OBSERVE		
<b>WHEN</b>	eventTime	2014-12-08T12:00:00.000+02:00		
<b>WHAT</b>	epcList	urn:epc:id:giai:735005385.9907412345676		
<b>WHERE</b>	readPoint	urn:epc:id:sgln:734005385.011.ts4711		
	bizLocation	urn:epc:id:sgln:734005385.011.sectionA		
<b>WHY</b>	bizStep	urn:epcglobal:cbv:bizstep:transporting		
	disposition	urn:epcglobal:cbv:disp:in_transit		
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345	
	source	urn:epc:id:sgln:734005385.481.0		
	destination	urn:epc:id:sgln:734005385.011.0		
	<i>Rail extensions</i>	compassDirection	<b>N</b>	
		directionIndicator	<b>1</b>	
		vehicleOrientation	2	
		vehiclePosition	1	
	vehicleAxleCount	4		
	proxyGIAI	urn:epc:id:giai:735005385.1907412345676		

**Figure 9-5** Illustration of direction change – object event 2

EPCIS event	<i>Event type</i>	<b>ObjectEvent</b>		
	Action	OBSERVE		
<b>WHEN</b>	eventTime	2014-12-08T12:15:00.000+02:00		
<b>WHAT</b>	epcList	urn:epc:id:giai:735005385.9907412345676		
<b>WHERE</b>	readPoint	urn:epc:id:sgln:734005385.011.ts <b>4740</b>		
	bizLocation	urn:epc:id:sgln:734005385.011. <b>sectionB</b>		
<b>WHY</b>	bizStep	urn:epcglobal:cbv:bizstep:transporting		
	disposition	urn:epcglobal:cbv:disp:in_transit		
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345	
	source	urn:epc:id:sgln:734005385.481.0		
	destination	urn:epc:id:sgln:734005385.011.0		
	<b>Rail extensions</b>	compassDirection	<b>SE</b>	
		directionIndicator	<b>1</b>	
		vehicleOrientation	2	
		vehiclePosition	1	
	vehicleAxleCount	4		
	proxyGIAI	urn:epc:id:giai:735005385.1907412345676		

**Figure 9-6** Illustration of direction change – object event **3**

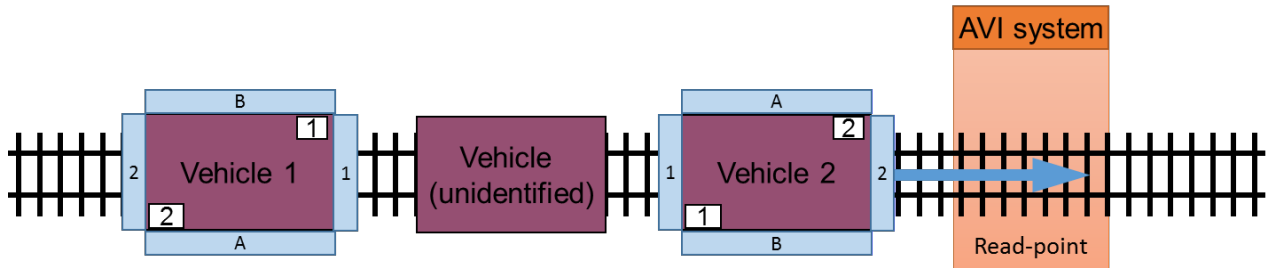
EPCIS event	Event type	ObjectEvent		
	Action	OBSERVE		
<b>WHEN</b>	eventTime	2014-12-08T12:30:00.000+02:00		
<b>WHAT</b>	epcList	urn:epc:id:giai:735005385.9907412345676		
<b>WHERE</b>	readPoint	urn:epc:id:sgln:734005385.011.ts <b>5560</b>		
	bizLocation	urn:epc:id:sgln:734005385.011.sc <b>D</b>		
<b>WHY</b>	bizStep	urn:epcglobal:cbv:bizstep:transporting		
	disposition	urn:epcglobal:cbv:disp:in_transit		
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345	
	source	urn:epc:id:sgln:734005385.481.0		
	destination	urn:epc:id:sgln:734005385.011.0		
	<b>Rail extensions</b>	compassDirection	<b>S</b>	
		directionIndicator	<b>2</b>	
		vehicleOrientation	2	
	vehiclePosition	1		
	vehicleAxleCount	4		
	proxyGIAI	urn:epc:id:giai:735005385.1907412345676		



### 9.3 Train passage event

The following example illustrates how a train passage can be expressed using a transaction event. The example shows a train passage consisting of three rail vehicles. Vehicle 2 passes first, followed by an unidentified vehicle, which is followed by vehicle 1.

**Figure 9-7** Train passage



**Note:** In this particular example, because the middle vehicle is unidentified, it does not appear in the `epcList`, nor does it have a corresponding `vehicleMasterGIAI` in the extensions.

**Figure 9-8** Train passage transaction event

EPCIS event	Event type	TransactionEvent	
	Action	ADD	
<b>WHEN</b>	eventTime	2014-12-12T12:00:00.000+02:00	
<b>WHAT</b>	epcList	urn:epc:id:giai:735005385.w2 urn:epc:id:giai:735005385.w1	
<b>WHERE</b>	readPoint	urn:epc:id:sgln:734005385.011.ts4711	
	bizLocation	urn:epc:id:sgln:734005385.011.scA	
<b>WHY</b>	bizStep	urn:epcglobal:cbv:bizstep:transporting	
	disposition	urn:epcglobal:cbv:disp:in_transit	
	bizTransactionList	urn:gs1:epcisapp:rail:btt:passage	<a href="http://transaction.examplerail.com/passage/xyz12345">http://transaction.examplerail.com/passage/xyz12345</a>
	source	urn:epc:id:sgln:734005385.481.0	
	destination	urn:epc:id:sgln:734005385.011.0	
	<b>Rail extensions</b>	trainAxleCount	12
		trainVehicleCount	3
	vehicle	vehiclePosition	1
		vehicleAxleCount	4
		vehicleUniquelyIdentified	True
		vehicleMasterGIAI	urn:epc:id:giai:735005385.w2
	vehicle	vehiclePosition	2
		vehicleAxleCount	4
		vehicleUniquelyIdentified	false
		vehicleMasterGIAI	
	vehicle	vehiclePosition	3
		vehicleAxleCount	4
		vehicleUniquelyIdentified	true
		vehicleMasterGIAI	urn:epc:id:giai:735005385.w1

## 10 EPCIS Query examples for rail vehicle visibility

The EPCIS Query Control Interface provides two modes of interaction:

1. In “on-demand” or “synchronous” mode, a client makes a request through the EPCIS Query Control Interface and receives a response immediately.
2. In “standing request” or “asynchronous” mode, a client establishes a **subscription** for a periodic query. Each time the periodic query is executed, the results are delivered asynchronously (or “pushed”) to a recipient via the EPCIS Query Callback Interface.

The EPCIS Query Callback Interface can also be used to deliver information immediately upon capture, bypassing the intermediate repository and pushed directly to the client in real-time instead.

All of these EPCIS interfaces are described in EPCIS 1.1.

### 10.1 On-demand queries via EPCIS Query Control Interface

#### 10.1.1 Observations of a specified vehicle since a specified date/time

Where and when has **vehicle 676** been seen **since 1 November, 2014 at noon UTC**?

**Figure 10-1** On-demand query by vehicle - Query parameters

Query parameter name	Value
EQ_action	OBSERVE
GE_eventTime	2014-11-01T12:00:00Z
MATCH_anyEPC	urn:epc:id:gjai: 735005385.9907412345676

#### 10.1.2 Observations of all vehicles at a specified read point in a specified window of time

What uniquely identified vehicles were seen at **waypoint4711 between 09:00 and 09:02 UTC on 1 November, 2014** (i.e., and therefore might reasonably be assumed to be **component wagons of a single trainset**)?

**Figure 10-2** On-demand query by read point - Query parameters

Query parameter name	Value
EQ_action	OBSERVE
GE_eventTime	2014-11-01T09:00:00Z
LT_eventTime	2014-11-01T09:02:00Z
EQ_readPoint	urn:epc:id:sgln: 734005385.011.ts4711

#### 10.1.3 Events for a given passage

What vehicles are currently formed in the train that passed readpoint y at time z?

If the passage ID is known the query can be formulated as follows.

**Figure 10-3** On-demand query by passage ID - Query parameters

Query parameter name	Value
EQ_bizTransaction_urn:gs1:epcisapp:rail:btt:passage	http://transaction.examplerail.com/passage/xyz12345

### 10.1.4 Passage-level queries

Provide all transaction events recorded during the week of 10-17 November, 2014 at readpoint 777777.000001.r47.

**Figure 10-4** On-demand query for passage events - Query parameters

Query parameter name	Value
eventType	TransactionEvent
GE_eventTime	2014-11-10T00:00:00Z
LT_eventTime	2014-11-17T00:00:00Z
EQ_readPoint	777777.000001.r47

## 10.2 Standing queries (Subscriptions) via EPCIS Query Call-back Interface

### 10.2.1 Notification whenever a specified vehicle is observed at any read point.

Please provide notification whenever **vehicle 676** is seen.

**Figure 10-5** Standing query by vehicle - Query parameters

Query parameter name	Value
EQ_action	OBSERVE
MATCH_anyEPC	urn:epc:id:giai:735005385.9907412345676
dest	specify the target URL – please see EPCIS 1.1, section 11.4 (“Bindings for Query Callback Interface”), for details
subscriptionID	assign the subscription a name

### 10.2.2 Notification whenever any uniquely identified vehicle is observed at a specified read point.

Please provide notification each time a uniquely (GIAI-) identified vehicle is seen **at waypoint4711**.

**Figure 10-6** Standing query by read point -Query parameters

Query parameter name	Value
EQ_action	OBSERVE
WD_readPoint	urn:epc:id:sgln:734005385.011.ts4711
dest	specify the target URL – please see EPCIS 1.1, section 11.4 (“Bindings for Query Callback Interface”), for details
subscriptionID	assign the subscription a name

## 11 Outside the scope of EPCIS: Determining rail anomalies and what HAS NOT been seen

*Which vehicles (or vehicle sides, i.e. tags) have \*not\* been observed anywhere in the last month?*

This query is not possible without knowing the vehicles that need to be checked. If these are known, a relatively straightforward query can be used. A zero result will indicate that the vehicles were not observed.

*Which trains are in an “unusual” formation, i.e. they are normally observed as “vehicle1-vehicle2-vehicle3-vehicle4”, but have just been observed as “vehicle1-vehicle5-vehicle3-vehicle4”?*

The vehicle IDs do not provide any information on the logical sequence. Other systems will need to provide that data. What can be done is providing a report of all observed vehicles per passage. This could be accomplished with a two-step query:

1. Select all relevant passages at a given read point (e.g. entering the marshalling yard). See query 8.1.4.
2. For each passage select all related events. See query 8.1.3.

## 12 Events and Queries – Examples (non-normative)

### 12.1 Example 1 – Rail vehicle observations

In this example two *tags* were observed. First tag 2 of vehicle **676**, after that tag 1 of vehicle **070**.

- In EPCIS these two events translate into two distinct **Object** events with action **OBSERVE**.
  - The first Object event relates to vehicle **676**. Its orientation is '2', meaning that end 2 is in front.
  - The second object event relates to vehicle **070**. Its orientation is '1', meaning that end 1 is in front.
- The passage ID for both observed vehicles is the same, meaning that the vehicles are assumed to be connected to each other as part of the same train set.
- The source and destination are expressing the origin and ultimate destination, using unique GLNs to represent each of the two termini.

```
<ns5:EPCISDocument xmlns:ns5="urn:epcglobal:epcis:xsd:1" schemaVersion="1.1"
creationDate="2014-10-17T11:43:05.127Z">
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        <eventTime>2014-12-08T12:00:00.000+02:00</eventTime>
        <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
        <epcList>
          <epc>urn:epc:id:giai:735005385.9907412345676</epc>
        </epcList>
        <action>OBSERVE</action>
        <bizStep>urn:epcglobal:cbv:bizstep:arriving</bizStep>
        <disposition>urn:epcglobal:cbv:disp:in_progress</disposition>
        <readPoint>
          <id>urn:epc:id:sgln:734005385.011.511</id>
        </readPoint>
        <bizLocation>
          <id>urn:epc:id:sgln:734005385.011.212</id>
        </bizLocation>
        <bizTransactionList>
          <bizTransaction
type="urn:gs1:epcisapp:rail:btt:passage">http://transaction.examplerail.com/passage/xyz12345
</bizTransaction>
          </bizTransactionList>
        <extension>
          <sourceList>
            <source
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.481.0</source>
            </sourceList>
          <destinationList>
            <destination
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.011.0</destination>
            </destinationList>
          </extension>
        <rail:compassDirection
xmlns:rail="urn:gs1:epcisapp:rail">NE</rail:compassDirection>
        <rail:directionIndicator
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:directionIndicator>
        <rail:vehicleOrientation
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehicleOrientation>
        <rail:vehiclePosition
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:vehiclePosition>
        <rail:axleCount xmlns:rail="urn:gs1:epcisapp:rail"
>8</rail:axleCount>
        <rail:proxyGIAI
xmlns:rail="urn:gs1:epcisapp:rail">urn:epc:id:giai:735005385.2907412345676</rail:proxyGIAI>
      </ObjectEvent>
      <ObjectEvent>
```

```

    <eventTime>2014-12-08T12:00:00.300+02:00</eventTime>
    <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
    <epcList>
      <epc>urn:epc:id:giai:735005385.9907412345070</epc>
    </epcList>
    <action>OBSERVE</action>
    <bizStep>urn:epcglobal:cbv:bizstep:arriving</bizStep>
    <disposition>urn:epcglobal:cbv:disp:in_progress</disposition>
    <readPoint>
      <id>urn:epc:id:sgln:734005385.011.511</id>
    </readPoint>
    <bizLocation>
      <id>urn:epc:id:sgln:734005385.011.212</id>
    </bizLocation>
    <bizTransactionList>
      <bizTransaction
type="urn:gs1:epcisapp:rail:btt:passage">http://transaction.examplerail.com/passage/xyz12345
</bizTransaction>
      </bizTransactionList>
    <extension>
      <sourceList>
        <source
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.481.0</source>
        </sourceList>
        <destinationList>
          <destination
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.011.0</destination>
          </destinationList>
        </extension>
      <rail:compassDirection
xmlns:rail="urn:gs1:epcisapp:rail">NE</rail:compassDirection>
      <rail:directionIndicator
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:directionIndicator>
      <rail:vehicleOrientation
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:vehicleOrientation>
      <rail:vehiclePosition
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehiclePosition>
      <rail:axleCount
xmlns:rail="urn:gs1:epcisapp:rail">8</rail:axleCount>
      <rail:proxyGIAI
xmlns:rail="urn:gs1:epcisapp:rail">urn:epc:id:giai:735005385.1907412345070</rail:proxyGIAI>
    </ObjectEvent>
  </EventList>
</EPCISBody>
</ns5:EPCISDocument>

```



**Note:** In this example the rail extension elements are nested in a rail element. This leads to a more compact XML representation but will prevent the contained elements from being queried via EPCIS Query Control Interface. If this is required the elements need to be specified as direct children of the objectEvent element.

## 12.2 Example 2 – Rail vehicle changing direction

In this example we show a rail vehicle that passes three subsequent read points. It is based on the figure introduced in paragraph 5.1, and assumes that the train passes points A, B and D.

```

<ns5:EPCISDocument xmlns:ns5="urn:epcglobal:epcis:xsd:1" schemaVersion="1.1"
creationDate="2014-10-17T11:43:05.127Z">
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        <eventTime>2014-12-08T12:00:00.000+02:00</eventTime>
        <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
        <epcList>
          <epc>urn:epc:id:giai:735005385.9907412345676</epc>
        </epcList>
        <action>OBSERVE</action>
        <bizStep>urn:epcglobal:cbv:bizstep:transporting</bizStep>
        <disposition>urn:epcglobal:cbv:disp:in_transit</disposition>
        <readPoint>
          <id>urn:epc:id:sgln:734005385.011.ts4711</id>
        </readPoint>
        <bizLocation>
          <id>urn:epc:id:sgln:734005385.011.scA</id>
        </bizLocation>
        <bizTransactionList>
          <bizTransaction
type="urn:gs1:epcisapp:rail:btt:passage">http://transaction.examplerail.com/passage/xyz12345
</bizTransaction>
          </bizTransactionList>
        <extension>
          <sourceList>
            <source
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.481.0</source>
            </sourceList>
          <destinationList>
            <destination
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.011.0</destination>
            </destinationList>
          </extension>
          <rail:compassDirection
xmlns:rail="urn:gs1:epcisapp:rail">N</rail:compassDirection>
          <rail:directionIndicator
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:directionIndicator>
          <rail:vehicleOrientation
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehicleOrientation>
          <rail:vehiclePosition
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:vehiclePosition>
          <rail:axleCount xmlns:rail="urn:gs1:epcisapp:rail">8</rail:axleCount>
          <rail:proxyGIAI
xmlns:rail="urn:gs1:epcisapp:rail">urn:epc:id:giai:735005385.2907412345676</proxyGIAI:proxyG
IAI>
        </ObjectEvent>
      <ObjectEvent>
        <eventTime>2014-12-08T12:15:00.000+02:00</eventTime>
        <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
        <epcList>
          <epc>urn:epc:id:giai:735005385.9907412345676</epc>
        </epcList>
        <action>OBSERVE</action>
        <bizStep>urn:epcglobal:cbv:bizstep:transporting</bizStep>
        <disposition>urn:epcglobal:cbv:disp:in_transit</disposition>
        <readPoint>
          <id>urn:epc:id:sgln:734005385.011.ts4740</id>
        </readPoint>
        <bizLocation>
          <id>urn:epc:id:sgln:734005385.011.scB</id>

```





```
        </bizLocation>
        <bizTransactionList>
          <bizTransaction
type="urn:gs1:epcisapp:rail:btt:passage">http://transaction.examplerail.com/passage/xyz12345
</bizTransaction>
          </bizTransactionList>
        <extension>
          <sourceList>
            <source
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.481.0</source>
            </sourceList>
          <destinationList>
            <destination
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.011.0</destination>
            </destinationList>
          </extension>
          <rail:compassDirection xmlns:rail="urn:gs1:epcisapp:rail"
>SE</rail:compassDirection>
            <rail:directionIndicator
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:directionIndicator>
            <rail:vehicleOrientation
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehicleOrientation>
            <rail:vehiclePosition
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehiclePosition>
            <rail:axleCount xmlns:rail="urn:gs1:epcisapp:rail">8</rail:axleCount>
            <rail:proxyGIAI
xmlns:rail="urn:gs1:epcisapp:rail">urn:epc:id:giai:735005385.2907412345676</rail:proxyGIAI>
          </ObjectEvent>
        <ObjectEvent>
          <eventTime>2014-12-08T12:30:00.000+02:00</eventTime>
          <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
          <epcList>
            <epc>urn:epc:id:giai:735005385.9907412345676</epc>
          </epcList>
          <action>OBSERVE</action>
          <bizStep>urn:epcglobal:cbv:bizstep:transporting</bizStep>
          <disposition>urn:epcglobal:cbv:disp:in_transit</disposition>
          <readPoint>
            <id>urn:epc:id:sgln:734005385.011.ts5560</id>
          </readPoint>
          <bizLocation>
            <id>urn:epc:id:sgln:734005385.011.scD</id>
          </bizLocation>
          <bizTransactionList>
            <bizTransaction
type="urn:gs1:epcisapp:rail:btt:passage">http://transaction.examplerail.com/passage/xyz12345
</bizTransaction>
            </bizTransactionList>
          <extension>
            <sourceList>
              <source
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.481.0</source>
              </sourceList>
            <destinationList>
              <destination
type="urn:epcglobal:cbv:sdt:location">urn:epc:id:sgln:734005385.011.0</destination>
              </destinationList>
            </extension>
            <rail:compassDirection
xmlns:rail="urn:gs1:epcisapp:rail">S</rail:compassDirection>
              <rail:directionIndicator
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:directionIndicator>
              <rail:vehicleOrientation
xmlns:rail="urn:gs1:epcisapp:rail">2</rail:vehicleOrientation>
              <rail:vehiclePosition
xmlns:rail="urn:gs1:epcisapp:rail">1</rail:vehiclePosition>
```



```
                <rail:axleCount
xmlns:rail="urn:gs1:epcisapp:rail">8</rail:axleCount>
                <rail:proxyGIAI
xmlns:rail="urn:gs1:epcisapp:rail">urn:epc:id:giai:735005385.2907412345676</rail:proxyGIAI>
            </ObjectEvent>
        </EventList>
    </EPCISBody>
</ns5:EPCISDocument>
```



### 12.3 Example 3 – Train passage Transaction event

The following example illustrates how a train passage can be expressed using a transaction event. The example shows a train passage consisting of three rail vehicles. Vehicle 2 passes first, followed by an unidentified vehicle, which is followed by vehicle 1.

```
<ns5:EPCISDocument xmlns:ns5="urn:epcglobal:epcis:xsd:1" schemaVersion="1.1"
creationDate="2014-10-17T11:43:05.127Z">
  <EPCISBody>
    <EventList>
      <TransactionEvent>
        <eventTime>2014-12-12T12:00:00.000+02:00</eventTime>
        <eventTimeZoneOffset>+02:00</eventTimeZoneOffset>
        <bizTransactionList>
          <bizTransaction type="urn:gs1:epcisapp:rail:btt:passage">
http://transaction.examplerail.com/passage/xyz12345</bizTransaction>
          </bizTransactionList>
          <epcList>
            <epc>urn:epc:id:giai:735005385.w2</epc>
          </epcList>
          <action>ADD</action>
          <bizStep>urn:epcglobal:cbv:bizstep:transporting</bizStep>
          <disposition>urn:epcglobal:cbv:disp:in_transit</disposition>
          <readPoint>
            <id>urn:epc:id:sgln:734005385.011.ts4711</id>
          </readPoint>
          <bizLocation>
            <id>urn:epc:id:sgln:734005385.011.scA</id>
          </bizLocation>
          <extension>
            <sourceList>
              <source type="urn:epcglobal:cbv:sdt:location">
urn:epc:id:sgln:734005385.481.0</source>
            </sourceList>
            <destinationList>
              <destination type="urn:epcglobal:cbv:sdt:location">
urn:epc:id:sgln:734005385.011.0</destination>
            </destinationList>
          </extension>
          <rail:trainAxleCount xmlns:rail="urn:gs1:epcisapp:rail"
>12</rail:trainAxleCount>
          <rail:trainVehicleCount xmlns:rail="urn:gs1:epcisapp:rail"
>3</rail:trainVehicleCount>
          <rail:vehicle xmlns:rail="urn:gs1:epcisapp:rail"
xmlns="urn:gs1:epcisapp:rail">
            <vehiclePosition>1</vehiclePosition>
            <vehicleAxleCount>4</vehicleAxleCount>
            <vehicleUniquelyIdentified>true</vehicleUniquelyIdentified>
            <vehicleMasterGIAI>urn:epc:id:giai:735005385.vehicle2</vehicleMasterGIAI>
          </rail:vehicle>
          <rail:vehicle xmlns:rail="urn:gs1:epcisapp:rail"
xmlns="urn:gs1:epcisapp:rail">
            <vehiclePosition>2</vehiclePosition>
            <vehicleAxleCount>4</vehicleAxleCount>
            <vehicleUniquelyIdentified>>false</vehicleUniquelyIdentified>
            <vehicleMasterGIAI/>
          </rail:vehicle>
          <rail:vehicle xmlns:rail="urn:gs1:epcisapp:rail"
xmlns="urn:gs1:epcisapp:rail">
            <vehiclePosition>3</vehiclePosition>
            <vehicleAxleCount>4</vehicleAxleCount>
            <vehicleUniquelyIdentified>true</vehicleUniquelyIdentified>
            <vehicleMasterGIAI>urn:epc:id:giai:735005385.vehicle1</vehicleMasterGIAI>
          </rail:vehicle>
        </TransactionEvent>
      </EventList>
    </EPCISBody>
  </ns5:EPCISDocument>
```



</ns5:EPCISDocument>

## 12.4 Example 4 – Query for observations of a specified vehicle since a specified date/time

Where and when has **vehicle 676** been seen **since 1 November, 2014 at noon UTC?**

```
<ns3:EPCISQueryDocument xmlns:ns3="urn:epcglobal:epcis-query:xsd:1" schemaVersion="1"
creationDate="2014-12-12T07:27:03.898Z">
  <EPCISBody>
    <ns3:Poll>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>EQ_action</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>OBSERVE</string>
          </value>
        </param>
        <param>
          <name>GE_eventTime</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xsi:type="xs:dateTime">2014-11-
01T12:00:00Z</value>
        </param>
        <param>
          <name>MATCH_anyEPC</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>urn:epc:id:giai:735005385.9907412345676</string>
          </value>
        </param>
      </params>
    </ns3:Poll>
  </EPCISBody>
</ns3:EPCISQueryDocument>
```

## 12.5 Example 5 – Query for observations of all vehicles at a specified read point in a specified window of time

*What uniquely identified vehicles were seen at waypoint4711 between 09:00 and 09:02 UTC on 1 November, 2014 UTC? (i.e. and therefore might reasonably be assumed to be component wagons of a single trainset)*

```

<ns3:EPCISQueryDocument xmlns:ns3="urn:epcglobal:epcis-query:xsd:1" schemaVersion="1"
creationDate="2014-12-12T07:45:08.966Z">
  <EPCISBody>
    <ns3:Poll>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>EQ_action</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>OBSERVE</string>
          </value>
        </param>
        <param>
          <name>GE_eventTime</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xsi:type="xs:dateTime">2014-11-
01T09:00:00Z</value>
        </param>
        <param>
          <name>LT_eventTime</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xsi:type="xs:dateTime">2014-12-
01T09:02:00Z</value>
        </param>
        <param>
          <name>WD_readPoint</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>urn:epc:id:sgln:734005385.011.ts4711</string>
          </value>
        </param>
      </params>
    </ns3:Poll>
  </EPCISBody>
</ns3:EPCISQueryDocument>

```

## 12.6 Example 6 – Query for all events for a given passage

*What vehicles belonged to a train that passed readpoint y at time z, given a specified passage ID?*

```
<ns3:EPCISQueryDocument xmlns:ns3="urn:epcglobal:epcis-query:xsd:1" schemaVersion="1"
creationDate="2014-12-12T07:27:03.898Z">
  <EPCISBody>
    <ns3:Poll>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>EQ_bizTransaction_urn:gs1:epcisapp:rail:btt:passage</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>passage-123456</string>
          </value>
        </param>
      </params>
    </ns3:Poll>
  </EPCISBody>
</ns3:EPCISQueryDocument>
```

## 12.7 Example 7 – Passage-level queries

*Provide all transaction events recorded during the week of 10-17 November 2014 at readpoint 777777.000001.*

```
<ns3:EPCISQueryDocument xmlns:ns3="urn:epcglobal:epcis-query:xsd:1" schemaVersion="1"
creationDate="2014-11-17T12:22:59.523Z">
  <EPCISBody>
    <ns3:Poll>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>eventType</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>TransactionEvent</string>
          </value>
        </param>
        <param>
          <name>GE_eventTime</name>
          <value xsi:type="xs:dateTime">2014-11-10T00:00:00Z</value>
        </param>
        <param>
          <name>LT_eventTime</name>
          <value xsi:type="xs:dateTime">2014-11-17T00:00:00Z</value>
        </param>
        <param>
          <name>EQ_readPoint</name>
          <value xsi:type="ns3:ArrayOfString">
            <string>urn:epc:id:sgln:7777777.000001.r47</string>
          </value>
        </param>
      </params>
    </ns3:Poll>
  </EPCISBody>
</ns3:EPCISQueryDocument>
```



## 12.8 Example 8 – Standing query (subscription) for notification whenever a specified vehicle is observed at any read point

Please provide notification whenever **vehicle 676** is seen.

```
soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:epcglobal:epcis-query:xsd:1">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:Subscribe>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>EQ_action</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>OBSERVE</string>
          </value>
        </param>
        <param>
          <name>MATCH_anyEPC</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>urn:epc:id:giai:735005385.9907412345676</string>
          </value>
        </param>
      </params>
      <dest>http://specifyYourTargetURLhere</dest>
      <subscriptionID>specifySubscriptionNameHere</subscriptionID>
    </urn:Subscribe>
  </soapenv:Body>
</soapenv:Envelope>
```

## 12.9 Example 9 – Standing query (subscription) for notification whenever any uniquely identified vehicle is observed at a specified read point.

*Please provide notification each time a uniquely (GIAI-) identified vehicle is seen **at waypoint4711**.*

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:epcglobal:epcis-query:xsd:1">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:Subscribe>
      <queryName>SimpleEventQuery</queryName>
      <params>
        <param>
          <name>EQ_action</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>OBSERVE</string>
          </value>
        </param>
        <param>
          <name>WD_readPoint</name>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns3:ArrayOfString">
            <string>urn:epc:id:sgln:734005385.011.ts4711</string>
          </value>
        </param>
      </params>
      <dest>http://specifyYourTargetURLhere</dest>
      <subscriptionID>specifySubscriptionNameHere</subscriptionID>
    </urn:Subscribe>
  </soapenv:Body>
</soapenv:Envelope>

```

## 13 Abbreviations

AABM	Acoustic Axle Bearing Monitoring
APMS	Automatic pantograph monitoring systems
AVI	Automatic Vehicle Identification system
CBV	EPCIS Core Business Vocabulary
GIAI	Global Individual Asset Identifier
GLN	Global Location Number
HABD	Hot Axle-Box Detectors
RFID	Radio Frequency Identification
SGLN	Global Location Number in an EPCIS structure
TDS	GS1 Tag Data Standard
TMS	Train Management System
UN/LOCODE	United Nations Code for Trade and Transport Locations
URI	Uniform resource identifier
WILD	Wheel impact load detectors
WTMS	Wayside Train Monitoring System