

IBIS-IP Beschreibung der Dienste / Service description

DIENST-TrainSetInformationService / SERVICE-
TrainSetInformationService

DIENST-TrainSetManagementService / SERVICE-
TrainSetManagementService

DIENST-TrainSetDataService / SERVICE-TrainSetDataService
V2.2

Gesamtbearbeitung

Ausschuss für Telematik und Informationssysteme (ATI)

Gefördert durch:



Das dieser VDV-Schrift zugrundeliegende Vorhaben IP-KOM-ÖV wurde mit Mitteln des Bundesministeriums für Wirtschaft und Energie unter dem Förderkennzeichen 19P10003 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

aufgrund eines Beschlusses
des Deutschen Bundestages

IBIS-IP Beschreibung der Dienste / Service description

DIENST-TrainSetInformationService / SERVICE-
TrainSetInformationService
DIENST-TrainSetManagementService / SERVICE-
TrainSetManagementService
DIENST-TrainSetDataService / SERVICE-TrainSetDataService
V2.2

Sachbearbeitung

Unterausschuss für Telematik
(UA Telematik)

Autorenverzeichnis

Dr. Torsten Franke, IVU, Aachen
Michael Grotens, IVU, Aachen
René Fischli, Trapeze, Neuhausen
Horst Sander, ATRON, Markt Schwaben
Martin Streicher, ATRON, Markt Schwaben
Felix Elgner, Init, Karlsruhe

Der Anwender ist für die sorgfältige und ordnungsgemäße Anwendung der Schrift verantwortlich. Stellt der Anwender Gefährdungen oder Unregelmäßigkeiten im Zusammenhang mit der Anwendung dieser Schrift fest, wird eine unmittelbare Benachrichtigung an den VDV erbeten. Eine Haftung des VDV oder der Mitwirkenden an der Schrift ist, soweit gesetzlich zulässig, ausgeschlossen.

© Verband Deutscher Verkehrsunternehmen e. V. Köln 2015 | Alle Rechte, einschließlich des Nachdrucks von Auszügen, der fotomechanischen oder datenverarbeitungstechnischen Wiedergabe und der Übersetzung, vorbehalten.

Vorwort

Im Forschungsprojekt „Internet Protokoll basierte Kommunikationsdienste im ÖV - IP-KOM-ÖV“, gefördert vom Bundesministerium für Wirtschaft und Energie BMWi, wurde das Grundkonzept von IBIS-IP für die Fahrgastinformation entwickelt. Im Projekt wurde nur die Kommunikation innerhalb eines Fahrzeuges / Wagen betrachtet.

Der Betrieb von gekuppelten Wagen in einem Zugverband ist jedoch die übliche Praxis in Straßen-, Stadt und U-Bahn Verkehrsunternehmen. Für die Bereitstellung einer konsistenten Fahrgastinformation ist somit eine Kommunikation zwischen den Wagen notwendig.

Die vorliegende VDV-Schrift 301-2-14 beschreibt das Konzept und die notwendigen neuen drei IBIS-IP-Dienste für diese Kommunikation in einem aus mehreren Wagen bestehenden Zugverband und ermöglicht somit eine standardisierte Nutzung von IBIS-IP im Straßen-, Stadt und U-Bahn-Betrieb.

Foreword

In the research project "Internet Protocol based communication services in public transport - IP-KOM-ÖV", funded by the Federal Ministry of Economics and Energy BMWi, the basic concept of IBIS-IP for passenger information was developed. In the project, only the communication within a vehicle/coach was considered.

However, the operation of coupled wagons in a trainset is a common practice in urban rail transport companies. In order to provide consistent passenger information, communication between the wagons is therefore necessary.

This VDV requirement document 301-2-14 describes the concept and the necessary new three IBIS-IP services for this communication in a trainset consisting of several coaches and thus enables a standardised use of IBIS IP in urban rail operation.

Inhaltsverzeichnis / Content

Vorwort.....	4
Foreword	4
Inhaltsverzeichnis / Content	5
Abkürzungen / Abbreviations	8
1 Ausgangslage und Übersicht	9
1 Initial Situation and Overview.....	10
2 IBIS-IP Netzwerkkonzept für Traktionen	11
2.1 <i>Trennung der Netzwerke</i>	12
2.2 <i>Beispiele von Netzwerken</i>	12
2.2.1 Bordrechner mit integriertem managed Layer 2 Switch	13
2.2.2 Kombination mit managed Layer 2 Switch	13
2.2.3 Kombination mit 2 managed Layer 3 Switch mit redundanter Ringleitung	14
2.3 <i>Kupplungsübergang</i>	14
2 IP Network Concept for Trainsets.....	15
2.1 <i>Separation of the network</i>	16
2.2 <i>Network Examples</i>	16
2.2.1 On-board computer with integrated managed layer 2 switch	17
2.2.2 Combination with managed layer 2 switch	17
2.2.3 Combination with 2 managed layer 3 switches with redundant ring line	18
2.3 <i>Coupling transition</i>	18
3 Einführung in das Service-Konzept.....	19
3.1 <i>Flexibles Servicekonzept</i>	19
3.2 <i>Funktionen und Aufgaben der Dienste</i>	19
3.3 <i>Grundlagen der Bahnkommunikation in IBIS-IP</i>	19
3 Introduction to the Service Concept.....	21
3.1 <i>Flexible Service Concept</i>	21
3.2 <i>Functions and Tasks of the Services</i>	21
3.3 <i>Basic Principles for Trainset Communication in IBIS-IP</i>	21
4 TrainSetInformationService	23
4.1 <i>Purpose</i>	23
4.2 <i>Where is the service provided and by whom?</i>	23
4.3 <i>Service Publication via DNS-SD</i>	23
4.4 <i>Basic Protocols</i>	24

4.5	<i>Operations of the TrainSetInformationService</i>	24
4.5.1	GetTrainSetComposition	24
4.5.2	SubscribeTrainSetComposition	26
4.5.3	UnsubscribeTrainSetComposition	26
5	TrainSetManagementService	27
5.1	<i>Purpose</i>	27
5.2	<i>Where is the service provided and by whom?</i>	28
5.3	<i>Service Publication via DNS-SD</i>	28
5.4	<i>Basic Protocols</i>	28
5.5	<i>Operations of the TrainSetManagementService</i>	29
5.5.1	Operation SetSlaveMode	29
5.5.2	Operation SetNeutralMode	30
5.5.3	GetTrainSetComposition	31
5.5.4	SubscribeTrainSetComposition	32
5.5.5	UnsubscribeTrainSetComposition	32
6	TrainSetDataService	33
6.1	<i>Purpose</i>	33
6.2	<i>Where is the service provided and by whom?</i>	33
6.3	<i>Service Publication via DNS-SD</i>	34
6.4	<i>Basic Protocols</i>	34
6.5	<i>Operations of the TrainSetDataService</i>	34
6.5.1	Specific TrainSetSubscribeRequestStructure	35
6.5.2	Specific TrainSetUnsubscribeRequestStructure	35
6.5.3	Operation RetrieveTripRef	36
6.5.4	Operation SubscribeTripRef	37
6.5.5	Operation UnsubscribeTripRef	38
6.5.6	Operation RetrieveTripInformation	38
6.5.7	Operation SubscribeTripInformation	40
6.5.8	Operation UnsubscribeTripInformation	40
7	Service-to-Coach Assignment	41
8	Implementation Guide Line for existing IBIS-IP services	42
8.1	<i>General Approach for the Handling of Proxy Services</i>	42
8.2	<i>Specifics for HTTP proxy services</i>	42
8.3	<i>Specifics for UDP proxy services</i>	43
9	Re-Initialisation of valid Services.....	44
9.1	<i>Re-initialisation of Service Communication when a Trainset changes the Driving Direction</i>	44
9.1.1	Use case	44
9.1.2	Possible implementation for OBUs	44
9.1.3	Possible Implementation for Non-OBU Devices in the Trainset IP Network	45
9.1.4	Possible implementation for non-OBU devices in the coach IP network	46
10	Examples	47
10.1	<i>The master OBU requests the Status of Devices in a Slave Coach</i>	47

10.1.1	Use Case	47
10.1.2	Possible Implementation	47
10.2	<i>Master OBU provides Passenger Information for the Slave Coaches</i>	48
10.2.1	Use case	48
10.2.2	Possible Implementation	48
10.3	<i>Master OBU provides Trip Information for Slave OBUs in X-Y-Traffic</i>	49
10.3.1	Use case	49
10.3.2	Possible Implementation	49
10.4	<i>Master OBU provides recent Positioning Information for the Slave OBUs</i>	50
10.4.1	Use Case	50
10.4.2	Possible Implementation	50
11	Versionshistorie / Version history.....	51
11.1	<i>Version 2.2</i>	51
11.1.1	Funktionale Erweiterungen Functional Upgrade	51
11.1.2	Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections	51
11.1.3	Textliche Korrekturen Textual Corrections	51
Regelwerke – Normen und Empfehlungen / References		52
Impressum		53

Abkürzungen / Abbreviations

Die bereits in der VDV 301-1 definierten Abkürzungen werden an dieser Stelle nicht wiederholt.

The abbreviations already defined in VDV 301-1 are not repeated here.

1 Ausgangslage und Übersicht

Bislang fehlt in IBIS-IP eine Beschreibung, wie IBIS-IP-Dienste in einem aus mehreren Wagen (bzw. Waggonen) bestehenden Zugverband zu handhaben sind. Das vorliegende Dokument schließt diese Lücke durch drei neue Dienste.

Die Dienste sind zwar nicht sehr komplex, werden aber in komplexen Szenarien und nur für eine bestimmte Netzkonstellation eingesetzt. Deshalb enthält dieses Dokument fünf Dinge:

- 1.) Es beschreibt das zugrunde liegende IP-Netzwerkkonzept (Abschnitt 2).
- 2.) Er beschreibt die wesentlichen Anforderungen, Dienste und Operationen, die für den Zugbetrieb und die Anwendungsfälle spezifisch sind (Abschnitte 3 bis 6).
- 3.) Es beschreibt, wie bereits in IBIS-IP beschriebene Leistungen in einem Zugverband abzuwickeln sind, d.h. wie aus dem Zugverbundnetz auf die wageninternen Leistungen zugegriffen werden kann (Abschnitt 8).
- 4.) Es beschreibt, wie praktische Probleme, wie die Zuordnung eines Dienstes zu einem Wagen oder die Neuinitialisierung des Dienstes nach einer Änderung der Zugverbandszusammenstellung zu lösen sind (Abschnitte 7 und 9).
- 5.) Anhand von Beispielen wird das Zusammenspiel der verschiedenen Dienste erläutert (Abschnitt 9.1).

Im Folgenden verwenden wir die folgende Sprechweise:

Ein Zugverband besteht aus mehreren Wagen:



Ein Wagen im Sinne der VDV 301-2-14 entspricht dabei dem „Consist“ aus der IEC61375.

1 Initial Situation and Overview

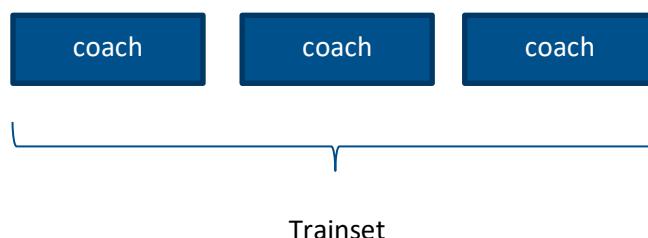
Up to now IBIS-IP lacks a description on how to handle IBIS-IP services in a trainset consisting of more than one coach. The document at hand closes this gap by means of three new services.

Although the services are not very complex, they are used in complex scenarios and only for a certain network constellation. That is why this document provides five things:

- 1st) It describes the underlying IP network concept (section 2)
- 2nd) It describes the key requirements, services and operations that are specific for the trainset operations and use cases (sections 3 to 6).
- 3rd) It describes how services which are already described in IBIS-IP are to be handled in a trainset, i.e. how coach internal services can be accessed from the trainset network (section 8).
- 4th) It describes how to solve practical problems like the assignment of a service to a coach or how to organise the re-initialisation of service after a change of the trainset composition (sections 7 and 9).
- 5th) By means of examples the interaction of the different services is explained (section 9.1).

In the following we use the following way of speaking:

A trainset consists of several coaches:



A coach in the sense of VDV 301-2-14 corresponds to the term „consist“ of IEC61375.

2 IBIS-IP Netzwerkkonzept für Traktionen

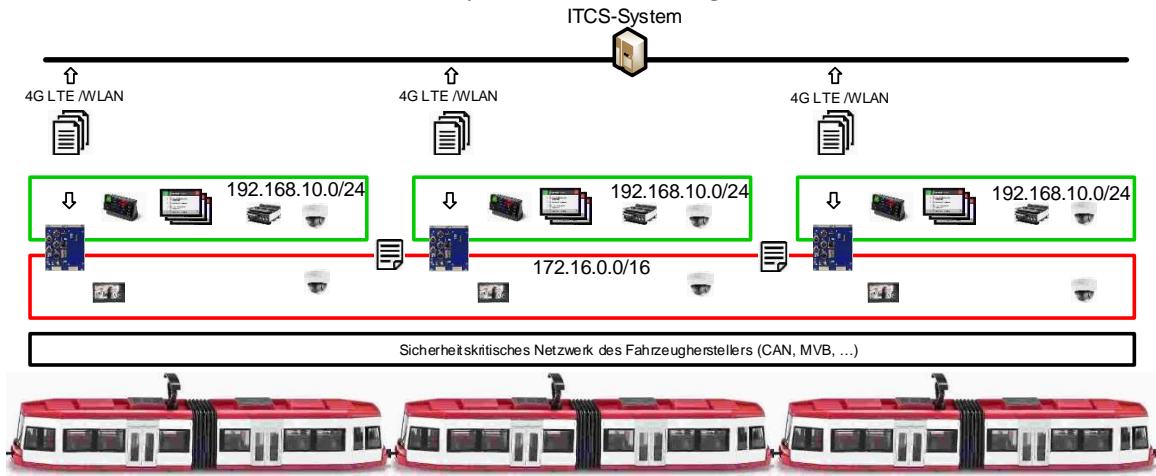
IBIS-IP Geräte in einem Wagen sollen mit IBIS-IP Geräte im gleichen und bei Bedarf in anderen Wagen kommunizieren können. Dazu werden zwei getrennte Netzwerke vorgeschlagen:

- Wagen-IP-Netzwerk (grün im Bild), in welchem die Kommunikation innerhalb des Wagens erfolgt
- Ein Zugverbands-IP-Netzwerk (rot im Bild)
Ein „Zugverbands-IP-Netzwerk“ ist ein IP-Netzwerk, das der Kommunikation zwischen Wagen dient, aber ohne irgendwelche sicherheitskritischen Teile eines train backbones (Im Sinne der IEC 61375) zu tangieren.

Ein Gerät, das in beiden Netzwerken kommunizieren möchte, muss sich in beiden Netzwerken befinden. Dies ist möglich durch zwei Netzwerk-Schnittstellen (Ethernet-Schnittstellen) oder die Verwendung einer Schnittstelle mit zwei IP-Adressen.

Grundsätzlich gehen wir davon aus, dass ein IP-Netzwerk des Fahrzeugherrstellers nicht verwendet werden kann, da dies sicherheitskritisch ist. Geräte aus dem sicherheitskritischen Netzwerk können über Schnittstellen angebunden werden, z.B. zu Überwachungszwecken.

Wie die IP-Kommunikation über die Kupplung erfolgt, ist für die nachfolgenden Betrachtungen nicht relevant. Das Thema wird in einem separaten Abschnitt (vgl. 2.3) betrachtet.



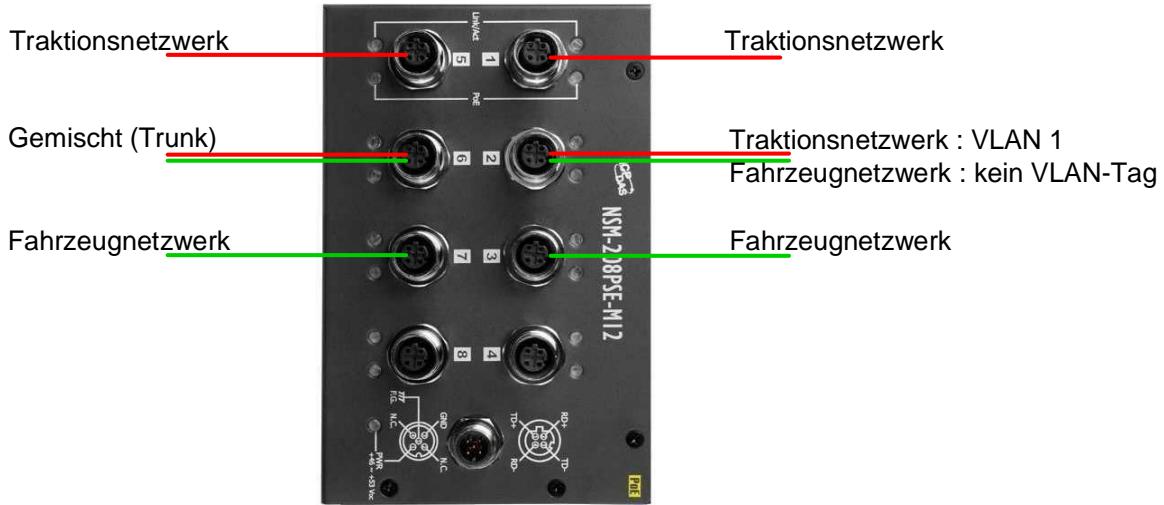
Der Einfachheit halber gehen wir in den nachfolgenden Beispielen davon aus, dass in allen Wagen-IP-Netzwerken gleiche Adressen für gleiche Geräte verwendet werden, jeweils aus dem Adressbereich 192.168.10.0/24. Auch besteht eine feste Zuordnung einer Fahrzeugnummer zu einem Adressbereich im Zugverbands-IP-Netzwerk, z.B. nach folgendem Schema 172.16.x.y, wobei x abhängig von der Fahrzeugnummer ist und y identisch zur letzten Stelle der IP-Adresse des 192.168.10.0/24 Netzwerk ist.

Es können aber in der Praxis andere Adressenbereiche und auch DHCP verwendet werden.

Doppelte IP-Adressen in nicht getrennten Netzwerken müssen vermieden werden.

2.1 Trennung der Netzwerke

Werden gemeinsame Netzwerkleitungen für beide Netzwerke verwendet werden, muss das Wagen-IP-Netzwerk an den Kupplungen begrenzt werden, da es sonst zu Adresskonflikten kommen kann. Diese Begrenzung kann durch VLAN (gemäß IEEE 802.1Q) erreicht werden. Dazu werden Datenpakete des Traktionsnetzwerk mit dem VLAN 1 gekennzeichnet. Die Datenpakete des Wagen-IP-Netzwerks haben kein VLAN-Tag. Bei Bedarf kann nun eine Trennung durch eine managed Layer 2 Switch realisiert werden.



2.2 Beispiele von Netzwerken

Die folgenden Unterkapitel zeigen 3 prinzipielle Beispiele auf, wie diese Netzwerke aufgebaut werden können. Wesentlich ist, dass die Netzwerkverbindungen zum benachbarten Fahrzeug frei von Datenpaketen des Fahrzeuglokalen Netzwerkes sind.

Die nachfolgenden Darstellungen zeigen je ein Fahrzeug, mit den Kupplungen links und rechts.

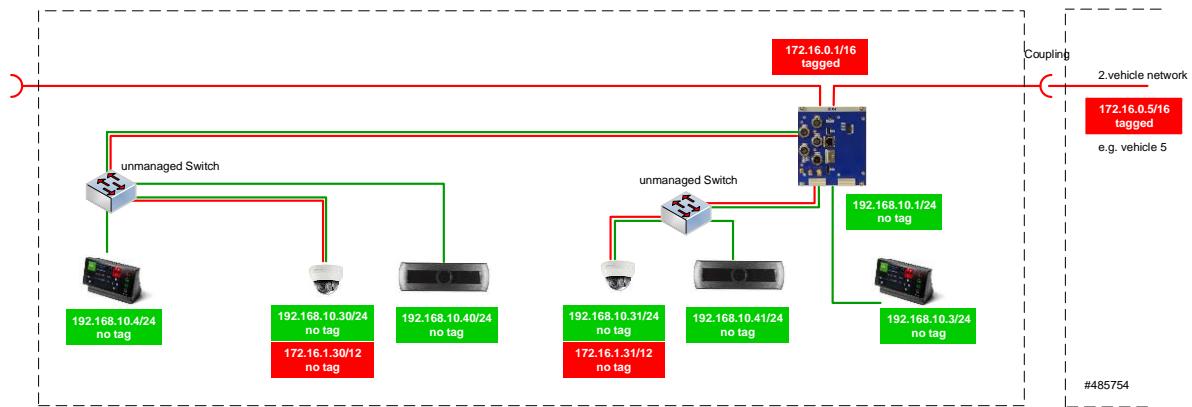
Das lokale Wagen-IP-Netzwerk ist grün und das Zugverbands-IP-Netzwerk rot eingezeichnet. Gemeinsame Netzwerkverbindungen sind rot und grün gekennzeichnet.

Welche Geräte sich nun in beiden Netzwerken befinden hängt vom Anwendungsfall ab.

Mögliche Kriterien für die konkrete Architektur sind:

- Benötigte Kommunikationsbandbreite
- Verfügbarkeit & Redundanz (Kabelbruch und Geräteausfall)
- Kosten
- Vorhandene nutzbare Kabel, Installationsmöglichkeiten
- Konfiguration von Komponenten vermeiden (Austauschfall)
- Erweiterbarkeit

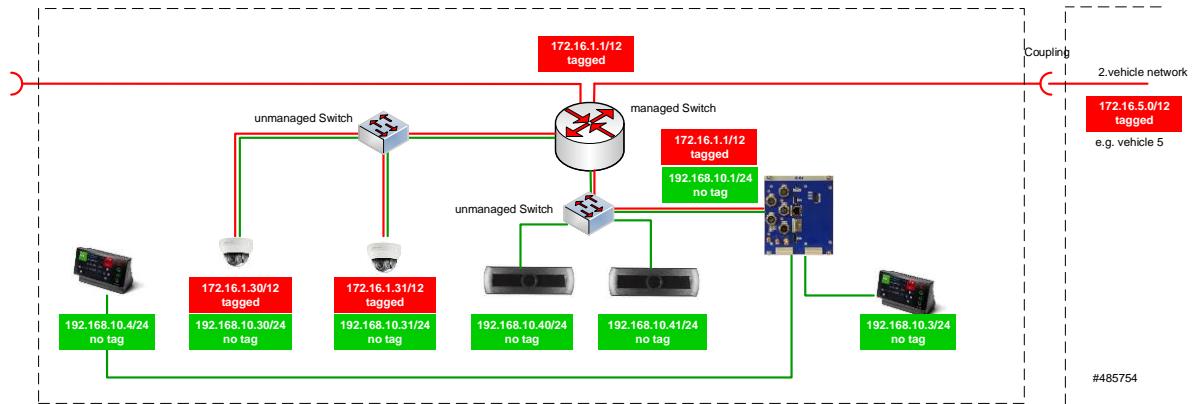
2.2.1 Bordrechner mit integriertem managed Layer 2 Switch



In diesem Beispiel werden für das Wagen-IP-Netzwerk nur unmanaged Switches benötigt. Die Auftrennung der Netzwerke erfolgt hier im Bordrechner.

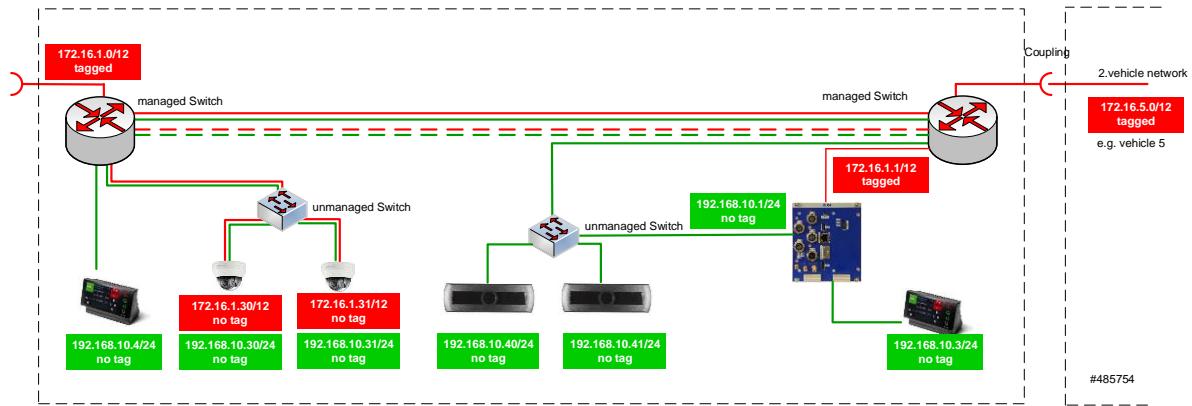
Bei dieser Lösung können einfache unmanaged Switches verwendet werden, die nicht konfiguriert werden müssen. Um ein Bedienterminal im zweiten Kopf des Wagens anzuschließen wird eine separate Leitung (insgesamt 2) über die gesamte Wagenlänge benötigt.

2.2.2 Kombination mit managed Layer 2 Switch



In diesem Beispiel wird die Auftrennung in einem separaten managed Layer 2 Switch gemacht. Nachteil ist, dass die Switches konfiguriert werden müssen, eine identische Konfiguration sollte angestrebt werden. Die Lösung kann verwendet werden, wenn der Bordrechner keinen managed Switch beinhaltet.

2.2.3 Kombination mit 2 managed Layer 3 Switch mit redundanter Ringleitung



Mittels 2 managed Layer 3 Switches lässt sich ein redundanter Backbone schaffen. Die Verkabelung im Wagen geht von diesen beiden Switches aus. Nachteil ist, dass die Switches konfiguriert werden müssen, eine identische Konfiguration sollte angestrebt werden.

Wird auf die redundante Leitung verzichtet, ist mit nur einem Kabel die Verbindung zwischen den beiden Kupplungen bzw. Köpfen des Wagens möglich.

Man beachte, dass durch die redundante Ringleitung die Verfügbarkeit des Netzwerkes bei einem Kabelbruch verbessert wird. Fällt aber eines der beiden Geräte aus, sind trotzdem große Teile des Netzwerkes betroffen. Die Kosten eines managed Switchs, der Ringleitungen unterstützt (Layer 3 Switch), sind deutlich höher als die Kosten von managed Switches ohne diese Funktion (Layer 2 Switch).

2.3 Kupplungsübergang

Für den Kupplungsübergang kann jedes Verfahren verwendet werden, welches zur IP-Übertragung verwendet werden kann. U. a. sind folgende Verfahren sind verfügbar:

- Powerline
- WLAN
- direkte Ethernet-Verbindung über spezielle Steckkontakte

Kriterien für die Auswahl des Verfahrens sind u.a. :

- Kosten der Komponenten und Installation
- Kompatibilität mit bestehenden Systemen
- Bandbreite
- Betriebssicherheit der Lösung
- Latenzzeiten

Bei der notwendigen Bandbreite ist zu beachten, dass die meiste Kommunikation lokal im Fahrzeug erfolgt und nicht über die Kupplung geführt wird.

2 IP Network Concept for Trainsets

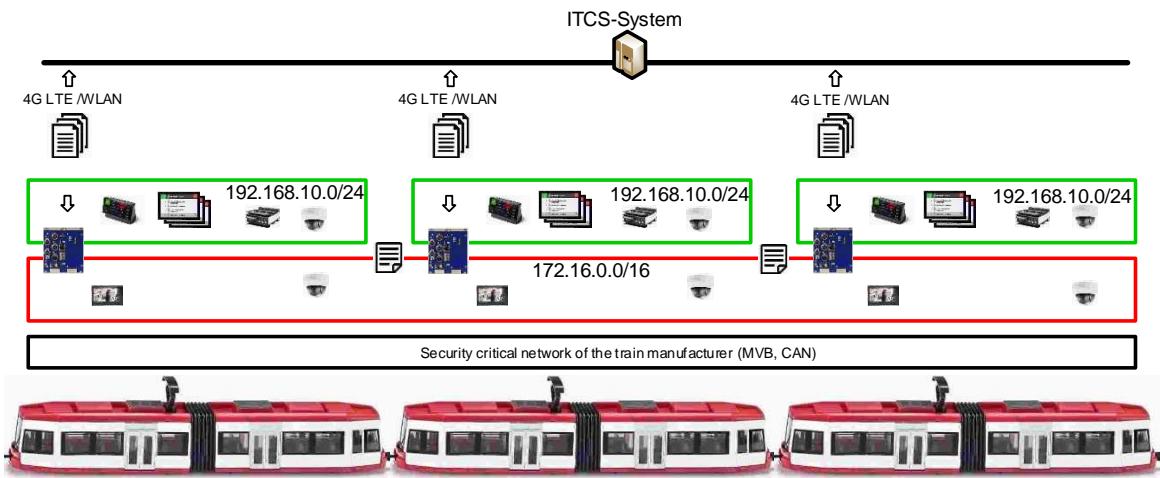
IBIS-IP devices in a coach should be able to communicate with other IBIS-IP devices in the same or if needed on another coach. To handle this we propose to use two separated IP-networks:

- A coach local IP-network (marked green in picture). In this network devices can communicate with other devices in the same coach
- A trainset IP-network (marked red in the picture)
The „trainset IP network“ is an IP network to allow for communication between coaches, but without touching any safety-critical parts of the train backbone (in the sense of IEC 61375).

A device which needs to communicate in both networks, must be part of both networks. This is possible with two network-interfaces or use of one interface with two different network addresses.

Basically we assume that an IP network of a coach manufacturer can't be used because of security reasons. Devices from the safety-critical network can be integrated via interfaces for monitoring purposes, for example.

Over the coupling an IP-communication is needed. How IP-communication takes place via the coupling is not relevant for the following considerations. This topic will be discussed separately in a later chapter (cf. 2.3).



For the sake of simplicity we assume in the following examples that the same addresses are used for the same devices in all coach IP networks, each from the address range 192.168.10.0/24.

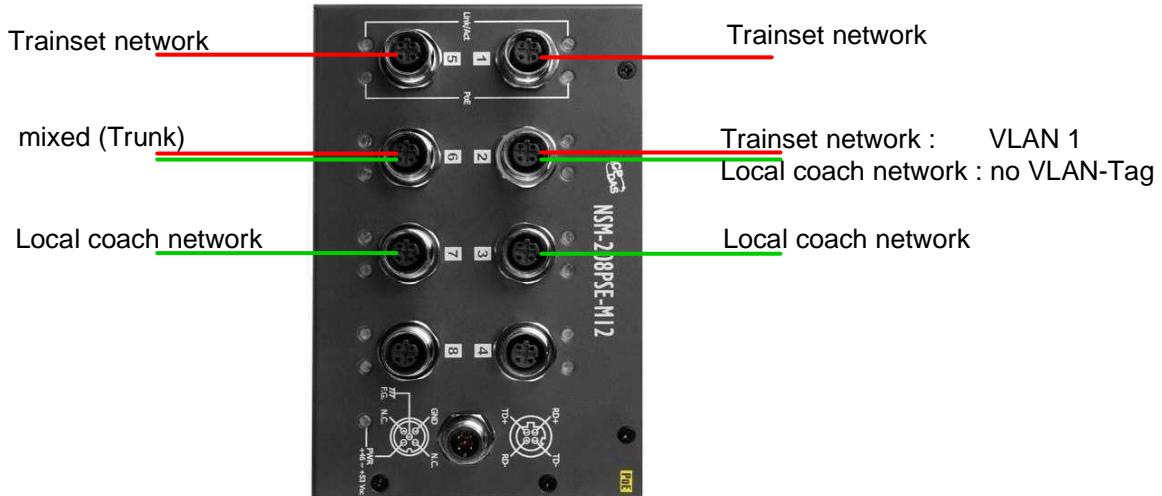
There is also a fixed assignment of a vehicle number to an address range in the train network, e.g. according to the following scheme 172.16.x.y, where x depends on the vehicle number and y is identical to the last digit of the IP address of the 192.168.10.0/24 network.

But other address ranges and also DHCP can be used in practice.

Duplicate IP addresses in non-separated networks must be avoided.

2.1 Separation of the network

If common network lines are used for both networks, the coach local network at the couplings must be restricted, otherwise address conflicts may occur. This restriction can be achieved by VLAN (according to IEEE 802.1Q). For this purpose, data packets of the trainset network are marked with VLAN 1. The data packets of the coach network do not have a VLAN tag. If required, separation can now be implemented by a managed layer 2 switch.



2.2 Network Examples

The following subchapters show 3 basic examples of how these networks can be built. It is essential that network connections to the adjacent coach are free of data packets of the coach local network.

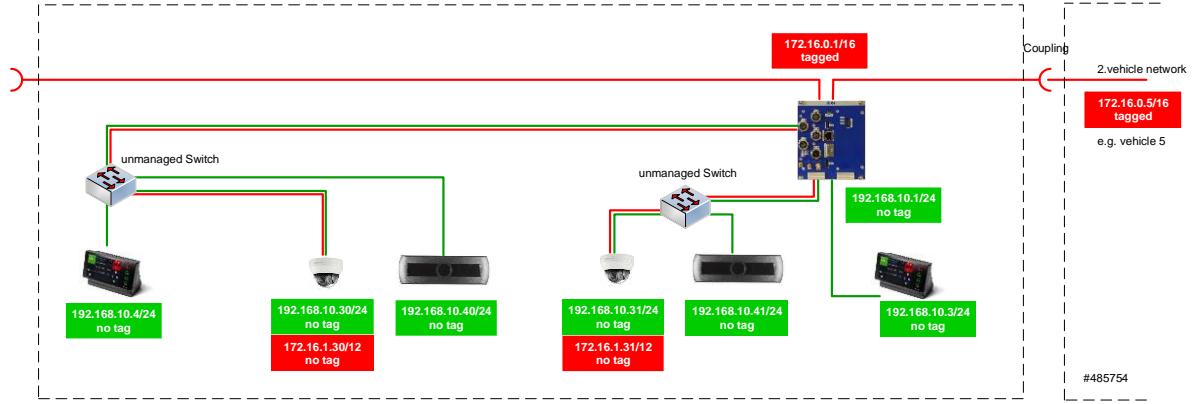
The following figures shows a coach with the couplings on the left and right.

The local coach network is shown in green and the trainset network in red. Shared network connections are marked in red and green.

Criteria for the specific architecture are:

- Required communication bandwidth
- Availability & redundancy (cable break and device failure)
- Costs
- Existing usable cables, installation opportunities
- Avoid configuration of components (replacement)
- Extensibility

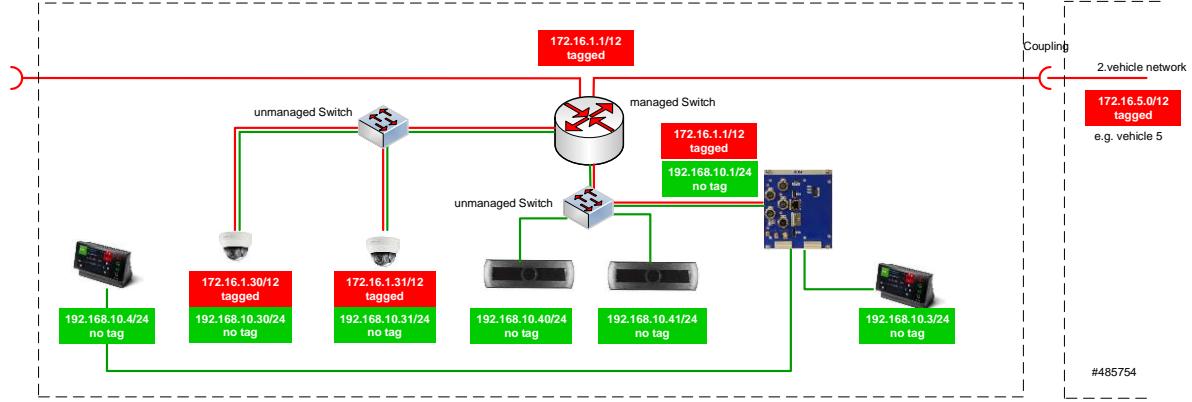
2.2.1 On-board computer with integrated managed layer 2 switch



In this example, only unmanaged switches are required for the vehicle network. The networks are separated in the on-board computer.

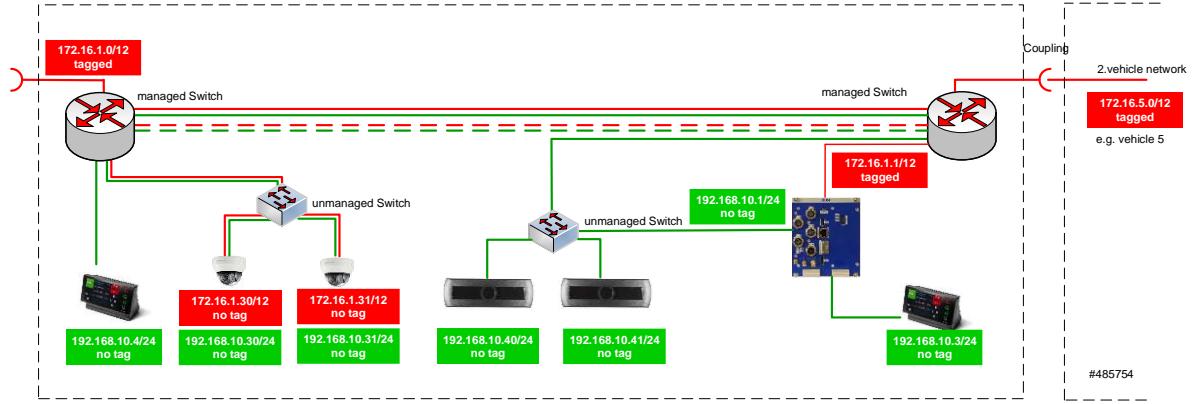
This solution allows the use of simple unmanaged switches that do not require configuration. To connect an operating terminal in the second head of the vehicle, a separate cable (2 in total) is required over the entire vehicle length.

2.2.2 Combination with managed layer 2 switch



In this example, the separation is done in a separate managed Layer 2 switch. Disadvantage is that the switches have to be configured, an identical configuration should be aimed for. The solution can be used if the on-board computer does not include a managed switch.

2.2.3 Combination with 2 managed layer 3 switches with redundant ring line



A redundant backbone can be created using 2 managed Layer 3 switches. The wiring in the vehicle comes from these two switches. Disadvantage is that the switches have to be configured, an identical configuration should be aimed for.

If the redundant line is eliminated, the connection between the two couplings or heads of the vehicle is possible with only one cable.

Note that the redundant ring line improves the availability of the network in the event of a cable break. However, if one of the two devices fails, large parts of the network are still affected. The costs of a managed switch that supports ring lines (layer 3 switch) are significantly higher than those of a managed switch without this function (layer 2 switch).

2.3 Coupling transition

Any method that can be used for IP transmission can be used for the coupling transition. The following procedures are available:

- Powerline
- WLAN
- direct Ethernet connection via special plug contacts

Criteria for the selection of the technology are, among other things:

- Cost of components and installation
- Compatibility with existing systems
- Bandwidth
- Operational reliability of the solution
- Latency

It should be noted that most communication takes place locally in the vehicle and is not routed via the coupling.

3 Einführung in das Service-Konzept

3.1 Flexibles Servicekonzept

In den folgenden Abschnitten werden Dienste beschrieben, die eine Reihe von Herausforderungen bzgl. der Kommunikation in einem Zugverband adressieren.

Die Dienste ermöglichen eine sehr flexible Nutzung. In den meisten Fällen reicht es aus, nur einen Teil der Dienste oder Dienste nur teilweise zu implementieren (d.h. nicht mit allen seinen Operationen). In Abschnitt 10 werden einige Anwendungsfälle skizziert, um zu veranschaulichen, wie unterschiedliche Anwendungsfälle durch unterschiedliche Konzepte der Serviceanwendung bedient werden können.

3.2 Funktionen und Aufgaben der Dienste

Zugverbände erfordern eine Lösung für die folgenden Herausforderungen:

- a) Alle Wagen, aus denen ein Zugverband besteht, müssen korrekt ermittelt werden, einschließlich ihrer Wagennummer, Reihenfolge und Orientierung.
Dieses Thema wird in Abschnitt 4 behandelt.
- b) Die Bordrechner in jedem Wagen müssen ihre Rolle im Zugverband zugewiesen bekommen, entweder Master, Slave oder Neutral.
Dieses Thema wird in Abschnitt 5 behandelt.
- c) Die Durchführung eines X-Y-Verkehrs und der damit verbundenen, abweichenden Fahrgastinformation in Master- und Slave-Wagen, ist ein Anwendungsfall, der durch keine der vorgenannten Operationen oder einen bestehenden IBIS-IP-Dienst abgedeckt ist.
Dieses Thema wird in Abschnitt 6 behandelt.
- d) Es muss eine sichere Methode geben, die im Zugverbands-IP-Netz verfügbaren Dienste dem Wagen zuzuordnen, zu dem sie gehören.
Dieses Thema wird in Abschnitt 7 behandelt.
- e) Die Informationen, die von bestehenden IBIS-IP-Diensten in einem Wagen-IP-Netz zur Verfügung gestellt werden, sollten auch im Zugverbands-IP-Netz verfügbar sein.
Diese Anforderung wird durch Proxy-Dienste im Zugverbands-IP-Netz abgedeckt. Diese Proxy-Dienste sind anders benannt, aber technisch identisch mit bestehenden IBIS-IP-Diensten. Ihr Umgang sollte sich an einige Regeln halten.
Einzelheiten sind in Abschnitt 8 beschrieben.
- f) Die Zusammensetzung des Zugverbands kann sich während eines Betriebstages ändern, z.B. durch eine Änderung der Fahrtrichtung oder durch das An- und Abkuppeln von Wagen. Bei jeder Änderung der Zugverbandszusammensetzung muss die gesamte IBIS-IP Dienstkommunikation im Zugverband neu initialisiert werden, so dass alle Anwendungen mit den aktuell gültigen Diensten verbunden sind. Es muss eine Möglichkeit für Anwendungen im Zugverbands-IP-Netz geben, um herauszufinden, dass sich die Zusammensetzung des Zugverbands geändert hat.
Dieses Thema wird in Abschnitt 9 behandelt.

3.3 Grundlagen der Bahnkommunikation in IBIS-IP

Die Kommunikation zwischen den verschiedenen Wagen eines Zugverbands basiert auf den folgenden Ideen:

- Die in VDV 301 beschriebenen Dienste und Operationen sollten soweit wie möglich genutzt werden. Wo immer möglich sollten vorhandene XSD-Dateien aus VDV 301 referenziert werden. Statt neue Services zu definieren, werden praxisorientierte Implementierungsrichtlinien gegeben, wie bestehende Services optimal genutzt werden können. Dies soll dazu beitragen, die Hemmschwelle für den Einsatz von IBIS-IP in Zügen durch die Wiederverwendung bestehender Dienste so gering wie möglich zu halten. Darüber hinaus soll es helfen, Änderungen an der VDV 301-Normung so gering wie möglich zu halten.
- Die Kommunikationsarchitektur ist ähnlich wie beim VDV 300 IBIS Zugbus, d.h. es gibt einen Master-Wagen (wo der Fahrer sitzt) und einen oder einige Slave-Wagen.
- Jeder Wagen ist mit einem eigenen Bordrechner ausgestattet. Der Bordrechner jedes Wagens ist sowohl an das Wagen-IP-Netz als auch an das Zugverbands-IP-Netz angeschlossen. Insbesondere Fahrgastinformationsgeräte fordern ihre Daten von der wagenspezifischen Bordrechner an. Dies ist wichtig, um eine wagenspezifische Fahrgastinformation zu ermöglichen.
- Wagenbezogene Dienste werden nur über DNS-SD im Wagen-IP-Netzwerk veröffentlicht.
- Für die Interaktion zwischen den Bordrechnern gibt es einige traktionspezifische Dienste, die über DNS-SD im Zugverbands-IP-Netz veröffentlicht werden. Diese traktionspezifischen Dienste lassen sich in zwei verschiedene Zwecke unterteilen:
 - o um einen Zugverband im Sinne von IBIS-IP aufzubauen, bestehend aus einem Master-Wagen und einem oder mehreren Slave-Wagen.
 - o um fahrbetriebsbezogene Informationen auszutauschen, die nur in Zugverbänden auftreten, wie z.B. X-Y-Verkehr.
- Geräte in einem Wagen können Zugriff auf Informationen von Geräten aus anderen Wagen erhalten. Dies wird ermöglicht über Service-Proxys, die im Zugverbands-IP-Netzwerk angeboten werden. Die Bordrechner in beiden Wagen, die ja sowohl im Wagen-IP-Netzwerk als auch im Zugverbands-IP-Netzwerk sind, können dabei eine Vermittler-Rolle einnehmen und dieses Service-Proxys anbieten.
- Es gibt keine einfache allgemeine Regel, ob ein Dienst, der von einem Gerät in einem Slave-Wagen angeboten wird, nur im Wagen-IP-Netz verfügbar sein soll ODER nur im Zugverbands-IP-Netzwerk (über den entsprechenden Proxy-Dienst) ODER in beiden Netzen. Je nach Anforderung des Kunden sind unterschiedliche Lösungen möglich. Hier sind einige Überlegungen, die im Voraus gemacht werden sollten:
 - o Sind die Informationen abhängig von der Position eines Wagens innerhalb eines Zugverbands? (Dann sollten diese Informationen nur im Wagennetz zur Verfügung gestellt werden.)
 - o Müssen die Informationen in "echter" Echtzeit vorliegen, wie bei Kameras? (Dann sollte man Transformationen und komplexes Routing vermeiden.)
 - o Wo kann ich mit einer anderen Netzwerkkonfiguration Komplexität hinzufügen oder reduzieren?
 - o Wo muss ich mit IP-Adressenkonflikten rechnen?

3 Introduction to the Service Concept

3.1 Flexible Service Concept

In the following sections there are services proposed to address a number of challenges related to communication in a trainset.

These services allow for a very flexible usage. In most of the cases it is sufficient to implement only a part of the services or to implement services partly (i.e. not with all its operations). In section 10 there are some use cases outlined to illustrate how different use cases can be served by different concepts of service application.

3.2 Functions and Tasks of the Services

Trainsets require a solution for the following challenges:

- a) All coaches a trainset consists of have to be determined correctly, including their coach number, sequence and orientation.
This topic is addressed in section 4.
- b) On-board units in each coach must have their role in the trainset assigned, either, master, slave or neutral. This topic is addressed in section 5.
- c) The handling of an X-Y-traffic and the diverging passenger information in master and slave coaches, is a use case which is not covered with none of the before mentioned operations or any existing IBIS-IP operation. This topic is covered in section 6.
- d) There must be a safe way to assign services which are available in the trainset IP network to the coach they belong to.
This topic is addressed in section 7.
- e) The information provided by existing IBIS-IP services in a coach IP network should also be available in the trainset IP network.
This requirement is covered by proxy services in the trainset IP network. These proxy services are differently named but are technically identical to existing IBIS-IP services. Their handling should obey some rules. Details are described section 8.
- f) The trainset composition might change during an operation day, e.g. due to a change of the driving direction or due to coupling/decoupling of coaches. Whenever the trainset composition changes all IBIS-IP service communication in the trainset has to be re-initialised, so that all applications are connected to the recently valid services. There must be a way for applications in the trainset IP network to find out that the trainset composition changed. This topic is covered in section 9.

3.3 Basic Principles for Trainset Communication in IBIS-IP

Communication between different coaches in a trainset is based on the following ideas:

- Services and operations described in VDV 301 shall be used as far as possible. Wherever possible existing XSD files from VDV 301 should be referenced. Instead of defining new

services, practice-oriented implementations guidelines are given how to make best use of existing services. This shall help to keep the thresholds for the use of IBIS-IP in trainsets as little as possible by the re-use of existing services. Furthermore it shall help to keep changes to the VDV 301 standardization as little as possible.

- The communication architecture is similar to the VDV 300 IBIS Zugbus, i.e. there is a single master coach (where the driver sits) and some slave coaches.
- Each coach is equipped with its own on-board unit. The on-board unit of each coach is connected to both, the coach IP network and the trainset IP network. In particular passenger information devices request their data from the coach specific on-board unit. This is important to allow for a coach specific passenger information.
- Coach related services are published via DNS-SD in the coach network only.
- There are some trainset specific services published via DNS-SD in the trainset IP network for the interaction between the on-board units. These trainset specific services can be divided in two different purposes:
 - o To setup a trainset, consisting of one master coach and a one or more slave coaches.
 - o To exchange operation related information which only occur in trainset situations, like X-Y-traffic
- Devices in one coach can get access to information from devices in other coaches. This is made possible by service proxies that are offered in the trainset IP network. The on-board units in both coaches, which are both in the coach IP network and in the trainset IP network, can play an intermediary role and offer these service proxies.
- There is no simple general rule whether a service provided by a device in a slave coach should be available in the coach network only OR in the trainset network only (by means of the corresponding proxy service) OR in both networks. Depending on the requirements of a customer, different solutions are possible. Here are some considerations, that should be made in advance:
 - o Are the information dependent on the position of a coach within a trainset?
(Then these information should be provided in the coach network only.)
 - o Does the information have to be in “true” realtime, like for cameras?
(Then one should avoid transformations and complex routing.)
 - o Where do I add or reduce complexity with a different network setup?
 - o Where do I have to expect IP address conflicts?

4 TrainSetInformationService

4.1 Purpose

Coach manufacturer know best how to determine which coaches participate in a trainset. For each coach of a trainset there is an information about the coach number, the sequence of the coaches, and the orientation.

This information is needed by on-board units but potentially also by other devices and applications, such as CCTV systems which store operational information (e.g. line number, destination) within the videos.

An HTTP service in the meaning of IBIS-IP, called TrainSetInformationService, is proposed here. It provides an operation to request the composition of the trainset. As usual in IBIS-IP there is not only a “Get” operation but there are also a “Subscribe” operation and an “Unsubscribe” operation proposed.

Please note:

There might be other solutions, in particular in retro-fit situations, which serve the same purpose: to determine the coach number, the sequence of the coaches, and the orientation of the coaches in a train set correctly. There might be existing proprietary interfaces or even the old fashioned VDV 300/IBIS Zugbus could be used for instance.

Ultimately, all that matters is that the on-board unit in the master vehicle has the information about coach number, sequence and orientation.

4.2 Where is the service provided and by whom?

The **TrainSetInformationService** is provided in the coach IP network of each coach.

The on-board computers (in the following simply called “train computers”) of the coach manufacturer can provide this information. In this case, all devices and applications in the coach IP network can get the information about the trainset composition from this service, i.e. both, the on-board unit and peripheral devices like cameras or passenger information displays.

Alternatively the service can be provided by the OBU. In that case the OBU would get the trainset composition information from another source (see the remark in the box in 4.1). But the OBU would provide the service for other applications or devices in the coach IP network.

4.3 Service Publication via DNS-SD

The publication of the service follows the rules described in VDV 301-2-0, section “Publication via DNS-SD”. The service name of the TrainSetInformationService is

TrainSetInformationService.

The SRV record of the TrainSetInformationService looks as follows:

<_Service._Proto.Name TTL Class SRV Priority Weight Port Target>

An example would look as follows

```
<TrainSetInformationService._ibisip_http.local. 3600 IN SRV 10  
0 389 OnboardUnit_1.local.>
```

The TXT record does not have to provide information on the attributes **multicast** and **sntp-server**.

4.4 Basic Protocols

The **TrainSetInformationService** is an HTTP based service in the sense of IBIS-IP.

4.5 Operations of the TrainSetInformationService

Operation	Request/ Response	Data type used, data structure
GetTrainSetComposition	Request	
	Response	TrainSetInformationService. GetTrainSetCompositionResponse
SubscribeTrainSetComposition	Request	
	Response	
UnsubscribeTrainSetComposition	Request	
	Response	

4.5.1 GetTrainSetComposition

4.5.1.1 Request

There is no structure transmitted with the **GetTrainSetComposition** request.

4.5.1.2 Response

The response consists of a sequence of structures of type SingleCoachInATrainSet each containing the following information about a coach:

- CoachType
- CoachNumber
- FrontCabin (either A/B or 1/2 or None)
- RearCabin (either A/B or 1/2 or None)
- CoachPositionInTrainSet (master = 1, first slave = 2 and so forth)
- CoupledSide
- CoachState

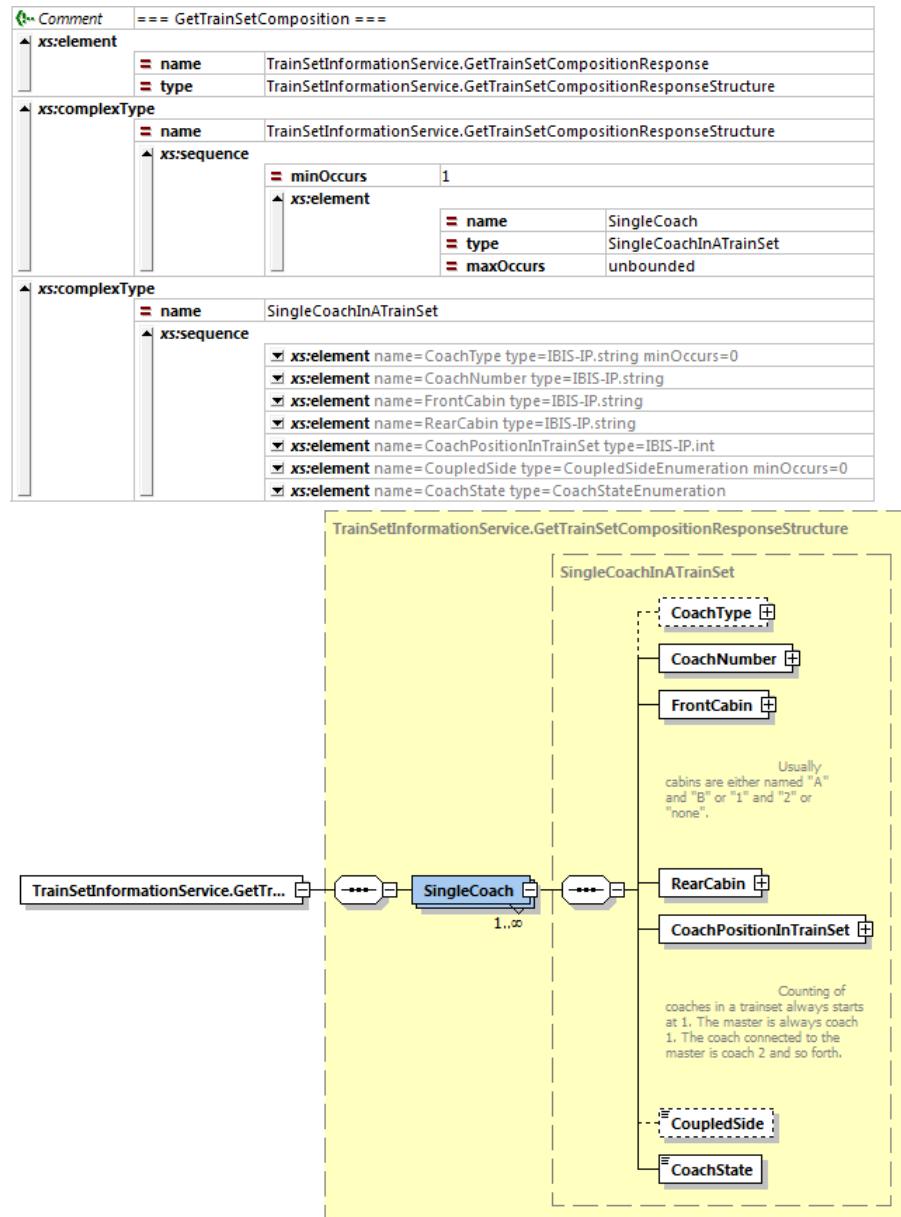
The CoupledSide is an enumeration and can have the following values:

= name	CoupledSideEnumeration
▲ xs:restriction	
= base	xs:string
☒ xs:enumeration	value=None
☒ xs:enumeration	value=A
☒ xs:enumeration	value=B
☒ xs:enumeration	value=A+B

The CoachState is an enumeration and can have the following values:

= name	CoachStateEnumeration
▲ xs:restriction	
= base	xs:string
☒ xs:enumeration	value=Master
☒ xs:enumeration	value=Slave
☒ xs:enumeration	value=Neutral
☒ xs:enumeration	value=Unknown

The response to a **GetTrainSetComposition** request returns a sequence of structures of type SingleCoachInATrainSet, one per coach.



4.5.2 SubscribeTrainSetComposition

For the subscription the data structures **SubscribeRequest** and **SubscribeResponse** as described in VDV 301-2-1 are used.

4.5.3 UnsubscribeTrainSetComposition

To terminate the subscription the data structures **UnsubscribeRequest** and **UnsubscribeResponse** as described in VDV 301-2-1 are used.

5 TrainSetManagementService

5.1 Purpose

The **TrainSetManagementService** serves one purpose: to inform all OBUs in a trainset about their activation state.

In the following we assume that a coach can drive forward and backward and thus have a driver cabin at either end of the coach. In a coach only one cabin can be “active” (operated by a driver), the other one is “passive”. A state with two active cabins is prohibited.

Depending on the activation state and the coach there are three possible states.

If one cabin is in an active state:

1st: the respective active coach/on-board unit is in **master state**

2nd: all other coaches/ on-board units are in **slave state**.

If no cabin is active, i.e. all cabins are passive:

3rd: all coaches/on-board units are in **neutral state**.

An on-board unit alone cannot distinguish between a slave state and a neutral state due to the “cabin passive” state. It has to be informed by the OBU that either changes from master state to neutral state (was the master) or vice versa (becomes the master).

The **TrainSetManagementService** is designed to communicate these information.

There are two requests provides by the **TrainSetManagementService** for that purpose:

- **SetSlaveMode** (cf. 5.5.1):
the on-board unit of the master coach knows its own master state due to the “active state” of the driver cabin. All other coaches/on-board units have to be informed that they are now in slave mode. This is done by the SetSlaveMode request.
- **SetNeutralMode** (cf. 5.5.2):
If an active cabin in the master coach is switched to passive the respective OBU sends actively a request to all the OBUs in the trainset IP network. This is done by the SetNeutralMode request to inform the other OBUs that a master is no longer available and that all OBUs have the same neutral state.

Please note:

It is recommended to use the TrainSetManagementService service to communicate the changes of the master/slave/neutral state although this information is in principal also available for each OBU through the TrainSetInformationService provided by a train computer. The reasons are:

1st: There is not always a train computer available which provides this information.

2nd: The moment of the SetSlaveMode/SetNeutralMode trigger can be determined by the master OBU and thus may take some other operational aspects (e.g. the availability of certain data or services) into account.

In addition the TrainsetManagementService provides an operation which allows to request the trainset composition from the trainset IP network.

5.2 Where is the service provided and by whom?

The **TrainSetManagementService** is provided in the trainset IP network only.

The **TrainSetManagementService** has to be provided by each OBU which participates in the trainset IP network.

5.3 Service Publication via DNS-SD

The publication of the service follows the rules described in VDV 301-2-0, section “Publication via DNS-SD”. The service name of the TrainSetManagementService is

TrainSetManagementService.

The SRV record of the TrainSetManagementService looks as follows:

<_Service._Proto.Name TTL Class SRV Priority Weight Port Target>

An example would look as follows

```
<TrainSetManagementService._ibisip_http.local. 3600 IN SRV 10  
0 389 OnboardUnit_1.local.>
```

The TXT record does not have to provide information on the attributes **multicast** and **sntp-server**.

5.4 Basic Protocols

The **TrainSetManagementService** is an HTTP based service in the sense of IBIS-IP.

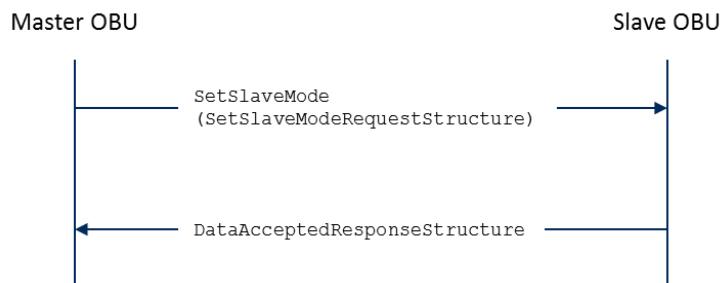
5.5 Operations of the TrainSetManagementService

Operation	Request/ Response	Data type used, data structure
SetSlaveMode	Request	SetSlaveModeRequestStructure
	Response	DataAcceptedResponseStructure
SetNeutralMode	Request	
	Response	DataAcceptedResponseStructure
GetTrainSetComposition	Request	
	Response	TrainSetInformationService. GetTrainSetCompositionResponse
SubscribeTrainSetComposition	Request	
	Response	
UnsubscribeTrainSetComposition	Request	
	Response	

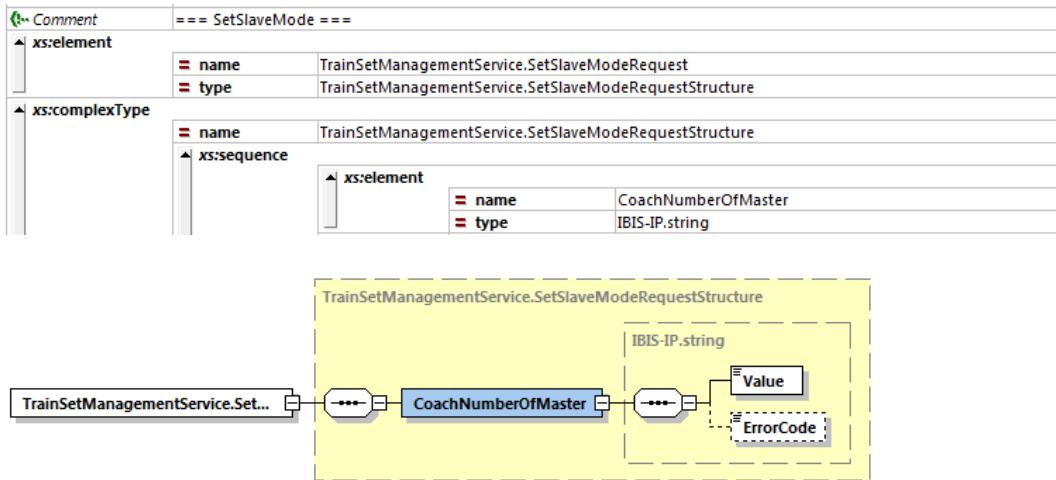
5.5.1 Operation SetSlaveMode

The operation **SetSlaveMode** is provided by the OBU of the slave coach. It is called by the OBU of the master coach, to inform the other OBUs that they are in slave mode. Based on the response of the slave coach OBUs the master OBU can make the assignment of “slave-coach-number-to-IP-address”. This is important to enable the correct addressing when data of a certain coach are needed.

The request is also used to transmit the number of the master coach to the slave OBU. Thereby the slave OBU can make the assignment of “master-coach-number-to-IP-address” too.

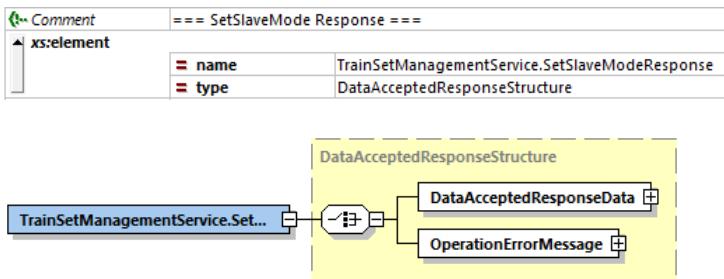


5.5.1.1 Request



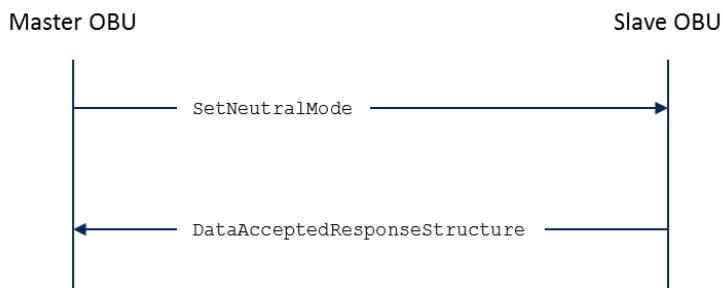
5.5.1.2 Response

For the acknowledgement of the request the **DataAcceptedResponseStructure** as described in VDV 301-2-1 is used.



5.5.2 Operation SetNeutralMode

The Operation **SetNeutralMode** is used, to inform OBUs in slave coaches, that a master OBU is no longer available. It is called by the master OBU when the OBU recognizes, that a coach is no longer the master coach. Once all slave OBUs are set in neutral mode the master OBU changes in neutral mode itself.

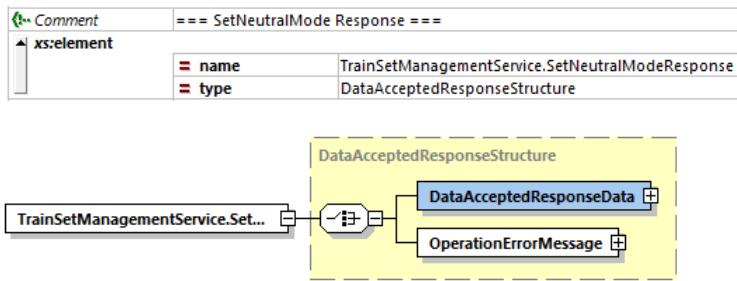


5.5.2.1 Request

With the request **SetNeutralMode** is no data transmitted.

5.5.2.2 Response

For the acknowledgement of the request the **DataAcceptedResponseStructure** is used.



5.5.3 GetTrainSetComposition

The operation **GetTrainSetComposition** is provided by all OBUs in the trainset network. It shall help to make clear which coach is in which state, in particular for devices in the trainset network which are not directly informed when the trainset composition changes during the operation of a trainset (cf. 9). The data structure is identical to the one provided by the **TrainSetInformationService**, but provided by another device.

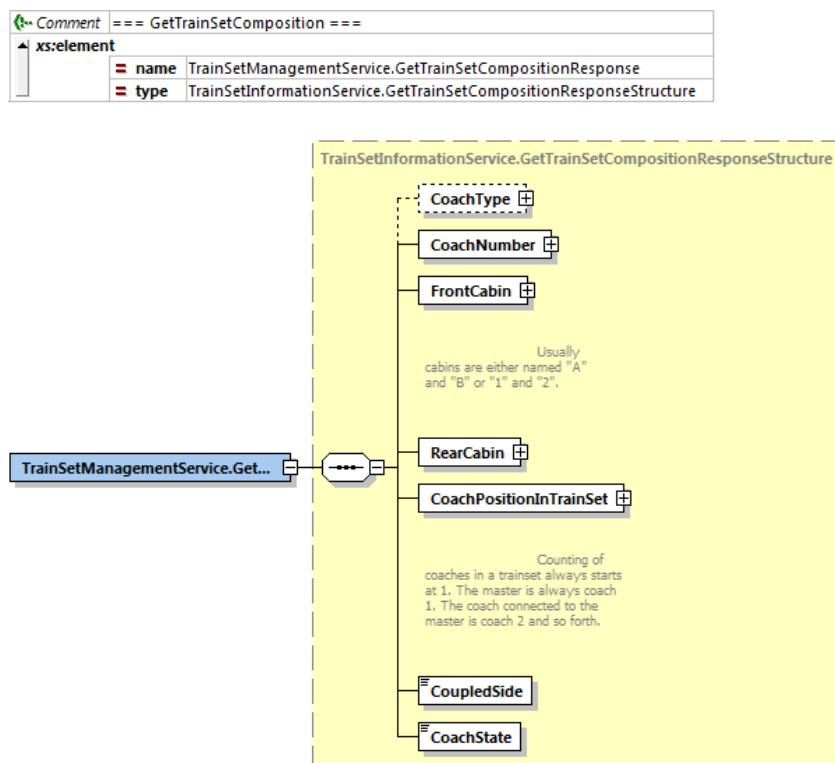
Please note that OBUs in a slave coach which provide the GetTrainSetComposition operation have to ensure that they provide the information on the latest state of the trainset composition. This can be done for instance by requesting the GetTrainSetComposition operation provided by the OBU of the master coach.

5.5.3.1 Request

With the request **GetTrainSetComposition** there is no data transmitted.

5.5.3.2 Response

For the acknowledgement of the request the **GetTrainSetCompositionResponseStructure** of the **TrainSetInformationService** is used.



5.5.4 SubscribeTrainSetComposition

For the subscription the data structures **SubscribeRequest** and **SubscribeResponse** as described in VDV 301-2-1 are used.

5.5.5 UnsubscribeTrainSetComposition

To terminate the subscription the data structures **UnsubscribeRequest** and **UnsubscribeResponse** as described in VDV 301-2-1 are used.

6 TrainSetDataService

6.1 Purpose

The **TrainSetDataService** serves one basic purpose: in case of an X-Y-traffic a master OBU provides information on the diverging travel path for the OBUs in the slave coaches.

In a trainset there might occur operational situations that are specific to trainsets. An important example is the so-called X-Y-traffic (also referred to as portion working). In an X-Y-traffic the trainset is established only temporary for a certain section of the travel path. The coaches of such a trainset do not have the same travel path (cf. Figure 1).

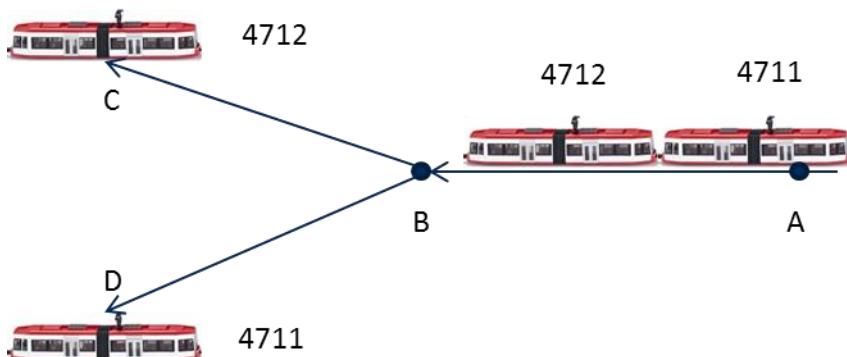


Figure 1: Example of an X-Y-traffic.

In the example the trainset consists of the coaches 4711 and 4712. Both coaches share the same travel path between A and B. At stop B both coaches are separated. Coach 4711 continues its way to destination D. Coach 4712 continues its way to destination C.

Between A and B both coaches make up a trainset in which 4712 is the master coach and 4711 is the slave coach. Nonetheless, passenger information in coach 4712 has to inform about the trip to destination C, while passenger information in coach 4711 has to inform about the trip to destination D.

It can be assumed that a trip selection is done on the master coach only. Dispatching actions of a fleet management centre are communicated to the master coach only. In both cases the slave coach has to be informed by means of specific operations.

The operations and data structures mentioned below are designed to handle this kind of situations. There are two different operations proposed to allow OBU manufacturers to choose the one that is easier to implement to meet the customers goals:

- **RetrieveTripRef** and
- **RetrieveTripInformation**.

Though the proposed solution is suited for the more complex X-Y-traffics it is also suited for the easier cases where the coaches are not separated during a trip.

6.2 Where is the service provided and by whom?

The **TrainSetDataService** is provided in the trainset IP network.

The **TrainSetDataService** is provided by the OBU of the master vehicle.

6.3 Service Publication via DNS-SD

The publication of the service follows the rules described in VDV 301-2-0, section “Publication via DNS-SD”. The service name of the TrainSetDataService is

TrainSetDataService.

The SRV record of the TrainSetDataService looks as follows:

<_Service._Proto.Name TTL Class SRV Priority Weight Port Target>

An example would look as follows

```
<TrainSetDataService._ibisip_http.local. 3600 IN SRV 10  
0 389 OnboardUnit_1.local.>
```

The TXT record does not have to provide information on the attributes **multicast** and **sntp-server**.

6.4 Basic Protocols

The **TrainSetDataService** is an HTTP based service in the sense of IBIS-IP.

6.5 Operations of the TrainSetDataService

Information concerning operational trips is dependent on the position of the coach within the trainset. Therefore the “Retrieve” operations of IBIS-IP are used. They allow to transmit the coach number as a parameter in the request.

In addition to the existing IBIS-IP concept, it is proposed to make use of the subscription mechanisms of IBIS-IP also for the parameter related Retrieve operations. Specifics of this definition are explained in sections 6.5.1 and 6.5.2.

Thereby on-board units of slave coaches can be informed immediately by a subscription.

Thus we have a triple of operations for each requested dataset: RetrieveX, SubscribeX and UnsubscribeX.

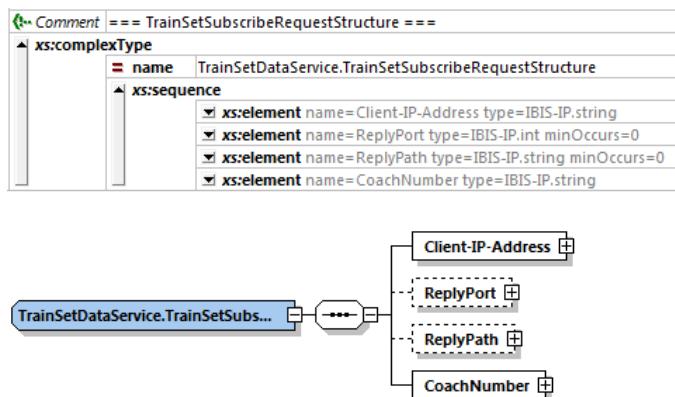
Operation	Request/ Response	Data type used, data structure
RetrieveTripRef	Request	TrainSetDataService.RetrieveTripRefRequestStructure
	Response	TrainSetDataService.RetrieveTripRefResponseStructure
SubscribeTripRef	Request	TrainSetDataService.TrainSetSubscribeRequestStructure
	Response	SubscribeResponseStructure
UnsubscribeTripRef	Request	TrainSetDataService.RetrieveTripRefRequestStructure
	Response	UnsubscribeResponseStructure

RetrieveTripInformation	Request	TrainSetDataService.RetrieveTripInformationRequestStructure
	Response	TrainSetDataService.RetrieveTripInformationResponseStructure
SubscribeTripInformation	Request	TrainSetDataService.TrainSetSubscribeRequestStructure
	Response	SubscribeResponseStructure
UnsubscribeTripInformation	Request	TrainSetDataService.RetrieveTripInformationRequestStructure
	Response	UnsubscribeResponseStructure

6.5.1 Specific TrainSetSubscribeRequestStructure

To meet the special requirements of a subscription which is dependent on the coach number a specific subscription request is defined here. In addition to the “normal” subscription parameters (i.e. Client-IP-Address, ReplyPort, ReplyPath) it also takes the CoachNumber as a request parameter into account.

This TrainSetSubscribeRequestStructure is defined only in the context of the TrainSetDataService.

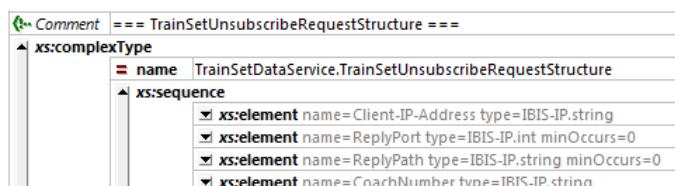


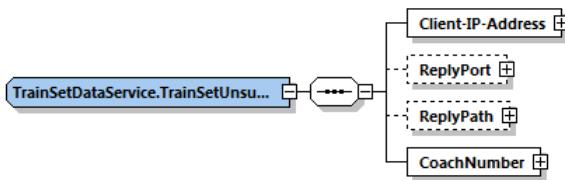
6.5.2 Specific TrainSetUnsubscribeRequestStructure

To meet the special requirements to be able to end a subscription for a specific coach number only a specific unsubscribe request is defined here. In addition to the “normal” unsubscribe request parameters (i.e. Client-IP-Address, ReplyPort, ReplyPath) it also takes the CoachNumber as a request parameter into account.

This TrainSetUnsubscribeRequestStructure is defined only in the context of the TrainSetDataService.

Though it is technically identical to the TrainSetSubscribeRequestStructure it is used here for the sake of consistency with the general IBIS-IP subscribe/unsubscribe behaviour.

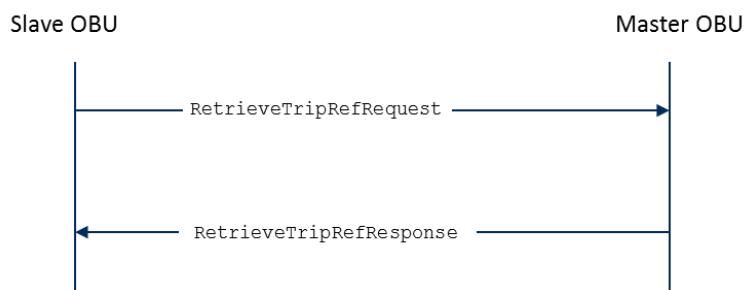




6.5.3 Operation RetrieveTripRef

The operation **RetrieveTripRef** is provided by the master OBU in a trainset. The operation **RetrieveTripRef** is used by OBUs in slave coaches to get a trip reference for the trip that has to be performed by the slave coach. Thereby the OBU in the slave coach can provide a passenger information which is widely independent from the passenger information of the master coach, in particular with respect to the travel path and destination.

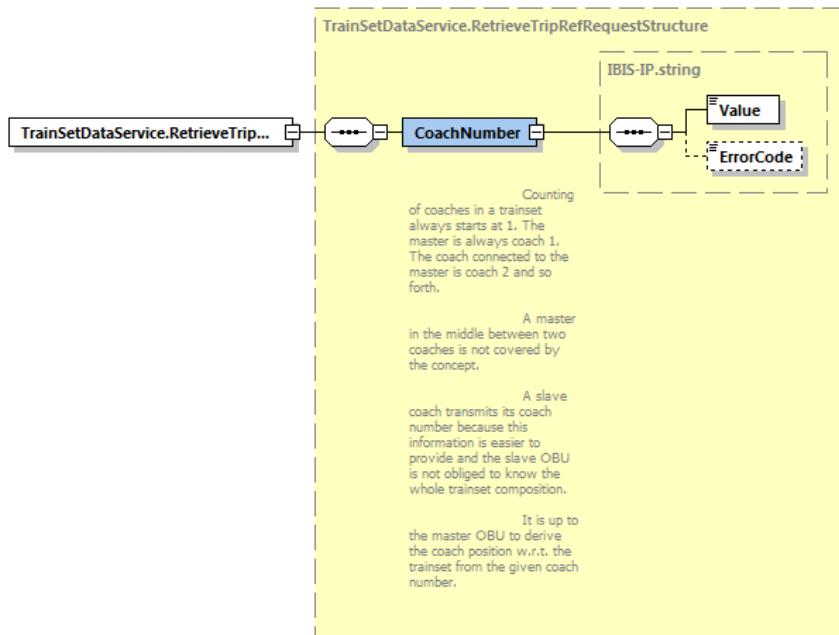
In order to work like that the data of the master coach has to contain not only the data of the trip to be performed by the master coach. In addition it has to contain at least the references to the trips which are performed by the slave coaches and information which of the slave coaches has to perform which trip (master coach is heading to X, first slave coach is heading to Y, second slave coach is heading to Z).



6.5.3.1 Request

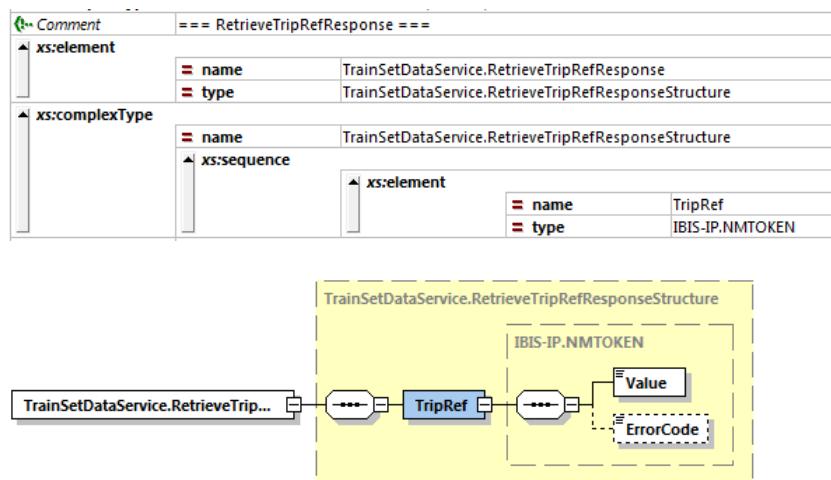
In a trainset of three (or more) coaches the travel paths and destinations of the second and third coach might be different as well. That is why the slave coaches transmit their **CoachNumber** within the request. From the slave's **CoachNumber** the on-board unit in the master coach can determine the position of the coach with respect to the trainset in order to response with the correct **TripRef**.

Comment	==== RetrieveTripRefRequest ====
xs:element	
= name	TrainSetDataService.RetrieveTripRefRequest
= type	TrainSetDataService.RetrieveTripRefRequestStructure
xs:complexType	
= name	TrainSetDataService.RetrieveTripRefRequestStructure
xs:sequence	
Comment	counting of coaches in a trainset always starts at 1. The master is always coach 1. The coach connected to the master is coach 2 and so forth
Comment	a master in the middle between two coaches is not covered by the concept
xs:element	
= name	CoachNumber
= type	IBIS-IP.string
xs:annotation	

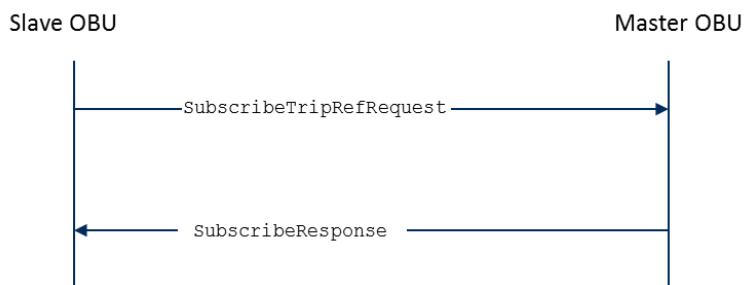


6.5.3.2 Response

The response returns the **TripRef** of the trip which has to be performed by the slave coach.



6.5.4 Operation SubscribeTripRef



6.5.4.1 Request

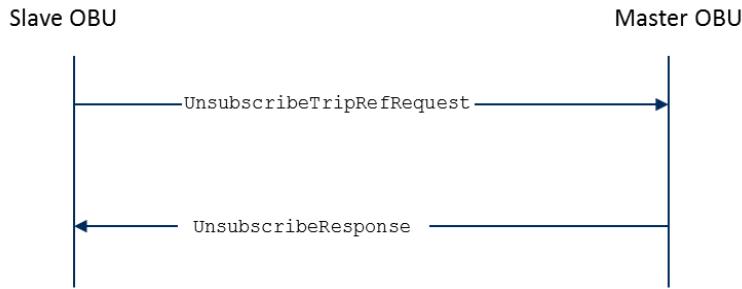
The **SubscribeTripRef** request makes use of the **TrainSetSubscribeRequestStructure** as defined in section 6.5.1.

6.5.4.2 Response

The data structure **SubscribeResponseStructure** is used (cf. VDV 301-2-1).

Please note: As always in IBIS-IP subscriptions: Once the subscription is initialised, the service provider sends event based updates via the **RetrieveTripRefResponseStructure**.

6.5.5 Operation UnsubscribeTripRef



6.5.5.1 Request

The **UnsubscribeTripRef** request makes use of the **TrainSetUnsubscribeRequestStructure** as described in section 6.5.1.

6.5.5.2 Response

The data structure **UnsubscribeResponseStructure** is used (cf. VDV 301-2-1).

6.5.6 Operation RetrieveTripInformation

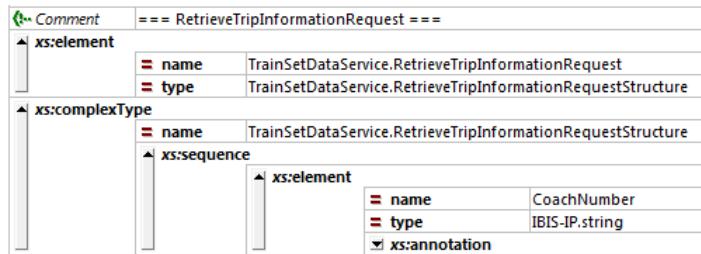
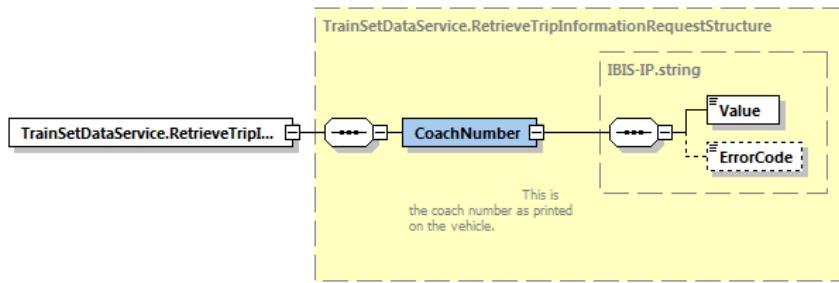
In contrast to the operation **RetrieveTripRef** (cf. 6.5.1) the use case here assumes that the OBU in the slave coach has no separate data supply. That is why the complete trip information (including the diverging travel path/destination of the slave coach) for the OBUs in the slave coaches is provided by the OBU of the master coach.

It is likely that dispatching actions received by the OBU in the master coach are easier to communicate to the OBUs in the slave coaches by means of this mechanism.



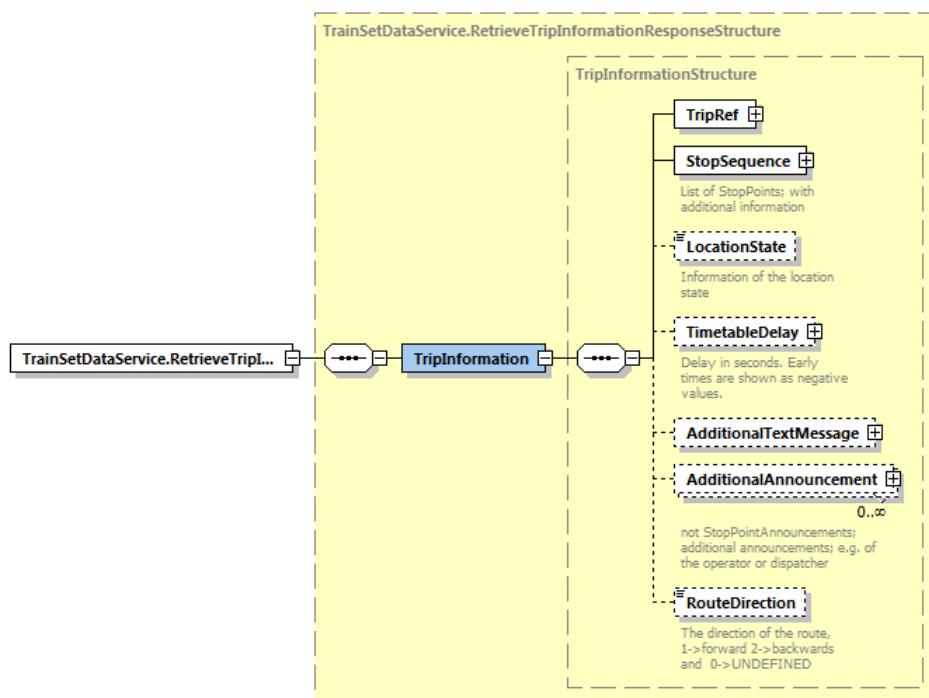
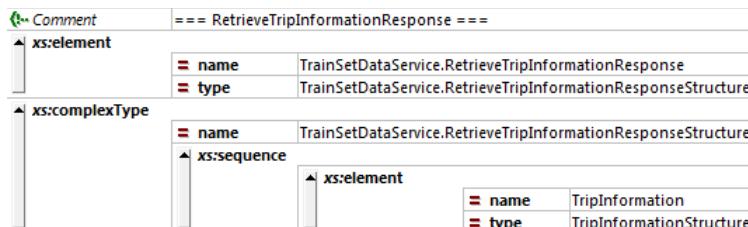
6.5.6.1 Request

As in the case of the **RetrieveTripRef** operation (cf. 6.5.3.1) the OBU in the slave coach provides the slave coach number within the request.

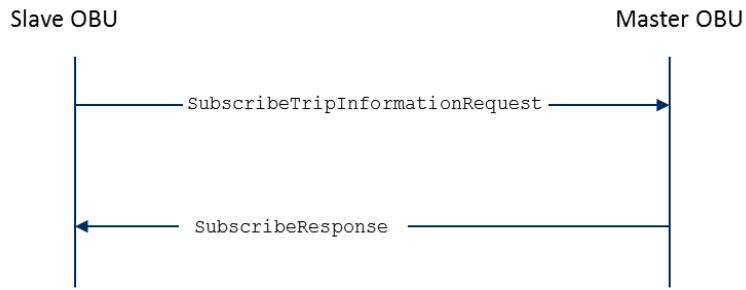


6.5.6.2 Response

The response makes use of the **TripInformationStructure** as described in VDV 301-2-1 to transmit the complete trip information.



6.5.7 Operation SubscribeTripInformation



6.5.7.1 Request

The **SubscribeTripInformation** request makes use of the **TrainSetSubscribeRequestStructure** as defined in section 6.5.1.

6.5.7.2 Response

The data structure **SubscribeResponseStructure** is used (cf. VDV 301-2-1).

Please note: As always in IBIS-IP subscriptions: Once the subscription is initialised, the service provider sends event based updates via the **RetrieveTripRefResponseStructure**.

6.5.8 Operation UnsubscribeTripInformation



6.5.8.1 Request

The **UnsubscribeTripInformation** request makes use of the **TrainSetUnsubscribeRequestStructure** as described in section 6.5.1.

6.5.8.2 Response

The data structure **UnsubscribeResponseStructure** is used (cf. VDV 301-2-1).

7 Service-to-Coach Assignment

It is important to know, which service in the trainset IP network belongs to which coach. To avoid any impact on the XSD definition of already existing IBIS-IP services, the service-to-coach assignment is done by means of the service publication in DNS-SD.

To allow for an easy and safe parsing of the service names the service naming in the trainset IP network has to obey certain rules.

According to the standard IBIS-IP service publication (for services in a vehicle IP network or coach IP network), the SRV record of a service has to follow the scheme

`<_Service._Proto.Name TTL Class SRV Priority Weight Port Target>`

In addition for a service in the trainset IP network the `_Service` field **MUST** be filled according to the following scheme

`_Service = ServiceName_CoachX_`

For example

`TrainSetManagementService_Coach4711_`
`CustomerInformationServiceProxy_Coach2345_`

would be compliant with the naming scheme for services in the trainset IP network.

It is important that there is a “Coach” in front of the coach number, to allow for a safe identification of the beginning of the coach number.

It is important that there is an underscore “`_`” after the coach number to allow for a safe identification of the end of the coach number.

Furthermore we RECOMMEND to extend the `_Service` field by a suffix with information about the device which provides the service, containing

- the manufacturer of the device,
- the device name and
- the ID of the device

according to the scheme

`_Manufacturer_DeviceName_DeviceID .`

This is to avoid trouble with naming ambiguities which were encountered with some DNS-SD libraries.

For example

If the TrainSetManagementService is provided on coach 4711 by a device from manufacturer = “ABC”, device name = “Superbox”, device ID = “2” the service name would be

`TrainSetManagementService_Coach4711_ABC_Superbox_2`

8 Implementation Guide Line for existing IBIS-IP services

As mentioned above, existing IBIS-IP services shall be used wherever this is possible.

Services which are used in a coach IP network can be made available to the users outside the coach IP network via proxy services. This requires of course devices which have the capability to be part of different IP networks at the same time.

In the following there are some use cases outlined to illustrate the handling in a tangible situation.

A distinction has to be made between those services which provide their data via a request/response/subscription (“HTTP services”, cf. 8.2) and those which publish their data via a fire and forget mechanism in a multicast group (“UDP services”, cf. 8.3).

8.1 General Approach for the Handling of Proxy Services

Proxy services are characterized by the following features:

- They are published in the trainset IP network
- Proxy services are published in the trainset IP network only when a respective service in the coach IP network exists.
- On the other hand there is no constraint that a proxy service has to be published for each service in a coach IP network. It is up to the project to decide, which proxy services are needed.
- Proxy services have a name that equals the service name, except for the extension “**Proxy_CoachX_**”, e.g. “**CustomerInformationServiceProxy_Coach4711_**” instead of “**CustomerInformationService**” only (cf. section 10.2). Thereby it is guaranteed that it is always clear which service belongs to which coach.
- For other devices than OBUs it is up to the project to find a solution on how to obtain the coach number information from somewhere inside the coach IP network (e.g. by means of the **CustomerInformationService/GetVehicleData** if applicable).
- The publication of the coach number in the service name avoids impacts on the data structures or operations of the services.
- The proxy services in the trainset IP network provide the same operations and functionalities as the “normal” services in the coach network.

8.2 Specifics for HTTP proxy services

HTTP type proxy services are characterized by the following specific features:

- In general a request received by a proxy service in the trainset IP network should be forwarded to the corresponding service in the coach IP network. The response received from the service in the coach IP network should be forwarded as a response of the proxy service.
- The forwarding of requests and responses between the trainset IP network and the coach IP network might be too time consuming. To avoid possibly long latencies due to

the realtime-forwarding of requests, it could be advisable to acquire the information in advance to respond faster. For instance this can be obtained by cyclical requests within the coach IP network.

It is up to the project, to decide which solution is more appropriate.

8.3 Specifics for UDP proxy services

UDP type proxy services are characterized by the following specific features:

- As the UDP services provide their data to a multicast address that is independent from the IP address of the devices or the coach number, data can in principle be sent to that multicast address from different sources.
- Only the devices of the master coach are allowed to publish data to a multicast address which is read from by different slave coaches. E.g. the GNSS location may only be provided by the master coach and the logical position may only be provided by the OBU of the master coach.
- In particular the service publication remains valid when the role of master and slave is switched from one coach to another as long as they all use the same multicast address.

9 Re-Initialisation of valid Services

The master/slave role of an OBU might change several times during the daily operation. That is why applications have to re-initialise their communication with IBIS-IP services in the trainset IP network accordingly.

It seems in general appropriate to re-initialise the service communication only when an OBU switches from the neutral state in the slave state.

At least the subscriptions to

TrainSetManagementService.SubscribeTrainSetComposition should not be reset, as this is the “anchor” for other devices to be informed about changes of the trainset composition.

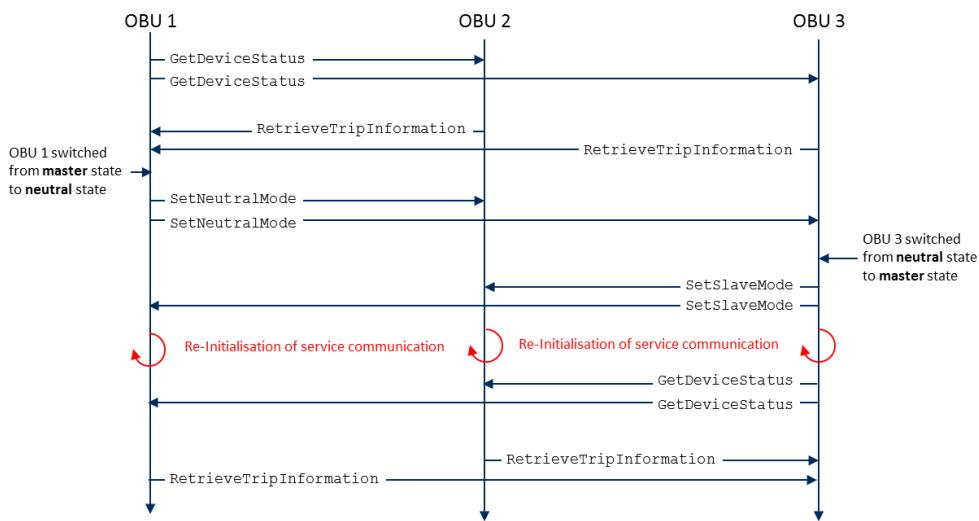
9.1 Re-initialisation of Service Communication when a Trainset changes the Driving Direction

9.1.1 Use case

In a trainset with three coaches the master/slave/neutral roles change whenever the driving direction changes. The service communication has to be re-initialised so that all service providers and service users are linked correctly by means of IBIS-IP services.

9.1.2 Possible implementation for OBUs

In a trainset with three coaches the coach in the middle is always either slave or neutral. When the trainset changes the driving direction another coach becomes the master and all service communication has to be re-initialised. This is illustrated in the figure below.



In the beginning OBU 1 is in master state. OBU 2 and 3 are in slave state. As a master OBU 1 requests the device status of OBU 2 and OBU 3 regularly (or uses descriptions), e.g. by means of the **DeviceManagementServiceProxy_Coach2/3.GetDeviceStatus**. On the other hand OBU 2 and OBU 3 retrieve the trip information from the master vehicle to provide up-to-date passenger information in their respective coach.

Then OBU 1 is switched from master state to neutral state by an external trigger.

When OBU 1 is switched to neutral state it sends a **SetNeutralMode** request to the slave OBUs 2 and 3.

Then there is a time period where all three coaches/OBUs are in neutral state.

After that time period OBU 3 is switched in master state by an external trigger.

When OBU 3 is switched from neutral to master state it sends a **SetSlaveMode** request to OBUs 1 and 2. On the reception of this event both OBUs start to re-initialise their service communication.

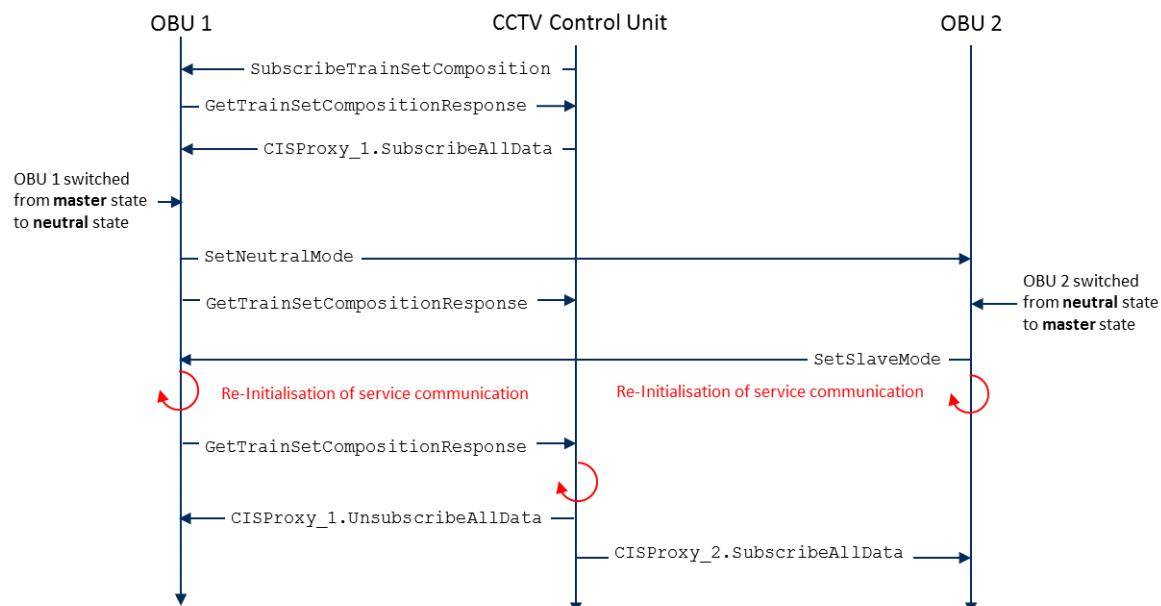
OBU 2 which formerly retrieved trip information from OBU 1, now has to switch to retrieve this information from OBU 3. OBU 1, which was formerly in master state, now has to switch to **RetrieveTripInformation** provided by OBU 3.

As OBU 3 is now master it requests the device status of the slave OBUs 1 and 2 regularly by means of **GetDeviceStatus** requests.

9.1.3 Possible Implementation for Non-OBU Devices in the Trainset IP Network

Only OBUs are directly informed about changes of the trainset composition by means of the **SetNeutralMode** and **SetSlaveMode** operations. Other devices can make use of the **GetTrainSetComposition** request (cf. 5.5.3) or the subscription to the trainset composition (cf. 5.5.4) which is provided by the **TrainSetManagementService**.

In the following example a trainset consisting of two coaches is considered. A CCTV control unit is connected to the trainset IP network. It could be mounted on any of the two coaches. The CCTV control unit likes to subscribe to the **CustomerInformationServiceProxy** (CISProxy) of the master OBU in order to store context information like line, destination or stop point name together with the video data.



Therefore, the CCTV control unit makes use of the operation **SubscribeTrainSetComposition** provided by the **TrainSetManagementService** of OBU 1, which is also the master in the example. The OBU 1 responds with a **GetTrainSetCompositionResponseStructure** which informs the CCTV control unit that

OBU 1 is recently the master. Thereby it is also clear that the **CustomerInformationServiceProxy** of OBU 1 is the one to subscribe to, so **CISProxy_1.SubscribeAllData** is used.

When coach 1 is switched from master mode to neutral mode OBU 1 informs OBU 2 by means of the **SetNeutralMode** operation. Due to the subscription the CCTV control unit is also informed about the change of the trainset composition by means of a **GetTrainSetCompositionResponse** from OBU 1.

When coach 2 is switched from neutral mode to master mode OBU 2 informs OBU 1 by means of the **SetSlaveMode** operation. Both OBUs can start to re-initialise their service communication.

Due to the subscription the CCTV control unit is also informed about the change of the trainset composition by means of a **GetTrainSetCompositionResponse** from OBU 1 and can start to re-initialise too. Thereby the CCTV control unit recognises that the master is now OBU 2.

Therefore the CCTV control unit performs an **UnsubscribeAllData** at the **CustomerInformationServiceProxy** of OBU 1 and a **SubscribeAllData** at the **CustomerInformationServiceProxy** of OBU 2.

Please note:

All other subscriptions or service communication in the trainset IP network (including potential video streams) network should be dropped (from both parties of the subscription) and should be re-initialised after a switch from neutral to master/slave state.

9.1.4 Possible implementation for non-OBU devices in the coach IP network

Most of the non-OBU devices and applications in a coach do not depend on the trainset composition. Devices which are connected to the coach IP network only, do not need to reinitialise services.

Nevertheless, non-OBU devices in the coach IP network might also need information about the trainset composition, e.g. to show that information somewhere on a passenger information display, or to save it in statistics data or in log files.

For those cases, the **GetTrainSetComposition** operation or the **SubscribeTrainSetComposition** operation, both provided by the **TrainSetInformationService** can be used.

They provide the same information as the equally named operations of the **TrainSetDataService**.

10 Examples

In the following sub-chapters there are some use cases with possible solutions described. Nonetheless it should be clear that the use cases represent just examples. There might be project specific circumstances which require to make use of the services in another way.

10.1 The master OBU requests the Status of Devices in a Slave Coach

10.1.1 Use Case

In a trainset with at least two coaches the driver in the master coach wants to know whether all IBIS-IP devices in the slave coach(es) are working properly.

10.1.2 Possible Implementation

The OBUs are the only devices connected to both IP networks. The OBU of coach 4712 is in slave state while the OBU of coach 4711 is in master state.

Each of the devices in the slave coach (4712) publishes its corresponding **DeviceManagementService** (DMS) in the coach IP network (green).

OBU 4712 publishes for each DMS service found in the coach IP network of coach 4712 an corresponding DMS proxy service the trainset IP network.



The master OBU finds all the DMS proxy services in the trainset IP network and sends status requests to these services (e.g. **GetDeviceStatus**) through the trainset IP network. The slave OBU does not directly send a response to these requests but forwards them to the addressed devices in the coach IP network instead. As soon as the slave OBU receives the response of the device via the coach IP network it forwards the information as a response to the original request via the trainset IP network.

Please note that thereby even the status of old VDV 300 devices can be monitored from the master coach.

10.2 Master OBU provides Passenger Information for the Slave Coaches

10.2.1 Use case

In a trainset with two or more coaches all coaches have the same destination and travel path. Passenger information displays in the slave coach can therefore show exactly the same information as passenger information displays in the master coach.

10.2.2 Possible Implementation

The master OBU provides a **CustomerInformationService** in its own coach IP network and a **CustomerInformationServiceProxy** in the trainset IP network.

The slave OBU provides a **CustomerInformationService** in the coach IP network of the slave coach. The slave OBU does not provide a **CustomerInformationServiceProxy** in the trainset IP network.

All displays in the slave coach make a subscription to the **CustomerInformationService/SubscribeAllData** provided by the slave OBU in the coach IP network.



The slave OBU finds the **CustomerInformationServiceProxy** in the trainset IP network. It is the only service of that type in the trainset IP network. The slave OBU subscribes to the **CustomerInformationServiceProxy/SubscribeAllData** and redistributes the received data by means of its own **CustomerInformationService** in the coach IP network.

10.3 Master OBU provides Trip Information for Slave OBUs in X-Y-Traffic

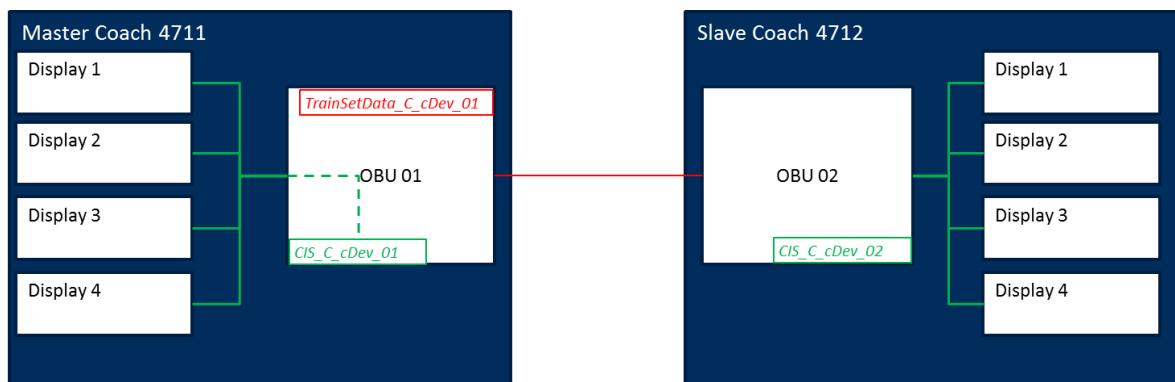
10.3.1 Use case

In a trainset with two or more coaches the master coach has a destination and travel path which is different from the destination and travel paths of the slave coach. Nonetheless, passenger information devices in the coaches have to show the correct data for each of the coaches.

10.3.2 Possible Implementation

The driver selects a trip on the OBU of the master coach. The OBU of the master coach provides both, a **CustomerInformationService** in the coach IP network of the master coach and a **TrainSetDataService** in the trainset IP network.

A slave OBU makes use of the **RetrieveTripInformation** request provided by the **TrainSetDataService**. The request informs the OBU of the master coach not only about the request itself but also about the position of the requesting coach with respect to the trainset. The master OBU searches in its own data for the trip information related to the slave coach position. Finally the master OBU provides the trip information found by means of the **RetrieveTripInformation** response. The slave OBU accepts the response and publishes the information through the **CustomerInformationService** in the coach IP network of the slave coach.



Please note that logical positioning information which is needed for the passenger information in the slave coach can be obtained through other proxy services (cf. 10.4).

10.4 Master OBU provides recent Positioning Information for the Slave OBUs

10.4.1 Use Case

In a trainset with two or more coaches the master OBU provides logical positioning information for the slave OBUs. The slave coaches might have the same destination/travel path or another one than the master OBU (X-Y-traffic).

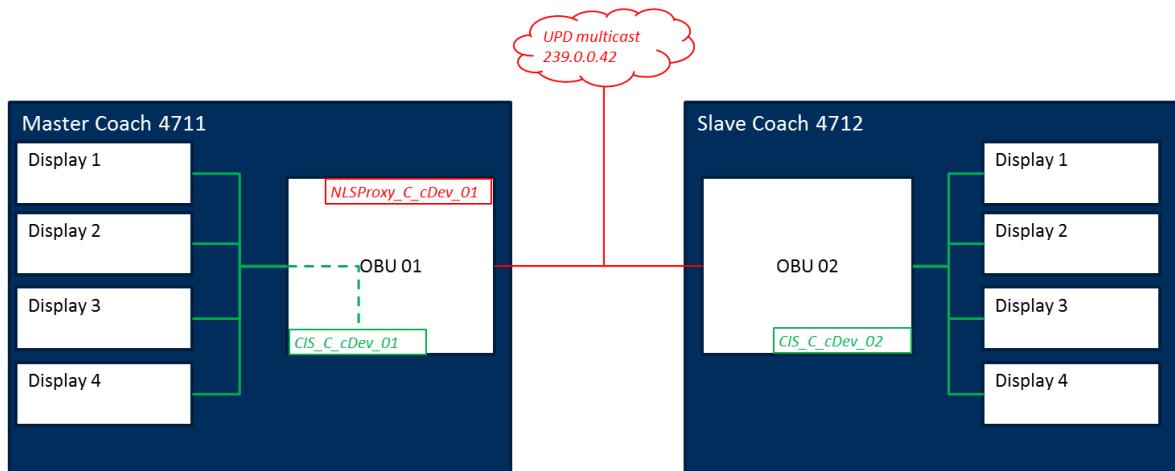
10.4.2 Possible Implementation

The OBUs are connected to both IP networks. The OBU 4712 is in slave state while the OBU of coach 4711 is in master state. It is assumed in the following that the slave OBU knows its travel path and destination e.g. by means of the **RetrieveTripInformation** request (cf. 10.3).

Nonetheless, this is not yet sufficient for the slave OBU to provide coach specific real-time passenger information through the **CustomerInformationService** in the coach IP-network. There is a real-time information about the position in the PT network necessary.

This positioning information can be taken from a **NetworkLocationServiceProxy** provided by the master OBU. The **NetworkLocationServiceProxy** publishes its data via UDP to a specific multicast address. The positioning information published there is read by the slave OBU.

By means of the data fields **NextStopPointRef**, **DistanceToNextStopPoint** and/or **LocationState** from the **NetworkLocationService.Data** structure, the OBU in the slave coach can determine the logical position of the coach within the PT network. Thereby the OBU in the slave coach can provide real-time passenger information in the coach IP network through the **CustomerInformationService**.



11 Versionshistorie / Version history

11.1 Version 2.2

11.1.1 Funktionale Erweiterungen Functional Upgrade

Keine/none

11.1.2 Technische Ergänzungen/Korrekturen Technical Upgrade/Corrections

- *GetTrainSetCompositionResponse* Struktur des TrainsetInformationService korrigiert und um eine Struktur SingleCoachInATrainSet ergänzt, es war zuvor nicht möglich, die Daten mehrerer coaches eines Trainset zu übertragen (vgl. 4.5.1), Tabellen und Bilder entsprechend angepasst.
GetTrainSetCompositionResponse structure of the TrainsetInformationService corrected and complemented by a SingleCoachInATrainSet structure, before it was not possible to transmit the data of several coaches of a trainset (cf. 4.5.1), tables and figures updated accordingly
- *GetTrainSetCompositionResponse* Struktur des TrainSetManagementService korrigiert, heißt nun TrainSetManagementService.GetTrainSetCompositionResponse statt TrainSetManagementService.GetTrainSetComposition (vgl. 5.5.3.2), Tabelle entsprechend angepasst.
GetTrainSetCompositionResponse structure of the TrainSetManagementService corrected, now correctly entitled TrainSetManagementService.GetTrainSetCompositionResponse instead of TrainSetManagementService.GetTrainSetComposition (cf. 5.5.3.2), table updated accordingly
- Zwei neue Strukturen TrainSetDataService.TrainSetSubscribeRequestStructure und TrainSetDataService.TrainSetUnsubscribeRequestStructure wurden eingeführt und erläutert in den Abschnitten 6.5.1 und 6.5.2, um eine korrekte Subscription auf eine “Retrive Daten Struktur” zu ermöglichen. Daraus ergeben sich weiterhin textliche Korrekturen in den Abschnitten 6.5, 6.5.4, 6.5.5, 6.5.7, and 6.5.8.
Two new structures TrainSetDataService.TrainSetSubscribeRequestStructure and TrainSetDataService.TrainSetUnsubscribeRequestStructure are introduced and explained in sections 6.5.1 and 6.5.2, to allow for a correct subscription to a “Retrive data structure”. This implies textual changes in sections 6.5, 6.5.4, 6.5.5, 6.5.7, and 6.5.8.

11.1.3 Textliche Korrekturen Textual Corrections

- Korrektur zur Beschreibung der CoupledSide-Enumeration in 4.5.1
correction of the description of the CoupledSide enumeration in 4.5.1
- Tippfehler in der Überschrift von Kapitel 5 korrigiert (TrainSet statt Trainset)
correction of a typo in the title of section 5 corrected (TrainSet instead of Trainset)
- Tippfehler in Kapitel 6.2 korrigiert (TrainSetDataService statt TrainSetDateService)
correction of a typo in section 6.2 corrected (TrainSetDataService instead of TrainSetDateService)

Regelwerke – Normen und Empfehlungen / References

- (1) VDV 301-2-0 IBIS-IP Beschreibung der Dienste / Service description
Basisdienste / Base Services
DeviceManagementService, SystemManagementService,
SystemDocumentationService V2.0, 02/2018
- (2) VDV 301-2-1 IBIS-IP Beschreibung der Dienste / Service description
Gemeinsame Datenstrukturen und Aufzählungstypen/
Common data structures and enumerations, 05/2017

Die IBIS-IP XSD-Dateien stehen unter www.vdv.de/ip-kom-oev.aspx zum Download bereit.

The IBIS-IP XSD files are available for download at www.vdv.de/ip-kom-oev.aspx.

Impressum

Verband Deutscher Verkehrsunternehmen e.V. (VDV)
Kamekestraße 37-39 · 50672 Köln
T 0221 57979-0 · F 0221 57979-8000
info@vdv.de · www.vdv.de

Ansprechpartner

Dipl.-Ing. Berthold Radermacher
T 0221 57979-141
F 0221 57979-8141
radermacher@vdv.de

Verband Deutscher Verkehrsunternehmen e. V. (VDV)
Kamekestraße 37-39 · 50672 Köln
T 0221 57979-0 · F 0221 57979-8000
info@vdv.de · www.vdv.de
