



HIM-SX 1.8 user manual

The HIM message exchange interface

Version 15

Stand 27. January 2021

Oliver Jürgens

Hacon Ingenieurgesellschaft mbH

Lister Straße 15

30163 Hannover

Inhaltsverzeichnis

1	Introduction	3
2	Main Usecases.....	4
3	Data transport and supported operations	5
4	HIM import results	6
5	Mapping from HIM Elements to HIM-SX.....	7
6	Examples.....	11
7	Description of the HIM-SX XSD elements	13
8	Version history	28
9	HIM-SX XSD	31

1 Introduction

HIM-SX (Hafas Information Manager - Situation Exchange) is a proprietary and XML based data format designed by HaCon. The main goal is to be able to create a representation of HIM messages to exchange data with other systems and allow import of HIM messages from external or other HIM systems. A basic knowledge of the HIM is required to work with this manual.

The data transfer and the XML design principles are the following:

- Use of an HTTP based interface for message transfer
- Provide an easy and human readable representation of HIM message data.
- Be backwards compatible - a HIM will accept all existing HIM-SX versions on the same endpoint
- Multiple HIM-SX endpoints for external system can be defined in the so called „Delivery system administration“
- Allow endpoint specific configuration and data transformation

The HIM is a product with optional features that can be purchased by customers and several feature releases are available every year. HIM-SX only represents a subset of the overall HIM functionality, grouped in different versions that build upon each other.

When it comes to importing the target HIM decides, according to its version and configuration, which HIM-SX elements are used to create a new message or discards the incoming message due to missing elements or validation constraints.

In conclusion the allowed subset of HIM-SX Elements depends on the Target HIM and the desired use case.

2 Main Usecases

It's important to be able to distinguish between the different use cases for which HIM-SX may be used.

1. Export a HIM-SX representation of a HIM message to other systems.

HIM messages can be exported as HIM-SX. The targets might be other HIMs or third party endpoints. The data of a HIM message will be transformed to HIM-SX and sent to a HTTP endpoint you can define within the HIM in form of a "Receiver system".

2. Send HIM-SX to a HIM to create HIM messages from it.

The HIM provides a HTTP endpoint and allows to specify multiple "Delivery systems", which are external systems that are allowed to create HIM messages via HIM-SX.

There are different types of messages that can be created:

- Rule based import – Like in the HIM UI you can define rules to match certain trips, which will be evaluated in the related HAFAS components.
- Trip based import – You can send a list of individual trips which are effected by the message content.
- HIM Use Case import – Use Case messages are highly preconfigured messages to reduce the required information input.

3. Allow a complex configuration for mapping of external data in the delivery system administration.

Not all data needed by the HIM might be provided by an external system. The HIM has the possibility to define defaults and mappings for certain data fields to consolidate incoming data.

To learn more about the configuration of "Receiver systems", "Delivery systems" and configuration / mapping possibilities you can study the general HIM user manual.

3 Data transport and supported operations

HIM-SX is transported via HTTP, however it is not a restfull API. The only HTTP method used to exchange data is HTTP POST for both import and export.

A basic status test of the HIM import interface via HTTP GET is supported.

The available endpoints of the HIM import interface are:

HIM-SX import: \${server:port}/\${basePath}/import/him-sx

Status check: \${server:port}/\${basePath}/import/status

\${basePath} is a general HIM configuration parameter. Ask your exchange partner or system administrator for details.

The allowed operations are transported within XML elements transported in the HIM-SX root element (`<him:himsx>`) itself:

Operation	Related HIM-SX elements	Comments
Create a new HIM message	<code><him:situationExchangeDelivery></code> <code><him:message id="Ext_ID1".....</code>	A new message will be created, if the id is unknown
Update an existing message	<code><him::situationExchangeDelivery></code> <code><him:message id="Ext_ID1".....</code>	The message will be updated, if the id is known
Deactivate a HIM message	<code><him:situationExchangeTermination></code> <code><him:message id="Ext_ID1".....</code>	Deactivate the external message "Ext_ID1"
Deactivate all HIM message of a delivery system	<code><him:situationExchangeTermination></code> <code><him:message id="TEST_SYSTEM".....</code>	Deactivate all messages of the delivery system "TEST_SYSTEM"
Delete a message		An external delete is not supported.
Status check (HTTP GET on \${base-Path}/import/status)		If the importer is running a HTTP 200 is returned.

4 HIM import results

After evaluation of the imported data the HIM returns a response according to. The optional HTTP HIM-SX response is defined in the element "**situationExchangeResponse**".

Result status	HTTP return codes
Message successfully processed	200
Syntactically invalid XML content	400
Import is not allowed	403
Invalid rest endpoints defined in the request	404
Validation erros	422
HIM internal errors - For unnown errors or bugs	500

5 Mapping from HIM Elements to HIM-SX

As mentioned above, the data of a HIM Message is transported in the **situationExchangeDelivery** element, if a new message should be created or an existing one be updated.

The HIM elements in the following table have a representation in HIM-SX 1.8. Earlier versions may have a limited set of supported elements. In some cases it is possible to define a mapping from external to internal values per delivery system.

For detailed information about the XSD, its version history, allowed fields and enumerations please refer to the HIM-SX XSD chapters.

HIM Element	HIM-SX Element	Mappings per Delivery System
HIM-Message itself	message	
Message category	categoryRef	<ul style="list-style-type: none"> Set a default category for incoming messages which overrides the original value. Define a mapping per incoming category code (e.g. external code 5 to internal code 1, "construction work" to "information") <ul style="list-style-type: none"> If there is a complex mapping defined and the incoming category code is not known use the default category if defined
Event period	eventPeriod	
Match Parameter	mode	
Affected Journeys (Result of "Show journeys")	affects	
Products	filter:products	
Vehicle Categories	filter:categories	
Operators	filter:operators	
Message Texts	messageTexts	
Priority (In Misc display Parameters)	priority	<ul style="list-style-type: none"> Set a default priority for incoming messages which overrides the original value. Define a mapping per incoming priorities <ul style="list-style-type: none"> If there is a complex mapping defined and the incoming priority is not known use the default priority if defined
Additional Links	links	<ul style="list-style-type: none"> Configure if the import of links is allowed or if it should be discarded
Publications, Push,	publications:	<ul style="list-style-type: none"> Set default publication channels, that override the

Twitter publication	publicationPeriods email push twitter	incoming publication channels <ul style="list-style-type: none"> Allow / disallow message push
---------------------	--	--

The matchparameters are transported within the different **mode** elements, where the matching rules are defined. The following table provides examples for each **mode** in relation to the HIM message templates. Not all message templates might be available within the target system, which depends on the acquired feature set.

Message Template	HIM-SX	Comment
Station	<pre><him:mode scope="TEXTUAL"> <him:station impact="AFFECTED_SECTION"> <him:affectedStop id="8000142" name="Hagen (Westf) Bahnhof" boarding="EVER"/> </him:station> </him:mode></pre>	Only one station
Multi Station	<pre><him:mode scope="TEXTUAL"> <him:stations impact="AFFECTED_SECTION"> <him:station id="8000152" name="Hannover Hbf" boarding="EVER"/> <him:station id="8011160" name="Berlin Hbf" boarding="EVER"/> </him:stations> </him:mode></pre>	One or more stations
Journey	<pre><him:mode scope="TEXTUAL"> <him:route bidirectional="false" impact="AFFECTED_SECTION"> <him:firstStop id="8000152" name="Hannover Hbf" boarding="EVER"/> <him:lastStop id="8011160" name="Berlin Hbf" boarding="EVER"/> </him:route> </him:mode></pre>	One route element
Route	<pre><him:mode scope="TEXTUAL"> <him:routes> <him:route bidirectional="true"> <him:stop id="8000152" name="Hannover " boarding="EVER"/> <him:stop id="8006552" name="Wolfsburg" boarding="EVER"/> <him:stop id="8011160" name="Berlin" boarding="EVER"/> </him:route> <him:route bidirectional="false"> <him:stop id="8000152" name="Hannover" boarding="EVER"/> <him:stop id="8000169" name="Hildesheim " boarding="EVER"/> </him:route> </him:routes> </him:mode></pre>	Multiple route elements. A route can contain 2 or more stop elements.

Message Template	HIM-SX	Comment
	<pre> </him:route> </him:routes> </him:mode> </pre>	
Country	<pre> <him:mode scope="TEXTUAL"> <him:country code="0080"/> </him:mode> </pre>	The <i>code</i> attribute contains a numeric UIC country code
Global	<pre> <him:mode scope="TEXTUAL"> <him:global/> </him:mode> </pre>	A global message
Infrastructure	<pre> <him:mode scope="TEXTUAL"> <him:infrastructure> <him:station id="8000152" name="Hannover Hbf"/> </him:infrastructure> </him:mode> </pre>	Refer to a certain station
Train	<pre> <him:mode scope="TEXTUAL"> <him:journey name="Bus 120" impact="AFFECTED_SECTION"> <him:firstStop id="980101" name="Hannover Aegi" boarding="EVER"/> <him:lastStop id="960537" name="Hannover Ahl" boarding="EVER"/> </him:journey> </him:mode> </pre>	Refer to a certain train
Lines	<pre> <him:mode scope="TEXTUAL"> <him:lines> <him:line id="5_rebus_200"> <him:direction flag="2" code="000705952" name="Güstrow ZOB"/> <him:direction flag="1" code="000705618" name="Laage Breesener Straße"/> </him:line> <him:line id="5_CEBNEU_100"> <him:direction flag="1" code="001800203" name="Celle Schlossplatz/Museum"/> <him:direction flag="2" code="001006599" name="BergenLangestraße"/> <him:direction flag="1" code="001800016" name="Celle Schlossplatz"/> <him:direction flag="1" code="001804268" name="Celle Gymnasium"/> </him:line> </him:lines> </him:mode> </pre>	The lines can contain multiple line elements.

Message Template	HIM-SX	Comment
Region	<pre> <him:mode scope="TEXTUAL"> <him:region radius="5000.0" x-coord="9.741979" y-coord="52.377231"> <him:stop id="8000152" name="Hannover Hbf"/> </him:region> </him:mode> </pre>	The <i>radius</i> attribute is defines as meters und the coordinates are based on decimal WSG 84
Operator	<pre> <him:mode scope="TEXTUAL"> <him:operator> <him:operatorRef> Operator known by HAFAS 1 </him:operatorRef> <him:operatorRef> Operator known by HAFAS 2 </him:operatorRef> </him:operator> </him:mode> </pre>	

In contrast of the rule based modes the mode **journeyList** has a different meaning. It states, that there are no rules but the individual matched journeys have to be used to create the message. The related journeys are defined within the **message:affects** element.

Message Template	Mode	Affects
Single journey	<pre> <him:mode scope="TEXTUAL"> <him:journeyList/> </him:mode> </pre>	<pre> <him:affects> <him:journeys routingAttributes="3f3"> <him:journey no="2046" matchId="2046" category="IC" name="IC 2046"> <him:affectedSection> <him:firstStop id="8000142" name="Hagen(Westf) Bahnhof" arrival="2020-07-10T13 :56:00.000+02:00" departure="2020-07-10T13 :58:00.000+02:00"/> <him:lastStop id="8000142" name="Hagen(Westf) Bahnhof" arrival="2020-07-10T13 :56:00.000+02:00" departure="2020-07-10T13 :58:00.000+02:00"/> </him:affectedSection> <him:start id="8010085" name="Dresden Hbf" departure="2020-07-10T07 :23:00.000+02:00"/> <him:destination id="8000207" name="Köln Hbf" arrival="2020-07-10T14:46 :00.000+02:00"/> </him:journey> </him:journeys> </him:affects> </pre>

6 Examples

In the following chapter you will see full examples for the operations message creation / update and deactivation

1. Example for the import of a station based message:

```
<?xml version="1.0" encoding="UTF-8"?>
<him:himsx version="1.8.0" supplierId="external" deliveryTimestamp="2020-07-
10T13:11:58.948+02:00" xmlns:him="http://www.hacon.de/hafas/himsx">
  <him:situationExchangeDelivery>
    <him:message id="123" qualifiedId="HIM_FREETEXT_123"
      sequenceNumber="1" version="1">
      <him:mode scope="TEXTUAL">
        <him:station impact="AFFECTED_SECTION">
          <him:affectedStop id="8000142"
            name="Hagen (Westf) Bahnhof" boarding="EVER"/>
        </him:station>
      </him:mode>
      <him:eventPeriod startDate="2020-07-10T13:11:00.000+02:00"
        endDate="2020-07-20T14:00:00.000+02:00"
        dailyStartTime="00:00:00.000+01:00"
        dailyEndTime="23:59:00.000+01:00"/>
      <him:messageTexts>
        <him:text>
          <him:tags>
            <him:tag>titleText</him:tag>
          </him:tags>
          <him:value lang="de">Titel </him:value>
          <him:value lang="en">Title</him:value>
        </him:text>
        <him:text>
          <him:tags>
            <him:tag>shortText</him:tag>
          </him:tags>
          <him:value lang="de">Kurztext</him:value>
          <him:value lang="en">Short text</him:value>
        </him:text>
        <him:text>
          <him:tags>
            <him:tag>longText</him:tag>
          </him:tags>
          <him:value lang="de">Lange Beschreibung</him:value>
          <him:value lang="en">Longer description</him:value>
        </him:text>
      </him:messageTexts>
      <him:publications>
        <him:publicationPeriods>
          <him:publicationPeriod start="2020-07-10T13:11:00.000+02:00"
            end="2020-07-20T14:00:00.000+02:00" channel="TIMETABLE"/>
        </him:publicationPeriods>
        <him:push publish="false"/>
      </him:publications>
      <him:priority>100</him:priority>
      <him:links>
        <him:link label="Google" url="https://google.de">
          <him:languages>
            <him:lang>DE</him:lang>
          </him:languages>
        </him:link>
      </him:links>
    </him:message>
  </him:situationExchangeDelivery>
</him:himsx>
```

```
        <him:categoryRef>0</him:categoryRef>
      </him:message>
    </him:situationExchangeDelivery>
  </him:himsx>
```

2. Example for the deactivation of an imported message:

```
<?xml version="1.0" encoding="UTF-8"?>
  <him:himsx version="1.8.0" supplierId="external"
    deliveryTimestamp="2020-07-10T13:11:58.948+02:00"
    xmlns:him="http://www.hacon.de/hafas/himsx">
    <him:situationExchangeTermination>
      <him:message id="123"/>
    </him:situationExchangeTermination>
  </him:himsx>
```

3. Example for the deactivation of all messages of a delivery system:

```
<?xml version="1.0" encoding="UTF-8"?>
<him:himsx version="1.8.0" supplierId="external"
deliveryTimestamp="2020-07-10T13:11:58.948+02:00"
xmlns:him="http://www.hacon.de/hafas/himsx">
  <him:situationExchangeTermination>
    <him:supplier id="123"/>
  </him:situationExchangeTermination>
</him:himsx>
```

7 Description of the HIM-SX XSD elements

In this section the elements and attributes of the HIM-SX document are explained and the related HIM-SX version is specified.

Legend:

- Elements are **bold**
- Attributes are *italic*
- Simple types (enumeration values) are underlined

<himsx>

The root element of a HIM-SX document

Element	Attribute	Documentation	Since version
	<i>version</i>	Defines the used HIM-SX version of the document	1.1
	<i>deliveryTimestamp</i>	Creation timestamp of the document	1.1
	<i>supplierId</i>	The identifier as expected by the receiving system	1.1
situationExchangeDelivery		A situationExchangeDelivery	1.1
situationExchangeTermination		A situationExchangeTermination	1.1
situationExchangeResponse		A situationExchangeResponse	1.1
situationExchangeDeliveryByTemplate		A situationExchangeDeliveryByTemplate	1.3.1
situationExchangeAnnouncement		A situationExchangeAnnouncement	1.4

<himsx> <situationExchangeDelivery>

Element	Attribute	Documentation	Since version
	<i>saveAction</i>		1.3.1
message			1.1

<himsx> <situationExchangeDelivery> <message>

Element	Attribute	Documentation	Since version
	<i>id</i>	Technical Id of the HIM message	1.1
	<i>qualifiedId</i>	Id as sent to the Datacollector. E.g. HIM_FREETEXT_1234	1.4
	<i>version</i>	Number representing the updates of an HIM message which were published to Hafas.	1.5
	<i>sequenceNumber</i>	Ongoing number for activations and deactivations	1.7
mode		Type of the HIM message	1.1
eventPeriod		Event Period of the HIM Message	1.1
texts		Message Texts sorted by the languages	1.1
messageTexts		Message texts sorted by the keys of the textfied	1.2.1
textByTemplate		Text templates to create the messagetext from	1.3.1
publicationPeriods		Publication channels with publication period	1.1
publications		Superset of all receivers of a HIM message. E.g. the Hafas based publication channels like TIMETABLE with publication period, Twitter, E-Mail and Push	1.4
affects		Single journeys of the message	1.1
priority		Priority of the HIM message	1.1

Element	Attribute	Documentation	Since version
delay		Delay minutes	1.1
filter		Filter to limit the results of rule based matching	1.1
links		Links to external sources	1.1
categoryRef		Him Message category	1.1
categories		Multiple Him message categories	1.1
additionalParameters		Additional key value based parameters	1.1
attachments		For Base64 encoded attachments. E.g. file uploads of links	1.2.1
comments		For the transport of HIM message comments	1.6.0
cause		Textual information about the cause of this message	1.7.0
effect		Textual information about the effect of this message	1.7.0

<himsx> <situationExchangeTermination> <message> or <supplier>

<himsx> <situationExchangeTermination> <message>

Attribute	Documentation	Since version
<i>id</i>	An identifier of this message	1.1
<i>sequenceNumber</i>	Ongoing number for activations and deactivations	1.7

<himsx> <situationExchangeTermination> <supplier>

Attribute	Documentation	Since version
<i>id</i>	An identifier of this message	1.1

<himsx> <situationExchangeResponse>

Attribute	Documentation	Since version
<i>status</i>		1.1
<i>statusCode</i>		1.1

Attribute	Documentation	Since version
<i>description</i>		1.1
<i>feedbackToken</i>		1.3.1

<himsx> <situationExchangeAnnouncement>

Attribute	Documentation	Since version
<i>status</i>	See <u>SaveActionEnum</u>	1.4
<i>msgCount</i>	Number of expected HIM messages	1.4

<SaveActionEnum>

Value	Documentation	Version
<u>BEGINN</u>	Announcement sent before the HIM-Sx Complete Export is started to indicate the beginning	1.4.0
<u>END</u>	Announcement sent after the HIM-Sx Complete Export has ended to indicate the end	1.4.0

<himsx> <situationExchangeDeliveryByTemplate>

Element	Attribute	Documentation	Since version
	<i>saveAction</i>		1.3.1
message			1.3.1

<himsx> <situationExchangeDeliveryByTemplate> <message>

Element	Attribute	Documentation	Since version
	<i>id</i>	External Id of this message. The internal Id might be different	1.3.1
humanAccidentAtStop		A humanAccidentAtStop template message contains a stop and an eventPeriod element	1.3.1
humanAccidentAtRoute		A humanAccidentAtRoute template message contains two stop and an eventPeriod element	1.3.1

<himsx> <situationExchangeDelivery> <message> <mode>

Element	Attribute	Documentation	Since version
station		See HIM message mode <i>station</i>	1.1
route		See HIM message mode <i>journey</i>	1.1
routes		See HIM message mode <i>route</i>	1.2
journey		See HIM message mode <i>train</i>	1.1
line		See HIM message mode <i>line</i>	1.1
country		See HIM message mode <i>country</i>	1.1
global		See HIM message mode <i>global</i>	1.1
journeyname		See HIM message mode <i>trainname</i>	1.1
region		See HIM message mode <i>region</i>	1.1
journeyList		Mode of Single journey messages	1.5
operator		See HIM message mode <i>operator</i>	1.5
lines		Mode for line messages	1.7
stations		Mode for multistation messages	1.7
infrastructure		Mode for infrastructure messages	1.8

<himsx> <situationExchangeDelivery> <message> <eventPeriod>

At the moment only a single eventPeriod can be defined

Element	Attribute	Documentation	Since version
	<i>startDate</i>		1.1

Element	Attribute	Documentation	Since version
	<i>endDate</i>		1.1
	<i>dailyStartTime</i>		1.1
	<i>dailyEndTime</i>		1.1
	<i>validDays</i>		1.1

<himsx> <situationExchangeDelivery> <message> <text>

Marked as deprecated since 1.7 - use **<himsx> <situationExchangeDelivery> <message> <messageTexts>** instead

Old element for transporting multilingual message texts

Element	Attribute				Documentation	Since version
	realtimeConflict					1.1
texts	Element	Attribute	Documentation	version		
		lang		1.1		
	title			1.1		
	short			1.1		
	long			1.1		
	twitter			1.1		
	teletext			1.1		
	mapCustomer			1.1		
	mapSale			1.1		
	ticker			1.1		
	customText			1.1		
	alternativeMessageSection			1.1		

<himsx> <situationExchangeDelivery> <message> <messageTexts>

Element	Attribute	Documentation	Since version
---------	-----------	---------------	---------------

Element	Attribute				Documentation	Since version
	<i>realtimeConflict</i>					1.2.1
text	Element	Attribute	Dokumentation	Version		1.2.1
	tags		A list of text tags	1.7.0		
	value	lang		1.2.1		

Default Values known by the HIM:

alternativeMessageSection
customText
longText
mapCustomer
mapSale
shortText
teletext
ticker
titleText
twitter
email

<himsx> <situationExchangeDelivery> <message> <textByTemplate> <template>

Element	Attribute	Documentation	Since version
code		Code of the Template	1.3.1
variable		List of Template Variables with code and value	1.3.1

<himsx> <situationExchangeDelivery> <message> <publicationPeriods> <publicationPeriod>

Element	Attribute	Documentation	Since version
start			1.1
end			1.1
channel		Using type <u>PublicationChannelEnum</u>	1.1

<himsx> <situationExchangeDelivery> <message> <publications>

Element	Attribute	Documentation	Since version
publicationPeriods		See the documentation publicationPeriods	1.4
email			1.4
push			1.4
twitter			1.4

<himsx> <situationExchangeDelivery> <message> <publications> <push>

Element	Attribute	Documentation	Since version
	<i>push</i>	true if the original message was push relevant	1.4
	<i>timestamp</i>	Pushtimestamp	1.5

<himsx> <situationExchangeDelivery> <message> <publications> <twitter>

Element	Attribute	Documentation	Since version
	<i>publish</i>	Set to true to publish	1.4
	<i>all</i>		1.4
twitterAccounts			1.4

<himsx> <situationExchangeDelivery> <message> <publications> <email>

Element	Attribute	Documentation	Since version
	<i>publish</i>	Set to true to publish	1.4
emailGroups		Email group codes	1.4
emailAddresses		Email addresses	1.4

<himsx> <situationExchangeDelivery> <message> <affects>

Element	Attribute	Documentation	Since version
	<i>usedAsFilter</i>		1.3.1

Element	Attribute	Documentation	Since version
	<i>trafficDays</i>	Hex Value representing the days when the journey is running	1.4
journeys			1.1
	<i>routingAttributes</i>		1.8

<himsx> <situationExchangeDelivery> <message> <affectedDisplays> <display>

Element	Documentation	Since version
id	Id of the display	1.2
name	Name of the display	1.2

<himsx> <situationExchangeDelivery> <message> <filter>

Element	Attribute	Documentation	Since version
products		A list of products	1.1
categories		A list of vehicle categories	1.1
operator		A list of operators	1.1
journeyNames		A list of journey names	1.1
journeyNumber		A list of journey numbers	1.7
lineIds		A list of line ids	1.7
directions		A list of directions	1.7

<himsx> <situationExchangeDelivery> <message> <links>

Element	Attribute	Documentation	Since version
	<i>url</i>		1.1
	<i>label</i>		1.1
publicationChannel		Using type PublicationChannelEnum	1.1
languages			1.1

<himsx> <situationExchangeDelivery> <message> <categories>

Element	Attribute	Documentation	Since version
categoryRef		A List of HIM message categories.	1.1

<himsx> <situationExchangeDelivery> <message> <additionalParameters> <parameter>

Element	Documentation	Since version
key		1.1
value		1.1

<himsx> <situationExchangeDelivery> <message> <attachements> <attachment>

Element	Attribute	Documentation	Since version
	<i>fileName</i>		1.2
	<i>name</i>		1.2
	<i>type</i>		1.2
languages		A list of languages Codes	1.2
data			1.2

<himsx> <situationExchangeTermination>

Element	Documentation	Since version
message	Termination message for a single him message attribute=id	1.1
supplier	Termination message for all messages of the given supplier identifier attribute=supplier	1.1

<himsx> <situationExchangeResponse>

Element	Documentation	Since version
status	HTTP Status code like response to indicate the status of the import	1.1
statusCode	A code representing a details status messe	1.1
description	Description of the result	1.1

Element	Documentation	Since version
feedbackToken	A token for the Feedback feature. With this token you can request detailed information about a previously and successfully imported message	1.4
problems	Returns information about mapping problems	1.5

Common elements and enumerations

<JourneyType>

Element	Attribute	Documentation	Since version
affectedSection			1.1
start		Start stop of the journey	1.1
destination		Last stop of the journey	1.1
	no	Number of the journey	1.1
	category		1.1
	name		1.1
	matchId		1.5
	trafficDays		1.4
	routingAttributes		1.8

<JourneyType> <affectedSection>

Element	Documentation	Since version
firstStop	Type: StopStructure First matchstop of the journey	1.1
lastStop	Type: StopStructure Last matchstop of the journey	1.1

<EventPeriodType>

Attribute	Documentation	Since version
<i>startDate</i>		1.1
<i>endDate</i>		1.1

Attribute	Documentation	Since version
<i>dailyStartTime</i>		1.1
<i>dailyEndTime</i>		1.1
<i>validDays</i>		1.1

<SimpleStopType>

Attribute	Documentation	Since version
<i>id</i>	Id of a stop	1.7
<i>name</i>	Name of a stop	1.7

<SimpleStopWithBoardingType> extends **SimpleStopType**

Attribute	Documentation	Since version
<i>boarding</i>	See BoardingEnum	1.7
<i>boardingRelevance</i>	See BoardingRelevanceEnum	1.8

<StopStructure> extends **SimpleStopType**

Attribute	Documentation	Since version
<i>departure</i>	Departure time at a stop	1.1
<i>arrival</i>	Arrival time at a stop	1.1

<DirectionType>

Attribute	Documentation	Since version
<i>name</i>	HAFAS line name	1.7
<i>code</i>	HAFAS code for direction	1.7
<i>flag</i>	HAFAS direction flag	1.7

<PublicationChannelEnum>

Value	Documentation	Version
<u>TIMETABLE</u>		1.1

Value	Documentation	Version
<u>RSS</u>		1.1
<u>RSS_LINE_10</u>		1.2.1
<u>RSS_LINE_30</u>		1.5
<u>RSS_LINE_50</u>		1.5
<u>RSS_LINE_60</u>		1.5
<u>RSS_LINE_70</u>		1.5
<u>RSS_LINE_80</u>		1.5
<u>RSS_LINE_90</u>		1.5
<u>HAFAS_PUSH_STAFF</u>		1.5
<u>HAFAS_PUSH_CUSTOMER</u>		1.5
<u>EMAIL</u>		1.1
<u>MESSAGELIST_SALE</u>		1.1
<u>MESSAGELIST_CUSTOMER</u>		1.1
<u>MAP_SALE</u>		1.1
<u>MAP_CUSTOMER</u>		1.1
<u>MODIFIED_PLANDATA</u>		1.1
<u>TELETEXT</u>		1.1
<u>WIDGET</u>		1.1
<u>CALLCENTER</u>		1.1
<u>CUSTOM1</u>		1.1
<u>CUSTOM2</u>		1.1
<u>MOBILE</u>		1.1
<u>MONITOR</u>		1.1
<u>EXTERNAL_INTERFACE</u>		1.1
<u>TWITTER</u>		1.1

Value	Documentation	Version
<u>TICKER</u>		1.1
<u>EXPORT</u>		1.1
<u>HAFAS_PUSH</u>		1.1
<u>LIVEMAP</u>		1.2.1

<SaveActionEnum>

Value	Documentation	Version
<u>SAVE</u>	Save the incoming changes, but don't publish them to any channels	1.3.1
<u>ACTIVE</u>	Save and publish the changes	1.3.1

<BoardingEnum>

Value	Documentation	Version
<u>IN</u>		1.1
<u>OUT</u>		1.1
<u>IO</u>		1.1
<u>EVER</u>		1.1

<BoardingRelevanceEnum>

Value	Documentation	Version
<u>IN</u>		1.8
<u>OUT</u>		1.8
<u>CHANGE</u>		1.8
<u>DRIVE_THROUGH</u>		1.8
<u>IN_MIDDLE</u>		1.8
<u>OUT_MIDDLE</u>		1.8
<u>CHANGE_MIDDLE</u>		1.8

<ScopeEnum>

Value	Documentation	Version
<u>TEXTUAL</u>	Default and indicating that a message contains textual information	1.7
<u>CANCELLATION</u>		1.7
<u>DETOUR</u>		1.7

8 Version history

Tracking of changes made the mentioned version

HIM-SX 1.8

Added:

<himsx> <situationExchangeDelivery> <message> <affects *routingAttributes*="">
<himsx> <situationExchangeDelivery> <message> <affects> <journeys *routingAttributes*="">
<himsx> <situationExchangeDelivery> <message> <mode> <infrastructure>

In **SimpleStopWithBoardingType** <boardingRelevance>

In **JourneyType** *routingAttributes*

Changed:

<himsx> <situationExchangeDelivery> <message> <mode> <stations> <station> Change Type from SimpleStopType to **SimpleStopWithBoardingType**

DirectionType *flag* - changed from byte to string

HIM-SX 1.7

Added:

<himsx> <situationExchangeDelivery> <message *sequenceNumber*="">
<himsx> <situationExchangeTermination> <message *sequenceNumber*="">
<himsx> <situationExchangeDelivery> <message> <mode> <lines> including new types <lineType> and <directionType>
<himsx> <situationExchangeDelivery> <message> <mode> <stations>
<himsx> <situationExchangeDelivery> <message> <mode *scope*=""> including new type <ScopeEnum>
<himsx> <situationExchangeDelivery> <message> <cause>
<himsx> <situationExchangeDelivery> <message> <effect>
<himsx> <situationExchangeDelivery> <message> <filter><journeyNames>
<himsx> <situationExchangeDelivery> <message> <filter><lineIds>
<himsx> <situationExchangeDelivery> <message> <filter><directions>

SimpleStopWithBoardingType to combine a **SimpleStopType** with a **BoardingEnum** to simplify the xsd

Changed:

<himsx> <situationExchangeDelivery> <message> <messageTexts> Removed the **key** from **texts** and replace it by a list of **tags**

<himsx> <situationExchangeDelivery> <message *version*=""> Type is now long instead of int

Extracted: Extracted base attributes "id" and "name" from the **StopStructure** to **SimpleStopType** to have to different types for stop entities and station with a journey context

Replaced: **StopStructure** by **SimpleStopType** or **SimpleStopWithBoardingType** where a Stop without arrival and departure times was sufficient

Smaler enhancements like removing unnecessary min/maxocurs with default values or moving certain

HIM-SX 1.6

Added:

<himsx> <situationExchangeDelivery> <message> <comments>

HIM-SX 1.5

Added: <himsx> <situationExchangeDelivery> <message **version=""**>

Added: <himsx> <situationExchangeResponse> <problems>

Added: <himsx> <situationExchangeDelivery> <message> <publications> <push **timestamp=""**>

Added: <himsx> <situationExchangeDelivery> <message> <mode> <journeyList>

Added: <himsx> <situationExchangeDelivery> <message> <mode> <operator>

Added: <PublicationChannelEnum> Added missing enum values

Extracted: <himsx> <situationExchangeDelivery> <message> <affects> <journey> Was moved to **JourneyType**

HIM-SX 1.4

Added:

<himsx> <situationExchangeDelivery> <message **qualifiedId=""**>

<himsx> <**situationExchangeAnnouncement**>

<himsx> <situationExchangeDelivery> <message> <publications>

<himsx> <situationExchangeDelivery> <message> <affects> <journeys **trafficDays=""**>

Changed:

<himsx> <situationExchangeDelivery> <message> <messageTexts **realtimeConflict=""**>

Usage is no longer required

HIM-SX 1.3.1

Added:

<himsx> <situationExchangeResponse> <feedbackToken>

<himsx> <**situationExchangeDeliveryByTemplate**>

<himsx> <situationExchangeDelivery> <affects> <journeys **usedAsFilter=""**>

<himsx> <situationExchangeDelivery> <message> <textByTemplate>

<himsx> <situationExchangeDelivery **saveAction=""**>

Changed:

<himsx> <situationExchangeDelivery> <message> <text> OR <messageTexts>: **It's either or now**

<himsx> <situationExchangeDelivery> <message> <publicationPeriods>: **Change to optional**

HIM-SX 1.2.1

Added:

```
<himsx> <situationExchangeDelivery> <message> <affectedDisplays>
<himsx> <situationExchangeDelivery> <message> <messageTexts>
<himsx> <situationExchangeDelivery> <message> <attachements>
<himsx> <situationExchangeDelivery> <message> <mode> <region x-coord="" y-coord="">
<himsx> <situationExchangeDelivery> <message> <mode> <routes>
<PublicationChannelEnum>: rss_line10, livemap
```

HIM-SX and HIM versions

The following list tracks the relation between HIM and HIM-SX versions

HIM-SX Version	HIM Version	Date
1.1	HIM 2	09.05.2014
1.2.1	HIM 2	05.08.2016
1.3.1	HIM 2 (1.08.0)	17.10.2017
1.4	HIM 2 (1.14.0)	30.04.2018
1.5	HIM 2 (1.17.0)	16.11.2018
1.6	HIM 4.0	06.19.2019
1.7	HIM 4.2	27.11.2019
1.8	HIM 4.3	14.04.2020

9 HIM-SX XSD

HIM-SX 1.8 is based on the following XSD:

```
<xsd:schema xmlns="http://www.hacon.de/hafas/himsx"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.hacon.de/hafas/himsx"
elementFormDefault="qualified"
version="1.8.0">

  <xsd:element name="himsx">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="HimSituationExchangeStructure"/>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="HimSituationExchangeStructure">
    <xsd:choice>
      <xsd:element name="situationExchangeDelivery">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="message">
              <xsd:complexType>
                <xsd:complexContent>
                  <xsd:extension base="HimMessageStructure">
                    <xsd:attribute name="id" type="xsd:string"
use="required"/>
                    <xsd:attribute name="qualifiedId" type="xsd:string"/>
                    <xsd:attribute name="version" type="xsd:long"
use="required"/>
                    <xsd:attribute name="sequenceNumber"
type="xsd:long"/>
                  </xsd:extension>
                </xsd:complexContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
          <xsd:attribute name="saveAction"
type="SaveActionEnum"
default="ACTIVATE"/>
        </xsd:complexType>
      </xsd:element>

      <xsd:element name="situationExchangeByTemplate">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="message"
type="HimTemplateMessageStructure"/>
          </xsd:sequence>
          <xsd:attribute name="saveAction"
type="SaveActionEnum"
default="ACTIVATE"/>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>
</xsd:schema>
```



```

<xsd:element name="situationExchangeTermination">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="message">
        <xsd:complexType>
          <xsd:attribute name="id" type="xsd:string"
            use="required"/>
          <xsd:attribute name="sequenceNumber" type="xsd:long"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="supplier">
        <xsd:complexType>
          <xsd:attribute name="id" type="xsd:string"
            use="required"/>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
<xsd:element name="situationExchangeResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="status" type="ResponseStatusEnum"/>
      <xsd:element name="statusCode" type="xsd:string"
        minOccurs="0"/>
      <xsd:element name="description" type="xsd:string"
        minOccurs="0"/>
      <xsd:element name="feedbackToken" type="xsd:string"
        minOccurs="0"/>
      <xsd:element name="problems" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="journeys" minOccurs="0">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="unknownStops" minOccurs="0">
                    <xsd:complexType>
                      <xsd:sequence>
                        <xsd:element name="stop"
                          type="StopStructure"
                          maxOccurs="unbounded"/>
                      </xsd:sequence>
                    </xsd:complexType>
                  </xsd:element>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="unknownJourneys"
              minOccurs="0">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="journey"
                    type="JourneyType"
                    maxOccurs="unbounded"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

```

    </xsd:complexType>
  </xsd:element>
  <xsd:element name="situationExchangeAnnouncement">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="announcement" maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="completeExport">
                <xsd:complexType>
                  <xsd:attribute name="status"
                                type="CompleteExportStatusEnum"/>
                  <xsd:attribute name="msgCount"
                                type="xsd:unsignedInt"/>
                </xsd:complexType>
              </xsd:element>
            </xsd:choice>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:choice>
<xsd:attribute name="version" type="VersionEnum" use="required"/>
<xsd:attribute name="deliveryTimestamp" type="xsd:dateTime"
use="required"/>
<xsd:attribute name="supplierId" type="xsd:string"/>
</xsd:complexType>

<xsd:complexType name="HimMessageStructure">
  <xsd:sequence>
    <xsd:element name="mode" type="HimMessageModeStructure"/>
    <xsd:element name="eventPeriod" type="EventPeriodType"/>
    <xsd:choice>
      <xsd:element name="texts">
        <xsd:annotation>
          <xsd:documentation xml:lang="en">
            Use the messageTexts element instead.
          </xsd:documentation>
        </xsd:annotation>
      <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
          <xsd:element name="text">
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name="title" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="short" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="long" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="twitter" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="teletext" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="mapCustomer" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="mapSale" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="ticker" type="xsd:string"
                              minOccurs="0"/>
                <xsd:element name="customText" type="xsd:string"

```

```

        minOccurs="0"/>
        <xsd:element name="alternativeMessageSection"
            type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="lang" type="xsd:language"
        use="required"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="realtimeConflict" type="xsd:long"
    use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="messageTexts">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="text" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="tags">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element type="xsd:string" name="tag"
                                        maxOccurs="unbounded"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="value" maxOccurs="unbounded">
                            <xsd:complexType>
                                <xsd:simpleContent>
                                    <xsd:extension base="xsd:string">
                                        <xsd:attribute name="lang"
                                            type="xsd:language"
                                            use="required"/>
                                    </xsd:extension>
                                </xsd:simpleContent>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:attribute name="realtimeConflict" type="xsd:long"/>
</xsd:element>
<xsd:element name="textByTemplate">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="template" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="variable" maxOccurs="unbounded"
                            minOccurs="0">
                            <xsd:complexType>
                                <xsd:attribute name="code" type="xsd:string"/>
                                <xsd:attribute name="value" type="xsd:string"/>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                    <xsd:attribute name="code" type="xsd:string"
                        use="required"/>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>

```

```

        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:choice>
<xsd:element name="publicationPeriods" minOccurs="0">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            Use the publicationPeriods in the publications element instead.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
            <xsd:element name="publicationPeriod">
                <xsd:complexType>
                    <xsd:attribute name="start" type="xsd:dateTime"
                        use="required"/>
                    <xsd:attribute name="end" type="xsd:dateTime"
                        use="required"/>
                    <xsd:attribute name="channel"
                        type="PublicationChannelEnum"
                        use="required"/>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="publications" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="publicationPeriods" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="publicationPeriod"
                            maxOccurs="unbounded">
                            <xsd:complexType>
                                <xsd:attribute name="start" type="xsd:dateTime"
                                    use="required"/>
                                <xsd:attribute name="end" type="xsd:dateTime"
                                    use="required"/>
                                <xsd:attribute name="channel"
                                    type="PublicationChannelEnum"
                                    use="required"/>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="email" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="emailGroups" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="groupCode" type="xsd:string"
                            minOccurs="0"
                            maxOccurs="unbounded"/>
                    </xsd:sequence>
                <xsd:attribute name="all" type="xsd:boolean"/>
            </xsd:complexType>
        </xsd:element>
    <xsd:element name="emailAddresses" minOccurs="0">

```

```

        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="mailAddress" type="xsd:string"
                    maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
<xsd:attribute name="publish" type="xsd:boolean"
    use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="push" minOccurs="0">
    <xsd:complexType>
        <xsd:attribute name="publish" type="xsd:boolean"
            use="required"/>
        <xsd:attribute name="timestamp" type="xsd:dateTime"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="twitter" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="twitterAccounts">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="accountCode" type="xsd:string"
                            minOccurs="0"
                            maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
        <xsd:attribute name="publish" type="xsd:boolean"
            use="required"/>
        <xsd:attribute name="all" type="xsd:boolean"/>
    </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="affects" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="journeys">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="journey" type="JourneyType"/>
                    </xsd:sequence>
                    <xsd:attribute name="usedAsFilter" type="xsd:boolean"/>
                    <xsd:attribute name="routingAttributes"
                        type="xsd:string"/>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="affectedDisplays" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="display" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>

```

```

        <xsd:element type="xsd:string" name="name"
                    minOccurs="0"/>
        <xsd:element type="xsd:string" name="id"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="priority" type="xsd:int" minOccurs="0"/>
<xsd:element name="delay" type="xsd:integer" minOccurs="0"/>
<xsd:element name="filter" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="products" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="productRef" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="categories" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="categoryRef" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="operator" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="operatorRef" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="journeyNames" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="journeyName" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="journeyNumbers" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="journeyNumber" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="lineIds" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="lineId" type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="directions" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence maxOccurs="unbounded">
                        <xsd:element name="direction" type="directionType"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>

```

```

        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="links" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
            <xsd:element name="link">
                <xsd:complexType>
                    <xsd:sequence minOccurs="0">
                        <xsd:element name="publicationChannels" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence maxOccurs="unbounded">
                                    <xsd:element name="channel"
                                        type="PublicationChannelEnum"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="languages" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence maxOccurs="unbounded">
                                    <xsd:element name="lang" type="xsd:language"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            <xsd:attribute name="url" type="xsd:anyURI"
                use="required"/>
            <xsd:attribute name="label" type="xsd:string"/>
        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="attachments" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="attachment" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="languages">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element type="xsd:language" name="lang"
                                        maxOccurs="unbounded"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element type="xsd:string" name="data"/>
                    </xsd:sequence>
                    <xsd:attribute type="xsd:string" name="fileName"
                        use="required"/>
                    <xsd:attribute type="xsd:string" name="name"/>
                    <xsd:attribute type="xsd:string" name="type"/>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:choice minOccurs="0">
    <xsd:element name="categoryRef" type="xsd:string"/>
    <xsd:element name="categories">

```

```

        <xsd:complexType>
            <xsd:sequence maxOccurs="unbounded">
                <xsd:element name="categoryRef" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:choice>
<xsd:element name="additionalParameters" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
            <xsd:element name="parameter">
                <xsd:complexType>
                    <xsd:attribute name="key" type="xsd:string"
                        use="required"/>
                    <xsd:attribute name="value" type="xsd:string"
                        use="required"/>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="comments" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
            <xsd:element name="comment">
                <xsd:complexType>
                    <xsd:simpleContent>
                        <xsd:extension base="xsd:string">
                            <xsd:attribute name="timestamp" type="xsd:dateTime"/>
                        </xsd:extension>
                    </xsd:simpleContent>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="cause" type="xsd:string" minOccurs="0"/>
<xsd:element name="effect" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="HimTemplateMessageStructure">
    <xsd:choice>
        <xsd:element name="humanAccidentAtStop">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="stop" type="SimpleStopType"/>
                    <xsd:element name="eventPeriod" type="EventPeriodType"
                        minOccurs="0"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="humanAccidentAtRoute">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="stop" type="SimpleStopType"
                        minOccurs="2" maxOccurs="unbounded"/>
                    <xsd:element name="eventPeriod" type="EventPeriodType"
                        minOccurs="0"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:choice>

```



```
</xsd:choice>
<xsd:attribute name="id" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:complexType name="HimMessageModeStructure">
  <xsd:choice>
    <xsd:element name="station">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="affectedStop"
            type="SimpleStopWithBoardingType"/>
        </xsd:sequence>
        <xsd:attribute name="impact" type="ImpactEnum"
          default="AFFECTED_SECTION"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="stations">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="station" maxOccurs="unbounded"
            type="SimpleStopWithBoardingType"/>
        </xsd:sequence>
        <xsd:attribute name="boarding" type="BoardingEnum"
          default="EVER"/>
        <xsd:attribute name="impact" type="ImpactEnum"
          default="AFFECTED_SECTION"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="infrastructure">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="station" maxOccurs="unbounded"
            type="SimpleStopType"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="route">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="firstStop"
            type="SimpleStopWithBoardingType"/>
          <xsd:element name="lastStop"
            type="SimpleStopWithBoardingType"/>
        </xsd:sequence>
        <xsd:attribute name="bidirectional" type="xsd:boolean"
          default="false"/>
        <xsd:attribute name="boarding" type="BoardingEnum"
          default="EVER"/>
        <xsd:attribute name="impact" type="ImpactEnum"
          default="AFFECTED_SECTION"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="routes">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="route" maxOccurs="unbounded">
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name="stop"
                  type="SimpleStopWithBoardingType"
                  maxOccurs="unbounded" minOccurs="2"/>
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:choice>
</xsd:complexType>
```

```

        <xsd:attribute type="xsd:boolean" name="bidirectional"
                      use="required"/>
    </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="journey">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="firstStop"
                          type="SimpleStopWithBoardingType"/>
            <xsd:element name="lastStop"
                          type="SimpleStopWithBoardingType"/>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string" use="required"/>
        <xsd:attribute name="boarding" type="BoardingEnum"
                      default="EVER"/>
        <xsd:attribute name="impact" type="ImpactEnum"
                      default="AFFECTED_SECTION"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="line">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="directions">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="direction" type="SimpleStopType"
                                      maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string" use="required"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="lines">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element type="lineType" name="line"
                          maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="country">
    <xsd:complexType>
        <xsd:attribute name="code" type="xsd:string" use="required"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="global"/>
<xsd:element name="journeyname">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="names">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="name" type="xsd:string"
                                      maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

```

```

        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:element name="region">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="stop" type="SimpleStopType"/>
        </xsd:sequence>
        <xsd:attribute name="radius" type="xsd:double" use="required"/>
        <xsd:attribute name="x-coord" type="xsd:double"/>
        <xsd:attribute name="y-coord" type="xsd:double"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="journeyList"/>
<xsd:element name="operator">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="operatorRef" type="xsd:string"
                maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:choice>
<xsd:attribute name="scope" type="ScopeEnum" default="TEXTUAL"/>
</xsd:complexType>

<!-- Base Type Definitions-->
<xsd:complexType name="SimpleStopType">
    <xsd:attribute name="id" type="xsd:string" use="required"/>
    <xsd:attribute name="name" type="xsd:string"/>
</xsd:complexType>

<xsd:complexType name="StopStructure">
    <xsd:complexContent>
        <xsd:extension base="SimpleStopType">
            <xsd:attribute name="departure" type="xsd:dateTime"/>
            <xsd:attribute name="arrival" type="xsd:dateTime"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="SimpleStopWithBoardingType">
    <xsd:complexContent>
        <xsd:extension base="SimpleStopType">
            <xsd:sequence>
                <xsd:element name="boardingRelevance"
                    type="BoardingRelevanceEnum"
                    maxOccurs="unbounded" minOccurs="0"/>
            </xsd:sequence>
            <xsd:attribute name="boarding" type="BoardingEnum"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="lineType">
    <xsd:sequence>
        <xsd:element type="directionType" name="direction"
            maxOccurs="unbounded" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute type="xsd:string" name="id" use="required"/>
    <xsd:attribute type="xsd:string" name="internalId"/>
    <xsd:attribute type="xsd:string" name="longName"/>

```

```
<xsd:attribute type="xsd:string" name="shortName"/>
</xsd:complexType>

<xsd:complexType name="directionType">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute type="xsd:string" name="name"/>
      <xsd:attribute type="xsd:string" name="code"/>
      <xsd:attribute type="xsd:string" name="flag"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="EventPeriodType">
  <xsd:attribute name="startDate" type="xsd:dateTime" use="required"/>
  <xsd:attribute name="endDate" type="xsd:dateTime" use="required"/>
  <xsd:attribute name="dailyStartTime" type="xsd:time"
    default="00:00:00"/>
  <xsd:attribute name="dailyEndTime" type="xsd:time"
    default="23:59:59"/>
  <xsd:attribute name="validDays" type="xsd:string"/>
</xsd:complexType>

<xsd:complexType name="JourneyType">
  <xsd:sequence>
    <xsd:element name="affectedSection">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="firstStop" type="StopStructure"/>
          <xsd:element name="lastStop" type="StopStructure"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="start" type="StopStructure"/>
    <xsd:element name="destination" type="StopStructure"/>
  </xsd:sequence>
  <xsd:attribute name="no" type="xsd:string" use="required"/>
  <xsd:attribute name="matchId" type="xsd:string"/>
  <xsd:attribute name="category" type="xsd:string"/>
  <xsd:attribute name="name" type="xsd:string"/>
  <xsd:attribute name="trafficDays" type="xsd:string"/>
  <xsd:attribute name="routingAttributes" type="xsd:string"/>
</xsd:complexType>

<!-- Enumerations -->
<xsd:simpleType name="BoardingEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="IO"/>
    <xsd:enumeration value="EVER"/>
    <xsd:enumeration value="IN"/>
    <xsd:enumeration value="OUT"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="BoardingRelevanceEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="IN"/>
    <xsd:enumeration value="OUT"/>
    <xsd:enumeration value="CHANGE"/>
    <xsd:enumeration value="DRIVE_THROUGH"/>
    <xsd:enumeration value="IN_MIDDLE"/>
    <xsd:enumeration value="OUT_MIDDLE"/>
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="CHANGE_MIDDLE"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ImpactEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="AFFECTED_SECTION"/>
    <xsd:enumeration value="END_OF_ROUTE"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ResponseStatusEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="OK"/>
    <xsd:enumeration value="ERROR"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PublicationChannelEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="CALLCENTER"/>
    <xsd:enumeration value="CUSTOM1"/>
    <xsd:enumeration value="CUSTOM2"/>
    <xsd:enumeration value="EMAIL"/>
    <xsd:enumeration value="EXPORT"/>
    <xsd:enumeration value="EXTERNAL_INTERFACE"/>
    <xsd:enumeration value="HAFAS_PUSH"/>
    <xsd:enumeration value="HAFAS_PUSH_STAFF"/>
    <xsd:enumeration value="HAFAS_PUSH_CUSTOMER"/>
    <xsd:enumeration value="LIVEMAP"/>
    <xsd:enumeration value="RSS"/>
    <xsd:enumeration value="RSS_LINE_10"/>
    <xsd:enumeration value="RSS_LINE_30"/>
    <xsd:enumeration value="RSS_LINE_50"/>
    <xsd:enumeration value="RSS_LINE_60"/>
    <xsd:enumeration value="RSS_LINE_70"/>
    <xsd:enumeration value="RSS_LINE_80"/>
    <xsd:enumeration value="RSS_LINE_90"/>
    <xsd:enumeration value="MAP_SALE"/>
    <xsd:enumeration value="MAP_CUSTOMER"/>
    <xsd:enumeration value="MESSAGELIST_CUSTOMER"/>
    <xsd:enumeration value="MESSAGELIST_SALE"/>
    <xsd:enumeration value="MOBILE"/>
    <xsd:enumeration value="MONITOR"/>
    <xsd:enumeration value="MODIFIED_PLANDATA"/>
    <xsd:enumeration value="TELETEXT"/>
    <xsd:enumeration value="TICKER"/>
    <xsd:enumeration value="TIMETABLE"/>
    <xsd:enumeration value="TWITTER"/>
    <xsd:enumeration value="WIDGET"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="VersionEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="1.0"/>
    <xsd:enumeration value="1.1"/>
    <xsd:enumeration value="1.2"/>
    <xsd:enumeration value="1.2.1"/>
    <xsd:enumeration value="1.3.0"/>
    <xsd:enumeration value="1.3.1"/>
    <xsd:enumeration value="1.4.0"/>
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="1.5.0"/>
<xsd:enumeration value="1.6.0"/>
<xsd:enumeration value="1.7.0"/>
<xsd:enumeration value="1.8.0"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SaveActionEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="SAVE"/>
    <xsd:enumeration value="ACTIVATE"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="CompleteExportStatusEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="BEGINN"/>
    <xsd:enumeration value="END"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ScopeEnum">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="TEXTUAL"/>
    <xsd:enumeration value="CANCELLATION"/>
    <xsd:enumeration value="DETOUR"/>
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```

The information contained in this documentation is the property of Hacon. The document including its annexes and any attachments are considered as confidential. By delivering these documents, Hacon presupposes that the customer accepts the agreement that the present documents must be treated confidentially and may not be made accessible to third parties without Hacon's written consent. The described solution is a software and consulting solution and will be continuously improved, thus content of the document and written realization of features may change without further notice.

© 2020 Hacon Ingenieurgesellschaft mbH