

Dokumentation

HAFAS-Internet

***HAFAS XML Schnittstelle zur
Ansteuerung von
Abfahrtstafeln***

Version 1.11

HaCon Ing. GmbH, 29.04.2010

INHALT

0	Versionkontrolle	1
1	Überblick	2
2	Der XML-Container	2
2.1	ResC.....	2
2.2	Bahnhofstafel-Ergebnis (StationBoardResult).....	3
2.2.1	SBRes	3
2.2.2	Err	3
2.2.3	SBReq	4
2.2.4	Start	4
2.2.5	Dest.....	5
2.2.6	StartT	5
2.2.7	EndT.....	5
2.2.8	JourneyList	5
2.2.9	Journey.....	6
2.2.10	ServiceDaysList	6
2.2.11	ServiceDays	7
2.2.12	MainStop.....	8
2.2.13	Product	8
2.2.14	BasicStop	8
2.2.15	Location.....	8
2.2.16	Station	9
2.2.17	ExternalId	9
2.2.18	HafasName	9
2.2.19	Platform	9
2.2.20	Address	10
2.2.21	Poi	10
2.2.22	Arr	10
2.2.23	Dep.....	10
2.2.24	JourneyAttributeList	10
2.2.25	JourneyAttribute	11
2.2.26	Attribute.....	11
2.2.27	AttributeCode.....	12
2.2.28	AttributeVariant.....	12
2.2.29	PassList.....	12
2.2.30	Time	12
2.2.31	Date	13
2.2.32	Text	13
2.2.33	NamedValue	13
2.2.34	Delay	14
2.2.35	StationInfoTextList	14
2.2.36	StationInfoText.....	14
2.2.37	InfoTextList	14
2.2.38	InfoText.....	15
2.2.39	NextUpdate.....	15
2.2.40	LastRTInfo	15

0 Versionskontrolle

Datum	Änderung
21.10.2005	Übersetzung in deutsche Sprache
20.03.2006	Version 1.6; Autor: Dietz Attribute planId 2.1 und trainId 2.2.9 hinzugefügt
20.09.2006	Version 1.7; Autor: Dietz Attribute periodBegin und periodEnd 2.1 hinzugefügt
30.01.2007	Version 1.8; Autor: Dietz Element LastRTInfo 2.2.40 hinzugefügt, PassList ist jetzt optional
22.06.2007	Version 1.9; Autor: Dietz Allgemeine Überarbeitung
30.10.2008	Version 1.10; Autor: Dietz Attribut link zum Element InfoText 2.2.38 hinzugefügt
29.04.2010	Version 1.11; Autor: Dietz Mögliche Fehlercodes des Elements Err 2.2.2 aufgelistet

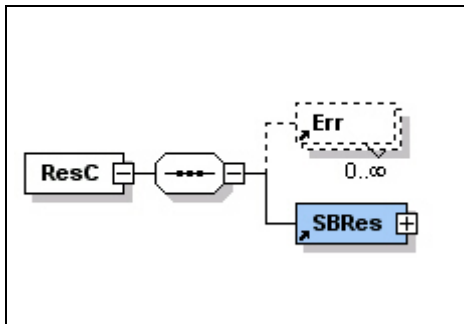
1 Überblick

Dieses Dokument beschreibt die XML-Schnittstelle der Ergebnisseite der HAFAS-Bahnhoftafeln. In den folgenden Kapiteln wird der verwendete XML-Dialekt beschrieben, inklusive Erläuterung der darin enthaltenen Elemente und Attribute.

2 Der XML-Container

Aus Gründen der Konsistenz sind alle externen Ergebnisse in Containerelementen eingeschlossen, das Wurzelement bildet hierbei immer Element **ResC**.

2.1 ResC



Beschreibung

Der Ergebniscontainer enthält beliebig viele Fehlerelemente (**Err**) und ein Bahnhoftafel-Ergebnis (**SRes**). Alle externen Ergebnisse teilen sich die folgenden drei Attribute: die Version (**ver**), den Erzeuger (**producer**) und die Sprache (**lang**). In dieser XML-Schnittstelle werden darüber hinaus auch die Plandatenversion (**planId**) sowie **Begin** (**periodBegin**) und **Ende** der Fahrplanperiode (**periodEnd**) mitgeliefert.

ver	Versionsnummer des verwendeten XML Dialekts
producer	Text mit Informationen über den XML Erzeuger
lang	iso-639 konforme Sprachkürzel
planId	Versionsnummer der verwendeten Plandaten
periodBegin	Begin der Fahrplanperiode
periodEnd	Ende der Fahrplanperiode

Syntax

```

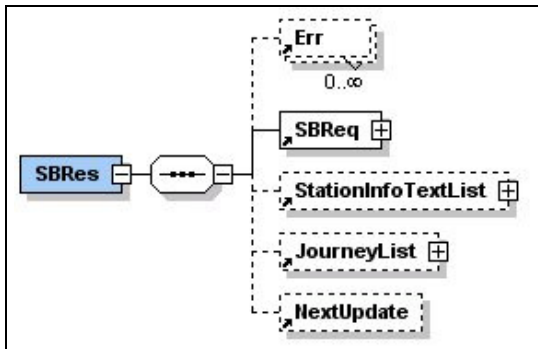
<!ELEMENT ResC (Err*, SRes)>
<!ATTLIST ResC
  ver CDATA #REQUIRED
  producer CDATA #REQUIRED
  lang CDATA #REQUIRED
  planId CDATA #IMPLIED
  periodBegin CDATA #IMPLIED
  periodEnd CDATA #IMPLIED
  
```

```

    periodEnd CDATA #IMPLIED
  >

```

2.2 Bahnhofstafel-Ergebnis (StationBoardResult)



2.2.1 SBRes

Beschreibung

Das Bahnhofstafel-Ergebnis besteht aus beliebig vielen Fehlerelementen, einer Bahnhofstafelanfrage, den optionalen Listen der Bahnhofsinformationstexte und Fahrten sowie einem oder keinem Update Element. Im Fehlerfalle wird nur das Fehlerelement mit der Bahnhofstafelanfrage zurückgeliefert.

Das einzige Attribut des Bahnhofstafel-Ergebnis-Elements ist das Richtungsflag.

dir **Richtungsflag**, mögliche Werte A(arrival) und D(departure).

Syntax

```

<!ELEMENT SBRes (Err*, SBReq, StationInfoTextList?, JourneyList?,
                NextUpdate?)>
<!ATTLIST SBRes
    dir (A | D) #REQUIRED
>

```

2.2.2 Err

Beschreibung

Im Fehlerfall werden ein oder mehrere Fehlerelemente zurückgeliefert, die den aufgetretenen Fehler näher erläutern. Dabei wird ein Fehler in drei Kategorien aufgeteilt: Fehler (error), Warnung (warning) und Nachricht (message).

Zurzeit unterscheiden wir die beiden Fehlercodes 740 und 890:

<Err code="H740" text="Your input is ambiguous" level="E"/>, wird zurückgegeben, wenn die angefragte Haltestelle nicht eindeutig aufgelöst werden konnte.

<Err code="H890" text="No trains in result" level="E"/>), kann zum einen bedeuten, dass der HAFAS-Server nicht antwortet oder, dass zu den angefragten Parametern keine Fahrten gefunden werden konnten.

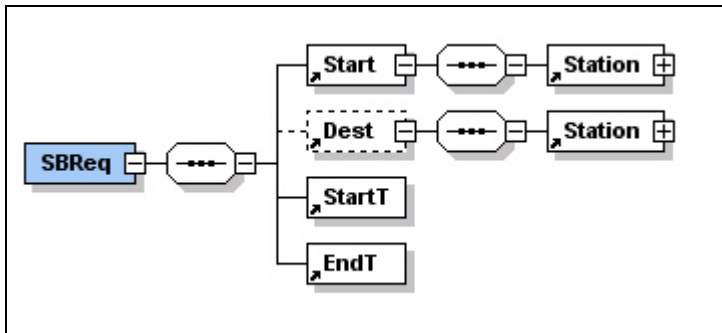
code alphanumerischer Fehlercode
text Fehlertext
level M, W oder E (message, warning oder error)

Syntax

```

<!ELEMENT Err EMPTY>
<!ATTLIST Err
  code CDATA #REQUIRED
  text CDATA #REQUIRED
  level (M | W | E) "E"
>
  
```

2.2.3 SBReq



Beschreibung

Die Bahnhofstafelanfrage enthält den angefragten Bahnhof (*Start*), einen optionalen Richtungsbahnhof (*Dest*) und die angefragte Zeitperiode (*StartT* und *EndT*).

dir **Richtungsflag**, identisch mit dem gleichnamigen Attribut aus dem Ergebnis der Bahnhofstafel.

Syntax

```

<!ELEMENT SBReq (Start, Dest?, StartT, EndT)>
<!ATTLIST SBReq
  dir (A | D) #REQUIRED
>
  
```

2.2.4 Start

Beschreibung

Start beschreibt den angefragten Bahnhof. Es gibt genau einen Bahnhof diesen Typs.

Syntax

```

<!ELEMENT Start (Station)>
  
```

2.2.5 Dest

Beschreibung

Der optionale Richtungsbahnhof wird durch dieses Element beschrieben.

Syntax

```
<!ELEMENT Dest (Station)>
```

2.2.6 StartT

Beschreibung

StartT gibt das Startdatum und die Startzeit der Anfrage an.

time Startzeit der Anfrage, Format ist HH:MM
date Startdatum der Anfrage, Format ist YYYYMMDD

Syntax

```
<!ELEMENT StartT EMPTY>  
<!ATTLIST StartT  
    time CDATA #REQUIRED  
    date CDATA #REQUIRED  
>
```

2.2.7 EndT

Beschreibung

EndT gibt das Enddatum und die Endzeit der Anfrage an.

time Endzeit der Anfrage, Format ist HH:MM
date Enddatum der Anfrage, Format ist YYYYMMDD

Syntax

```
<!ELEMENT EndT EMPTY>  
<!ATTLIST EndT  
    time CDATA #REQUIRED  
    date CDATA #REQUIRED  
>
```

2.2.8 JourneyList

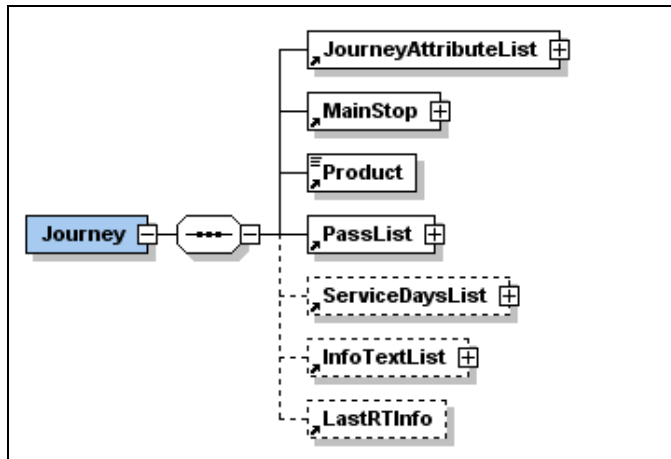
Beschreibung

Dieses Element enthält die eigentlichen Daten der Bahnhofstafelergebnisse. Es besteht aus einer Liste von Fahrten.

Syntax

```
<!ELEMENT JourneyList (Journey)*>
```

2.2.9 Journey



Beschreibung

Eine Fahrt (Journey) repräsentiert ein Verkehrsmittel (z.B. einen Zug oder einen Bus). Sie besteht aus einer Liste von Attributen, einem Haupthalt (abhängig von der Anfrage der Abfahrts- bzw. Ankunftshalt), dem eigentlichen Verkehrsmittel (z.B. Bus 42), einer Liste von durchfahrenen Bahnhöfen (PassList), den Verkehrstagen der Fahrt (Bei einer Anfrage für einen einzelnen Tag kann diese Angabe entfallen) und einer optionalen Liste von Textinformationen. Der Meldezeitpunkt einer eventuell vorhanden Verspätung wird im optionalen Element (LastRTInfo) hinterlegt.

Hinweis: Das Attribut (`trainId`) identifiziert eine Fahrt eindeutig innerhalb eines Plandatensets. Deshalb sollte zunächst immer geprüft werden, ob die `planIds` (siehe 2.1) zweier XML-Antworten übereinstimmen, bevor die `trainIds` zweier Fahrten auf Gleichheit untersucht werden.

Syntax

```
<!ELEMENT Journey (JourneyAttributeList, MainStop, Product, PassList?,
  ServiceDaysList?, InfoTextList?, LastRTInfo?)>
<!ATTLIST Journey
  trainId CDATA #REQUIRED
>
```

2.2.10 ServiceDaysList

Beschreibung

Die Verkehrstagerliste besteht aus einem oder mehr Verkehrstagen (`ServiceDays`).

Syntax

```
<!ELEMENT ServiceDaysList      (ServiceDays+)>
```

2.2.11 ServiceDays

Beschreibung

Dieses Element enthält Informationen über die Verkehrstage der Fahrt. Diese haben unterschiedliche, teilweise redundante Darstellungen:

- ein Bitfeld, in welchem jeder Tag der Fahrplanperiode durch ein Bit repräsentiert wird
- ein Text, welcher die fahrplanmäßigen Verkehrstage beschreibt (z.B. *fährt Mo-Fr*)
- ein Text, der die nicht planmäßigen Tage beschreibt (z.B. *nicht 13., 15. Apr, 3. Jun*)

Syntax

```
<!ELEMENT ServiceDays      (ServiceBits?, RegularServiceText?, IrregularServiceText?)>
```

2.2.11.1 ServiceBits

Beschreibung

Dieses Element enthält die Verkehrstage als Bitmuster, indem jeder Tag in der Fahrplanperiode durch ein Bit repräsentiert wird (erster Tag ist das ganz links liegende Bit). Das Bitfeld wird als hexadezimaler Wert gespeichert, überflüssige Bits werden auf 0 gesetzt.

Syntax

```
<!ELEMENT ServiceBits      (#PCDATA)>
```

2.2.11.2 RegularServiceText

Beschreibung

Das Element enthält einen Text (siehe 2.2.32), der die regelmäßigen Verkehrstage einer Fahrt beschreibt (z.B. Mo-Fr).

Syntax

```
<!ELEMENT RegularServiceText      Text+>
```

2.2.11.3 IrregularServiceText

Beschreibung

Die Element enthält einen Text (siehe 2.2.32), der die unregelmäßigen Verkehrstage beschreibt (z.B. *nicht 13., 15. Apr, 3. Jun*).

Syntax

```
<!ELEMENT IrregularServiceText      (Text+)>
```

2.2.12 MainStop

Beschreibung

MainStop beinhaltet ein BasicStop Element der Ankunfts- bzw. Abfahrts Haltestelle einer Fahrt.

Syntax

```
<!ELEMENT MainStop (BasicStop)>
```

2.2.13 Product

Beschreibung

Die Verkehrsmittel einer Fahrt werden mit diesem Element beschrieben.

name	Langname eines Verkehrsmittels (z.B. <i>BUS 566</i>)
image	Kurzname des Verkehrsmittels (z.B. <i>bus</i>), kann dazu genutzt werden um einen Bildnamen zu generieren (z.B. <i>bus.gif</i>)

Syntax

```
<!ELEMENT Product (#PCDATA)>
<!ATTLIST Product
  name CDATA #IMPLIED
  image CDATA #IMPLIED
>
```

2.2.14 BasicStop

Beschreibung

Das BasicStop-Element enthält alle Informationen eines Haltes. Es besteht aus einem Location-Element und aus den optionalen departure- (*dep*) und arrival- (*arr*) Elementen. Der Typ (*type*) zeigt an, ob der Halt ein virtueller Halt ist. Diese werden aus rein administrativen Gründen in die Verbindungsdaten eingefügt, um z.B. eine Tarifgrenze zu kennzeichnen. Der Index-Parameter (*index*) kennzeichnet dagegen die Position des Haltes in dem Laufweg des entsprechenden Verkehrsmittels (Zug, Bus usw.).

Syntax

```
<!ELEMENT BasicStop (Location, Arr?, Dep?)>
<!ATTLIST BasicStop
  type (NORMAL|VIRTUAL) "NORMAL"
  index CDATA #IMPLIED
>
```

2.2.15 Location

Beschreibung

Das Location-Element beschreibt einen geographischen Punkt, es kann eine Adresse (Address), ein Halt (Station) oder eine Sehenswürdigkeit (Poi) sein. Es besitzt x- und y-Koordinaten und z für die Höhe des Punktes. Der Typ der Koordinate ist standardmäßig WGS84, dieser Wert wird mit einer Million multipliziert, um Kommastellen zu vermeiden. Die id identifiziert eine Location eindeutig.

Syntax

```
<!ELEMENT Location (Station | Address | Poi)>
<!ATTLIST Location
    x CDATA #IMPLIED
    y CDATA #IMPLIED
    z CDATA #IMPLIED
    type (HAFAS_GEO | HAFAS_HANNOVER | WGS84) "WGS84"
    id ID #IMPLIED>
```

2.2.16 Station

Beschreibung

Das Element Station beschreibt einen Halt. Es besteht aus einer externen ID und einem Namen. Beide Elemente werden durch die HAFAS-Fahrplandaten bestimmt.

Syntax

```
<!ELEMENT Station (ExternalId, HafasName)>
```

2.2.17 ExternalId

Beschreibung

Die externen ID wird durch die HAFAS-Fahrplandaten festgelegt, der Parameter pooluic legt den verwendeten Pool fest und wird z.B. dann benötigt, wenn in den Daten mehrere Pools verwendet werden.

Syntax

```
<!ELEMENT ExternalId (#PCDATA)>
<!ATTLIST ExternalId
    pooluic CDATA #REQUIRED>
```

2.2.18 HafasName

Beschreibung

Dieses Element enthält den Namen eines Haltes und wird ebenfalls von den Fahrplandaten bestimmt.

Syntax

```
<!ELEMENT HafasName (Text+)>
```

2.2.19 Platform

Beschreibung

Das Gleis (Platform) beschreibt, an welchem Gleis das jeweilige Verkehrsmittel hält.

Syntax

```
<!ELEMENT Platform (#PCDATA)>
```

2.2.20 Address

Dieses Element beschreibt eine Adresse.

Syntax

```
<!ELEMENT Address (#PCDATA)>
```

2.2.21 Poi

Beschreibung

Dieses Element beschreibt eine Sehenswürdigkeit (point of interest).

Syntax

```
<!ELEMENT POI (#PCDATA)>
```

2.2.22 Arr

Beschreibung

Arr beschreibt die so genannte Ankunft (arrival) eines Verkehrsmittels an einem bestimmten Halt. Es enthält ein Zeitelement sowie optional das Ankunftsgleis und eine mögliche Verspätung (Delay). Der ebenfalls optionale Parameter `getOut` kennzeichnet, ob der Ausstieg an diesem Halt möglich ist.

Syntax

```
<!ELEMENT Arr (Time, Delay?, Platform?)>
<!ATTLIST Arr
  getOut (YES|NO) "YES"
>
```

2.2.23 Dep

Beschreibung

Analog zu 2.2.22 beschreibt Dep die Abfahrt (departure) ab einem bestimmten Halt.

Syntax

```
<!ELEMENT Dep (Time, Delay?, Platform?)>
<!ATTLIST Dep
  getIn (YES|NO) "YES"
>
```

2.2.24 JourneyAttributeList

Beschreibung

Dieses Element enthält ein oder mehrere JourneyAttribute.

Syntax

```
<!ELEMENT JourneyAttributeList (JourneyAttribute)*>
```

2.2.25 JourneyAttribute

Beschreibung

Ein JourneyAttribute ist ein Attribut, welches zu einer bestimmten Fahrt gehört. Solch ein Attribut kann z.B. nur an bestimmten Verkehrstagen oder nur für einen bestimmten Abschnitt der Fahrt gelten. Aus diesem Grunde beinhaltet es die Daten über das Attribut sowie Informationen über die Verkehrstage (siehe 2.2.11). Der betroffene Abschnitt der Fahrt wird durch die Parameter `from` und `to` bestimmt, die einen Index im Laufweg der Fahrt enthalten. Dieser Index beginnt mit 0.

Neben den „normalen“ Attributen „Bordrestaurant“ oder „Fahrradmitnahme“ werden auch der Zugname, die Produktklasse sowie der Richtungstext als spezielle Attribute abgelegt.

Syntax

```
<!ELEMENT JourneyAttribute (Attribute, ServiceDays?)>
<!ATTLIST JourneyAttribute
  from CDATA #IMPLIED
  to CDATA #IMPLIED
>
```

2.2.26 Attribute

Beschreibung

Ein Attribut enthält einen Code und einen detaillierten Text. Die folgenden Attributtypen werden unterschieden:

Type	Beschreibung
NORMAL	herkömmliches Attribut
DIRECTION	Beschreibung der Richtung eines Verkehrsmittels
CATEGORY	die Produktklasse
INTERNALCATEGORY	interne Produktklasse
OPERATOR	Betreiber
NAME	Name des Verkehrsmittels (z.B. ICE 747)
NUMBER	Nummer des Verkehrsmittels
EXTERNALID	externe Nummer des Verkehrsmittels

Der Attributtext kann in verschiedenen Varianten und unterschiedlichen Detailstufen vorkommen. Die Priorität eines Attributes ist eine ganze Zahl und legt die Reihenfolge der Darstellung von mehreren Attributen verschiedener Priorität fest.

Syntax

```
<!ELEMENT Attribute (AttributeCode?, AttributeVariant+)>
<!ATTLIST Attribute
  priority CDATA #IMPLIED
  type (NORMAL|DIRECTION|CATEGORY|
        INTERNALCATEGORY|OPERATOR|NAME|
```

```
NUMBER| EXTERNALID) "NORMAL"
>
```

2.2.27 AttributeCode

Beschreibung

Wenn der Attributtyp `NORMAL` ist, dann enthält dieses Element eine Mnemonik des zugehörigen Attributs. Bei dem Typ `CATEGORY` dagegen enthält es die Produktklasse (siehe auch die Beschreibung der „zugart“-Datei in der Transform-Input-Dokumentation).

Syntax

```
<!ELEMENT AttributeCode (#PCDATA)>
```

2.2.28 AttributeVariant

Beschreibung

Dieses Element enthält eine Variante des Attributes. Drei verschiedene Abstufungen sind möglich:

Type	Beschreibung
SHORT	Kurzbeschreibung (beschränkt auf 8 Zeichen)
NORMAL	normale Beschreibung
LONG	sehr detaillierte Beschreibung

Syntax

```
<!ELEMENT AttributeVariant (Text+)>
<!ATTLIST AttributeVariant
  type (SHORT|NORMAL|LONG) "NORMAL"
>
```

2.2.29 PassList

Beschreibung

Eine `PassList` enthält eine Liste aller durchfahrenen Halte (siehe 2.2.14) inklusive der Abfahrts- und Zielhaltestelle eines Verkehrsmittels. Wenn die `PassList` ein Teil einer Verbindung oder Bahnhofstafel ist, enthält sie nur die Halte, die für diese Verbindung relevant sind.

Syntax

```
<!ELEMENT PassList (BasicStop)*>
```

2.2.30 Time

Beschreibung

Das `Zeit`-Element beschreibt einen Zeitpunkt im Format `HH:MM`.

- `HH` enthält die Stunden in der 24 Stundendarstellung

- MM enthält die Minuten

Syntax

```
<!ELEMENT Time (#PCDATA)>
```

2.2.31 Date

Beschreibung

Das Datum ist ein String im Format YYYYMMDD.

- YYYY repräsentiert die vierstellige Jahreszahl
- MM repräsentiert den Monat
- DD repräsentiert den Tag

Syntax

```
<!ELEMENT Date (#PCDATA)>
```

2.2.32 Text

Beschreibung

Normalerweise wird dieses Element als Alternative zu #PCDATA benutzt, um mehrsprachige Informationen im XML-Strom transportieren zu können. Das Element `Text` kann momentan folgende Sprachen enthalten:

Parameter	Language
DE	Deutsch
NO	Norwegisch
EN	Englisch
FR	Französisch
MACRO	sprachunabhängige Makros, der Client muss diese selbst expandieren

Falls mehrere `Text`-Elemente gleichzeitig in einem anderen HAFAS-XML-Element auftreten, haben diese `Text`-Elemente die selbe Bedeutung in verschiedenen Sprachen.

Syntax

```
<!ELEMENT Text (#PCDATA)>
```

```
<!ATTLIST Text
  lang      (DE|EN|NO|FR|MACRO)  #IMPLIED
>
```

2.2.33 NamedValue

Beschreibung

Dieses Element beschreibt einen Wert, der zu einem Namen/Schlüssel zugeordnet ist. Es wird dazu benutzt, zusätzliche Parameter externer Module einbinden zu können, ohne die DTD erweitern zu müssen.

Syntax

```
<!ELEMENT NamedValue EMPTY>
<!ATTLIST NamedValue
  name      CDATA      #REQUIRED
  value     CDATA      #REQUIRED
>
```

2.2.34 Delay

Beschreibung

Die Verspätung (delay) beschreibt die Verspätung eines Zuges / Busses in Minuten ≥ 0 .

Syntax

```
<!ELEMENT Delay (#PCDATA)>
```

2.2.35 StationInfoTextList

Beschreibung

Dieses Element enthält beliebig viele (auch keine) StationInfoText.

Syntax

```
<!ELEMENT StationInfoTextList (StationInfoText)*>
```

2.2.36 StationInfoText

Beschreibung

StationInfoText enthält einen Halt und die dazugehörige Textinformation.

Syntax

```
<!ELEMENT StationInfoText (Station, InfoTextList?)>
```

2.2.37 InfoTextList

Beschreibung

Dieses Element enthält eine Liste von zusätzlichen Informationen zu einem Halt oder einer Verbindung.

Syntax

```
<!ELEMENT InfoTextList (InfoText)*>
```


2.2.38 InfoText

Beschreibung

Dieses Element enthält eine zusätzliche Information zu einem Halt oder einer Verbindung. Es setzt sich aus Texten verschiedener Länge (normal, kurz und detailliert) zusammen. Zusätzlich gibt es noch eine Priorität (priority) und einen Typ wie z.B. „normal“ Text oder „ticker“ um den Text beim Clienten wie in einem Ticker erscheinen zu lassen. Außerdem kann in „link“ eine URL auf weiterführende Informationen enthalten sein.

Syntax

```
<!ELEMENT InfoText EMPTY>
<!ATTLIST InfoText
    textL CDATA #IMPLIED
    text CDATA #IMPLIED
    textS CDATA #IMPLIED
    link CDATA #IMPLIED
    priority CDATA #IMPLIED
    type CDATA #IMPLIED
>
```

2.2.39 NextUpdate

Beschreibung

Dieses Element gibt an, wann das nächste Update der Daten durch einen Aufruf der HAFAS-Applikation erfolgen soll.

time Startzeitpunkt der Anfrage im Format HH:MM
date Datum der Anfrage im Format YYYYMMDD

Syntax

```
<!ELEMENT NextUpdate EMPTY>
<!ATTLIST NextUpdate
    time CDATA #REQUIRED
    date CDATA #REQUIRED
>
```

2.2.40 LastRTInfo

Beschreibung

Dieses Element gibt den Meldezeitpunkt der Verspätung zu einer bestimmten Fahrt an.

time Meldezeitpunkt der Verspätung im Format HH:MM
date Datum der Verspätungsmeldung im Format YYYYMMDD

Syntax

```
<!ELEMENT LastRTInfo EMPTY>
<!ATTLIST LastRTInfo
```

```
time CDATA #REQUIRED  
date CDATA #REQUIRED  
>
```