

# Vorstellung der FBS-RailML®-Schnittstelle

(RailML-Schemenversion 2.0-2.2)

**- Inhalte und Möglichkeiten -**

Stand 08/2010, ergänzt 03/2012

RailML-Version 2.0 • FBS-Schnittstellenversion 2.0.1

© Institut für Regional- und Fernverkehrsplanung

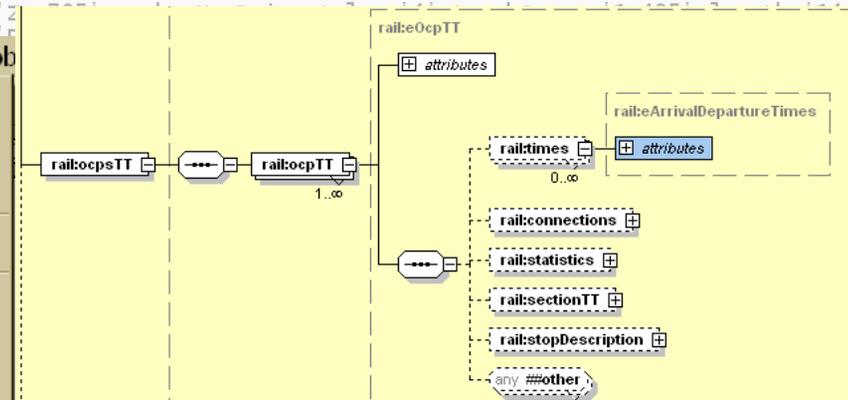
## Ausgabedaten aus FBS (Überblick)

- Infrastruktur
  - Betriebsstellen
  - Strecken und Streckengleise, Kilometrierung
  - Geschwindigkeiten, Neigungen, Bögen, Tunnel
- Fahrzeug- und Zugbildungsdaten
- Fahrplanperiode, Verkehrstage / Saisonierungen
- Zugattungen und Produktbezeichnungen
- Fahrplandaten - Zugteile
  - Laufwege der Züge, Ankunfts- und Abfahrtszeiten
  - Aufenthaltswahlungen (Halteart und -dauer)
  - Laufwegabschnittsdaten (Entfernungen, Zuschläge, Gleise)
  - Wagenreihungsangaben (Buchungsdaten)
- Verkettung von Zugteilen zu betrieblichen und verkehrlichen Zügen
- Umlaufpläne

## Bedienung aus FBS über iPLAN: Menüpunkt *Netz exportieren*

File Edit View Network

- Neues Netz Strg+N
- Netz öffnen... Strg+O
- letzte Dateien:
- Schließen Strg+F4
- Netz speichern Strg+S
- Netz speichern unter... Strg+U
- Alle Netze speichern Strg+A
- Netz exportieren**
- Drucken... Strg+P
- Druckereinstellung...
- Beenden Alt+F4



Exportieren nach RailML@2

- IVV ASS
- Microbus2 V2
- RailML@2**
- RailML Timetable 1.00
- Text V1.0
- XML

Exportieren nach RailML@2

Zieldatei: C:\Programme\FBS\Beispiele Original\iPLAN Ostsaachsen 2020 RailML\OstsachsenRailML.xml

Umfang der zu exportierenden Infrastruktur

- Strecken und Streckengleise
- Betriebsstellen
- alle
- zulässige Geschwindigkeiten
- Höhen / Neigungen
- Bögen
- Tunnel

Umfang der zu exportierenden Fahrplandaten

- Direktverbindungen (als "commercial trains")
- Streckengleise, Entfernungen und Fahrzeiten
- Züge und Zugteile filtern ...
- Zeitraum einschränken
- 13.12.2020 ... 13.12.2020

Umlaufpläne exportieren

Zum Bearbeiten eines Gültigkeitszeitraums klicken Sie mit der rechten Maustaste auf einen Umlaufplan.

Umlaufplan	gültig im FBS von - bis	gültig im Export von - bis
<input checked="" type="checkbox"/> Tzf. BR 642	13.12.2020-11.12.2021	13.12.2020-11.12.2021
<input checked="" type="checkbox"/> Tzf. BR 612	23.12.2020-29.12.2020	23.12.-29.12.

Einstellungen als Vorlage speichern

Ok Abbrechen

```
</trainOrder>
</formation>
<formation id='fmt_2' name='DB.612' length='51.750' speed='160'
  <trainOrder>
    <vehicleRef orderNumber='1' vehicleRef='veh_DB.612' vehicle
  </trainOrder>
</formation>
<formation id='fmt_3' name='DB.234+Bimd268+wLAB176+2xBocm242'
  <trainOrder>
    <vehicleRef orderNumber='1' vehicleRef='veh_DB.234' vehicle
    <vehicleRef orderNumber='2' vehicleRef='veh_Bimd268' vehic
    <vehicleRef orderNumber='3' vehicleRef='veh_wLAB176' vehicl
    <vehicleRef orderNumber='4' vehicleRef='veh_Bocm242' vehicl
```

## Filtermöglichkeiten beim Export (Überblick)

Ein Zug darf durch einen Filter nicht „auseinandergerissen“ werden, d. h. ein Filter muss immer einen vollständigen Zuglauf erfassen, so dass es nicht zu zwei Züge oder Zugteile mit gleicher Bezeichnung kommen kann. Da RailML ein allgemeingültiges Schnittstellenformat

ist, sollten die exportierten Daten unabhängig vom Zielprogramm sein.

The screenshot displays the 'Umfang der zu exportierenden Infrastruktur' and 'Umfang der zu exportierenden Fahrplandaten' sections. The 'Umfang der zu exportierenden Infrastruktur' section includes checkboxes for 'Strecken und Streckengleise', 'Betriebsstellen', 'Bögen', and 'Tunnel'. The 'Umfang der zu exportierenden Fahrplandaten' section includes checkboxes for 'Direktverbindungen (als "commercial trains")', 'Streckengleise, Entfernungen und Fahrzeiten', 'Züge und Zugteile filtern ...', and 'Zeitraum einschränken'. The 'Züge und Zugteile filtern' dialog box is open, showing various filter options such as 'nur Züge der folgenden Ebene(n) ausgeben:', 'nur Züge mit mind. einem der folgenden Status einbeziehen:', 'keinesfalls Züge mit einem der folgenden Status einbeziehen:', 'Züge nach Gattungen filtern', 'Zugteile nach Produkten filtern', and 'Zugteile nach Linien filtern'.

**Umfang der zu exportierenden Infrastruktur**

- Strecken und Streckengleise
- Betriebsstellen
  - alle
  - alle
  - ohne reine Zugfolgestellen
  - ohne reine Fahrzeitmesspunkte
- Bögen
- Tunnel

**Umfang der zu exportierenden Fahrplandaten**

- Direktverbindungen (als "commercial trains")
- Streckengleise, Entfernungen und Fahrzeiten
- Züge und Zugteile filtern ...
- Zeitraum einschränken
  - 13.12.2020 ... 13.

**Züge und Zugteile filtern**

- nur Züge der folgenden Ebene(n) ausgeben:
  - Test, Studie, Regelfahrplan, Sonderfahrplan, Ad-hoc-Trasse, gefahrene Zeiten
- nur Züge mit mind. einem der folgenden Status einbeziehen:
  - angelegt/konstruiert, zu bestellen, bestellt (EVU->EIU), Bestellung bearbeitet (EIU), angeboten (EIU->EVU), bestätigt (EVU->)
- keinesfalls Züge mit einem der folgenden Status einbeziehen:
- Züge nach Gattungen filtern
  - nur Züge der folgenden Gattung(en) einbeziehen
  - CS,D,FZ,ICE-T,ÖBB,OBC,OBE
- Zugteile nach Produkten filtern
- Zugteile nach Linien filtern
  - nicht Zugteile mit der/den folgenden Linienbezeichnung(en) einbeziehen
  - ÖBB 20 V,ÖBE 95

## Infrastruktur - Betriebsstellen

- eine Identifikation ist außer anhand der **id** auch anhand einer Abkürzung (DB Netz: *DS100*) oder (optional) anhand einer Betriebsstellenummer möglich
- grundlegende verkehrliche und betriebliche Eigenschaften (z. B. Reise-/Güterverkehr, Zugmelde- oder Zugfolgestelle)
- optional Koordinaten und Angaben zur Gemeinde

Betriebsstelle bearbeiten

Abkürzung:  ①

interner Name:  ②

Verkehrsname 1. Sprache:  ②

Verkehrsname 2. Sprache:  ③

Sie können potentielle Stellen für Zeilenumbrüche (Silbentrennung) in beiden verkehrlichen Namen mit | kennzeichnen. Die Zeichen | werden nicht mit ausgegeben.

Typ:  ④

Entgeltklasse:

Bemerkungen:

Dienststellen-Nr.:

interne Bahnhofsnr.:  ⑤

⑥ { Gemeinde:   
Gemeinde-Nr.:  PLZ:

Zeitzone:  h relativ zu GMT

⑦ { Koordinate Länge:  °  
Koordinate Breite:  °  
Höhe:  m

```

<ocp id='ocp_DBZ' abbreviation='DBZ' number='8010026' name='Bautzen'>
  <propOperational operationalType='station' orderChangeable='true' ensuresTrainSec
  <propService passenger='true' service='true' goodsLoading='true' />
  <propEquipment>
    <summary hasHomeSignals='true' hasStarterSignals='true' hasSwitches='true' />
  </propEquipment>
  <propOther>
    <additionalName value='Budyšin' type='trafficName' />
    <additionalName value='Bft. Bautzen Pbf.' type='operationalName' />
  </propOther>
  <area name='Bautzen, Stadt' number='14272010' zip='2625' />
  <geoCoord coord='14.43250 51.17220 201.47' />
</ocp>
  
```

① DBZ  
② Bautzen  
③ Budyšin  
④ Pbf. b  
⑤ 8010026  
⑥ Bautzen, Stadt  
⑦ 14.43250 51.17220 201.47

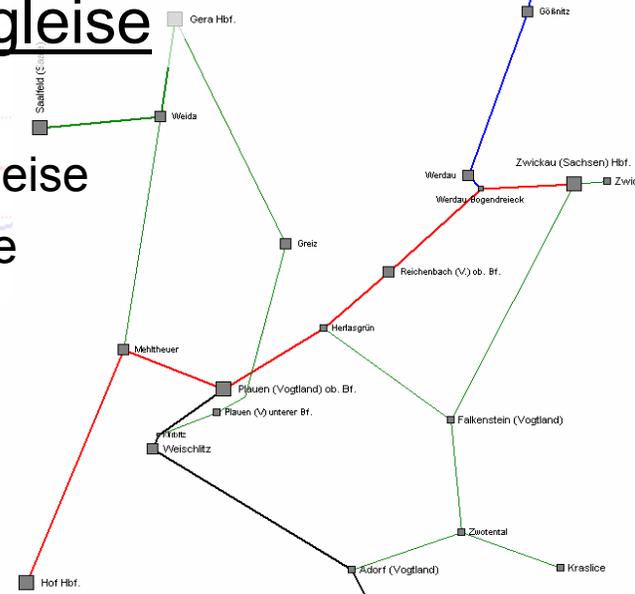
## Infrastruktur - Strecken und Streckengleise

- beschreibt Netztopologie
- Entfernungen, Kilometrierung, Anzahl Streckengleise
- optional Koordinaten und Angaben zur Gemeinde

```

<track id='tr_80.6217_1' name='DBW-DNKW' type='mainTrack'>
  <trackTopology>
    <trackBegin id='trn_DBW_80.6217_1' pos='0' absPos='9430'>
      <macroscopicNode ocpRef='ocp_DBW' />
    </trackBegin>
    <trackEnd id='trn_DNKW_80.6217_1' pos='9430' absPos='23370'>
      <macroscopicNode ocpRef='ocp_DNKW' />
    </trackEnd>
    <mileageChanges>
      <mileageChange id='mch_80.6217_1_0' absPos='9430' pos='0' dir='down' />
      <mileageChange id='mch_80.6217_1_9424' absPosIn='6' type='missing' absPos='23376' pos='9424' dir='down' />
    </mileageChanges>
    <crossSections>
      <crossSection id='trn_DBW_76A_80.6217_1' pos='1290' absPos='8140' ocpRef='ocp_DBW_76A' />
      <crossSection id='trn_DSML_80.6217_1' pos='3570' absPos='5860' ocpRef='ocp_DSML' />
      <crossSection id='trn_DPT_80.6217_1' pos='5580' absPos='3850' ocpRef='ocp_DPT' />
      <crossSection id='trn_DNKW_H_80.6217_1' pos='8930' absPos='500' ocpRef='ocp_DNKW_H' />
    </crossSections>
  </trackTopology>
</track>
<trackGroups>
  <line id='ln_80.6212' name='DN-DG' uicNumber='80' lineNumber='6212'>
    <trackRef ref='tr_80.6212_1' />
    <trackRef ref='tr_80.6212_2' />
  </line>

```



## Fahrplanperiode und Feiertage

erster Geltungstag des Fahrplanes:

letzter Geltungstag des Fahrplanes:

gesetzliche Feiertage innerhalb des Geltungsbereiches:

frei definierbare Verkehrstagezeiträume (Standard-Bereiche, z.B. Ferien):

Abkürzung	Bezeichnung	Verkehrstage
-----------	-------------	--------------

**Feiertage definieren**

Name des Verkehrstagebereiches (z.B. "Sommerferien"):

Der Bereich umfasst folgende Tage:

Bereiche einfügen ▼    veränd. Feiertage ▼

Doppelklicken Sie auf einen Tag, um diesen ein- bzw. auszuschließen.  
Halten Sie die Shift-Taste gedrückt zum Auswählen eines Bereiches.

	Dezember 2020	Januar 2021	Februar 2021	März 2021	April 2021	Mai 2021
Mo	7 14 21 28	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31
Di	1 8 15 22 29	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25
Mi	2 9 16 23 30	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26
Do	3 10 17 24 31	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27
Fr	4 11 18 25	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28
Sa	5 12 19 26	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29
So	6 13 20 27	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30

<timetablePeriods>

<timetablePeriod id='ttp\_2020\_21' name='2020/21' startDate='2020-12-13' endDate='2021-12-11'>

<holidays>

```

<holiday holidayDate='2020-12-25' />
<holiday holidayDate='2020-12-26' />
<holiday holidayDate='2021-01-01' />
<holiday holidayDate='2021-04-02' />
<holiday holidayDate='2021-04-04' />
<holiday holidayDate='2021-04-05' />
<holiday holidayDate='2021-05-01' />
    
```

## Verkehrstage / Saisonierungen

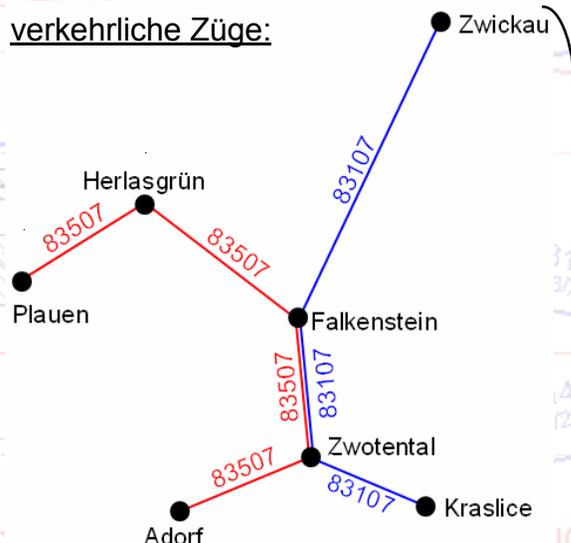
- redundante Codierung der Tage in einer Verkehrstagerregelung (*operatingPeriod*):
  - im Attribut *bitMask* für jeden Tag über die gesamte Periode sowie
  - in den Unterelementen *operatingDay* und *specialService* aufgeschlüsselt
- Hintergrund:
  - Bitmaske lässt sich softwaretechnisch gut weiterverarbeiten, jedoch nicht eindeutig in eine Nutzereingabe zurückverwandeln
  - Regel- und Sonderverkehrstage zur exakten Wiederherstellung der Eingabe

```

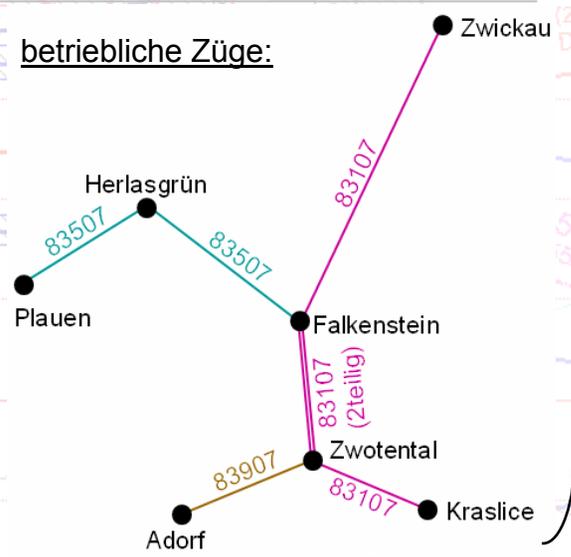
<operatingPeriod id='opp_4' name='täglich; nicht 24.,31.12.; 9.,16.,23.7.' description='nicht 24.,31.12.; 9.,16.,23.7.' timetablePeriodRef='ttp_20'
  <operatingDay operatingCode='1111111' startDate='2020-12-13' endDate='2021-12-11' />
  <specialService type='exclude' singleDate='2020-12-24' />
  <specialService type='exclude' singleDate='2020-12-31' />
  <specialService type='exclude' singleDate='2021-07-09' />
  <specialService type='exclude' singleDate='2021-07-16' />
  <specialService type='exclude' singleDate='2021-07-23' />
</operatingPeriod>
<operatingPeriod id='opp_5' name='B vS' description='verkehrt Samstag, nicht an Feiertagen' timetablePeriodRef='ttp_2020_21' bitMask='00000100001'
  <operatingDay operatingCode='0000010' onRequest='true' startDate='2020-12-13' endDate='2021-12-11'>
    <operatingDayDeviance operatingCode='1111110' holidayOffset='-1' ranking='2' />
    <operatingDayDeviance operatingCode='0000000' holidayOffset='0' ranking='1' />
  </operatingDay>
</operatingPeriod>
<operatingPeriod id='opp_6' name='B Mo-Fr[nS]' description='verkehrt Dienstag-Freitag sowie an Feiertagen, die auf Montag-Freitag fallen' timetablePeriodRef='ttp_2020_21'
  <operatingDay operatingCode='0111100' onRequest='true' startDate='2020-12-13' endDate='2021-12-11'>
    <operatingDayDeviance operatingCode='0000001' holidayOffset='1' ranking='2' />
    <operatingDayDeviance operatingCode='1111100' holidayOffset='0' ranking='1' />
  </operatingDay>
</operatingPeriod>

```

verkehrliche Züge:



betriebliche Züge:



## Züge und Zugteile

Zur Abbildung des Flügelns, des Verstärkens, von Kurswagen und Mehrfachbespannung werden in RailML die Züge in kleinste („unteilbare“) Einheiten konstanter Eigenschaften zerlegt, die als *Zugteile* bezeichnet werden.

km	Zug	VL 83107	VL 83507
0,0	Zwickau (Sachsen) Hbf.	7.16	...
13,9	Voigtsgrün		...
22,7	Lengendorf (Vogtland)	7.38	...
28,3	Rodewisch	7.44	...
30,4	Auerbach (Vogtl) unt. Hp.	7.47	...
35,3	Falkenstein (Vogtland)	7.52	...
-1,3	Plauen (Vogtland) ob. Bf.	7.18	...
13,3	Herlasgrün	7.32	...
24,9	Eich (Sachsen)	x 7.46	...
30,6	Auerbach (Vogtl) ob. Hp.	x 7.52	...
31,9	Auerbach (Vogtland)	x 7.53	...
35,3	Falkenstein (Vogtland)	7.58	...
35,3	Falkenstein (Vogtland)	8.02	8.02
42,1	Grünbach	x 8.09	x 8.09
45,6	Muldenberg	x 8.12	x 8.12
47,1	Muldenberg Floßplatz	x 8.14	x 8.14
52,7	Schöneck (Vogtland)	8.21	8.21
53,6	Schöneck (V) Ferienpark	x 8.22	x 8.22
58,0	Zwotental	8.27	8.27
58,0	Zwotental	8.32	...
61,5	Zwota-Zeichenbach	x 8.37	...
63,8	Zwota	x 8.40	...
66,3	Klingenthal	8.43	...
69,5	Kraslice-Pod vlekem	x 8.47	...
70,7	Kraslice	8.49	...
58,0	Zwotental	...	8.34
61,0	Gunzen	...	x 8.37
67,2	Siebenbrunn	...	x 8.47
70,9	Adorf (Vogtland)	...	8.52
	nach		

verkehrlicher Zug **83107**  
 verkehrlicher Zug **83507**  
 betrieblicher Zug **83107**  
 betrieblicher Zug **83507**  
 vereinigtter Abschnitt:  
 betrieblicher Zug **83107**  
 (2teilig)  
 betrieblicher Zug **83107**  
 betrieblicher Zug **83907**

2 verk. Züge  
 3 betr. Züge  
 Σ 6 Zugteile

## Betriebliche und verkehrliche Züge

- Zugteile werden sowohl zu **betrieblichen** als auch zu **verkehrlichen** Zügen zusammengesetzt.
- Betriebliche Züge repräsentieren die Sichtweise eines Eisenbahn-(infrastruktur)unternehmens (EIU) auf die Züge. Betrieblich gibt es im vereinigten Abschnitt nur *einen* Zug.
- Verkehrliche Züge repräsentieren die Sichtweise eines Reisenden auf die Züge, welcher entsprechend der umsteigefreien Fahrtmöglichkeiten im gemeinsamen Abschnitt *zwei* (vereinigte) Züge sieht.
- Die im Tabellenfahrplan im Spaltenkopf und in anderen verkehrlichen Auskunftsmedien angegebene Nummer ist daher nicht identisch mit der eigentlichen betrieblichen Zugnummer.  
(Der Begriff *Zugnummer* wird hier nicht immer „sauber“ verwendet.)
- Aus dem Attribut *trainNumber* der betrieblichen Züge ist erkennbar, unter welcher (offiziellen) Zugnummer der Zug beim EIU geführt wird.
- Die betrieblichen und verkehrlichen Züge können am Attribut *type* (= *operational* oder *commercial*) unterschieden werden.

## Verbindung von Zügen und Zugteilen

- Die eigentlichen Zuginformationen und -eigenschaften wie Zeiten, Fahrzeuge usw. sind an den *Zugteilen* zu finden

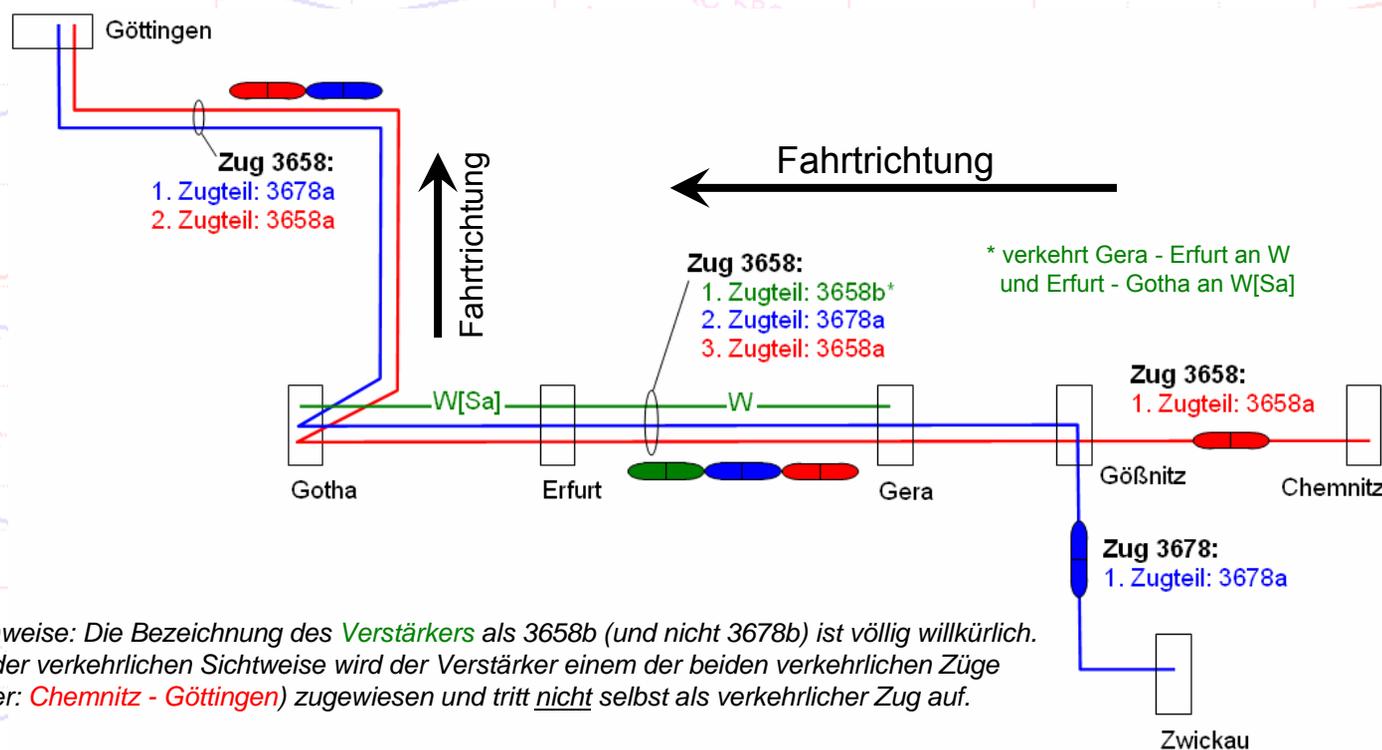
```
<trainPart id=' tp_95001_DBW-DZ' trainNumber=' 95001' categoryRef=' cat_OBE' >
  <formatonTI formatonRef=' fmt_0' />
  <operatingPeriodRef ref=' opp_1' />
  <ocpsTT>
    <ocpTT ocpRef=' ocp_DBW' trackInfo=' 6' ocpType=' begin' >
      <times scope=' scheduled' departure=' 07: 45: 31' /> ...
    </ocpTT>
    ...
    <ocpTT ocpRef=' ocp_DEB' trackInfo=' 1' ocpType=' stop' >
      <times scope=' scheduled' arrival=' 08: 14: 47' departure=' 08: 15: 17' />
    </ocpTT>
    ...
    <ocpTT ocpRef=' ocp_DZ' ocpType=' end' >
      <times scope=' scheduled' arrival=' 08: 40: 40' /> ...
    </ocpTT>
  </ocpsTT>
```

```
<train id=' tro_95001' type=' operational' trainNumber=' 95001' >
  <trainPartSequence sequence=' 1' >
    <trainPartRef ref=' tp_95001_DH-DBW' position=' 1' />
    <trainPartRef ref=' tp_20201_DH-DBW' position=' 2' />
  </trainPartSequence>
  <trainPartSequence sequence=' 2' >
    <trainPartRef ref=' tp_95001_DBW-DZ' position=' 1' />
  </trainPartSequence>
</train>
```

```
<train id=' trc_95001' name=' 95001' type=' commercial' >
  <trainPartSequence sequence=' 1' >
    <trainPartRef ref=' tp_95001_DH-DBW' position=' 1' />
  </trainPartSequence>
  <trainPartSequence sequence=' 2' >
    <trainPartRef ref=' tp_95001_DBW-DZ' position=' 1' />
  </trainPartSequence>
</train>
```

## Beispiel: Kombination aus Flügeln, Verstärken und Fahrtrichtungswechsel (1/2)

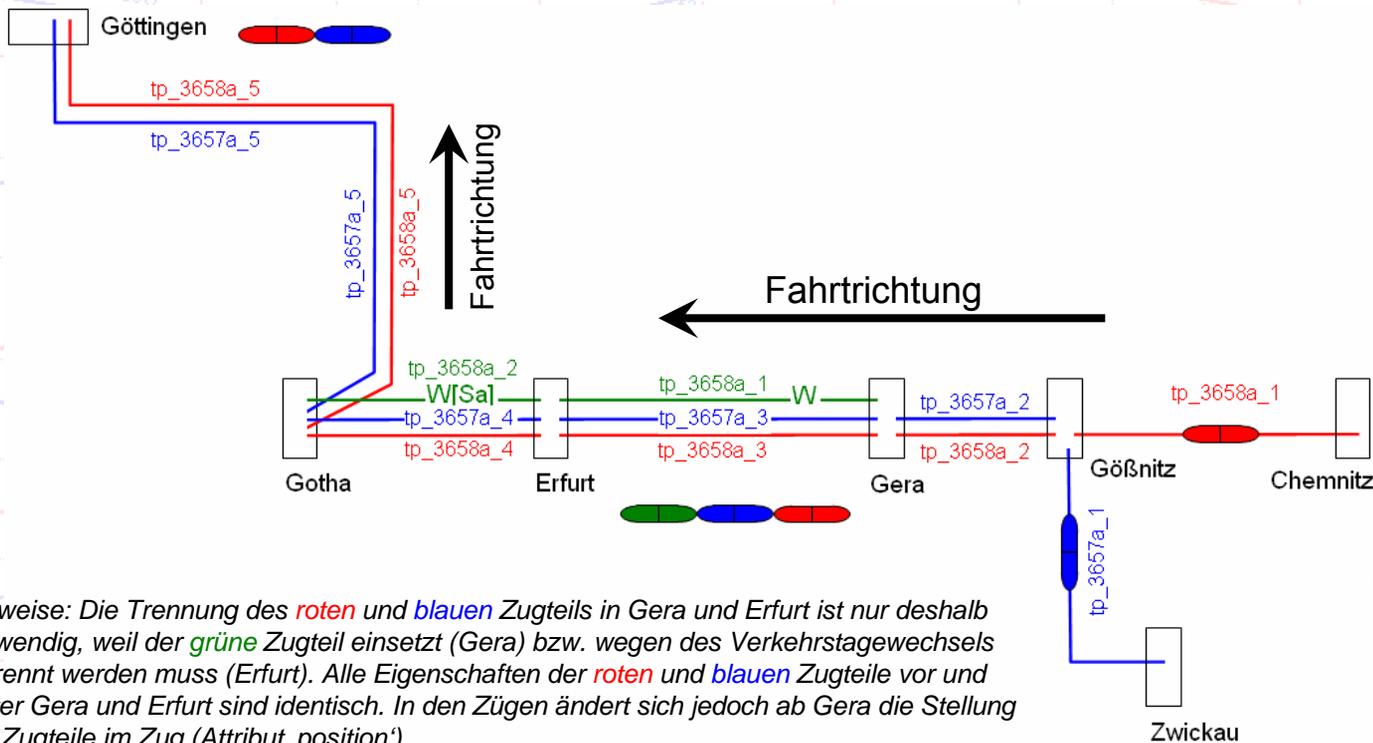
Situation aus zwei verkehrlichen Zügen (Chemnitz - Göttingen und Zwickau - Göttingen) und einem Verstärker (Gera - Gotha), die zwei betriebliche Züge bilden (3658 Chemnitz - Göttingen und 3678 Zwickau - Glauchau)



Hinweise: Die Bezeichnung des Verstärkers als 3658b (und nicht 3678b) ist völlig willkürlich. In der verkehrlichen Sichtweise wird der Verstärker einem der beiden verkehrlichen Züge (hier: Chemnitz - Göttingen) zugewiesen und tritt nicht selbst als verkehrlicher Zug auf.

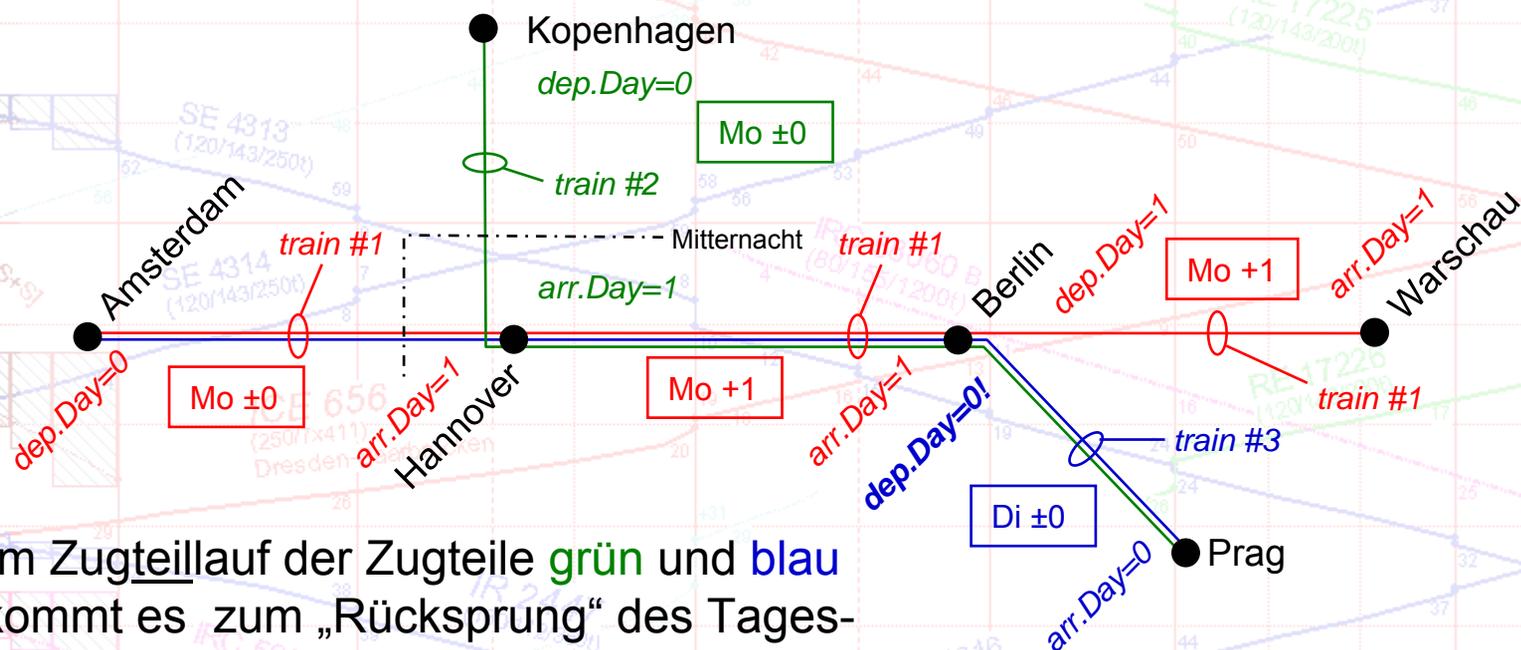
## Beispiel: Kombination aus Flügeln, Verstärken und Fahrtrichtungswechsel (2/2)

Abbildung in RailML - mit 12 Zugteilen:



*Hinweise: Die Trennung des roten und blauen Zugteils in Gera und Erfurt ist nur deshalb notwendig, weil der grüne Zugteil einsetzt (Gera) bzw. wegen des Verkehrstageswechsels getrennt werden muss (Erfurt). Alle Eigenschaften der roten und blauen Zugteile vor und hinter Gera und Erfurt sind identisch. In den Zügen ändert sich jedoch ab Gera die Stellung der Zugteile im Zug (Attribut ,position').*

## Kombination aus Zugteil- und Mitternachtsübergängen



- Im Zugteillauf der Zugteile grün und blau kommt es zum „Rücksprung“ des Tagesindex (arrival/departureDay) von 1 auf 0.
- I. d. R. beginnt jeder neue Zug mit depDay=0 zu zählen, weshalb es in der Verkettung der Zugteile zu einem Rücksprung kommen kann.
- Der Rücksprung geht i. d. R. mit einem *scheinbaren* Verkehrstageswechsel einher (operatingPeriod wechselt, effektive Verkehrstage wechseln nicht).

## Weitere Gedanken zu Mitternachtsübergangen

- Die Verkehrstage-Angabe (`operatingPeriod`) ist nie allein interpretierbar, sondern immer nur im Zusammenhang mit dem jeweiligen Tagesindex (`arrival/departureDay`). Um festzustellen, wann ein Zug tatsächlich fährt, muss man z. B. die Bitmaske um die Anzahl Stellen aus Tagesindex verschieben.
- Ob ein effektiver Wechsel der Verkehrstage innerhalb des Zuglaufs stattfindet, kann nur durch Vergleichen der *verschobenen* Bitmasken herausgefunden werden. Ein Wechsel der `operatingPeriod` allein ist nicht zwingend ein effektiver Verkehrstagewechsel.
- Es kann beliebig viele Kombinationen aus Verkehrstage-Angabe (`operatingPeriod`) und Tagesindex (`arrival/departureDay`) geben, die inhaltlich das gleiche bedeuten.
- Das schreibende Programm ist frei in der Wahl der Kombination, diese kann auch „unterwegs“ wechseln. Ein Beginnen mit Tagesindex 0 wird nicht erzwungen. (?)
- Der Tagesindex beginnt i. d. R. bei der ersten *Abfahrt* mit 0 zu zählen. Demzufolge kann -1 vorkommen bei der ersten Ankunft im seltenen Fall, dass der Zug am ersten Bahnhof über Mitternacht steht und zuvor „von außerhalb“ kommt.

## “Verschobene” Bitmaske für Mitternachtsübergang

- Für operatingPeriods, die „nach Mitternacht“ benutzt werden (d. h. die Züge referenzieren, die über Mitternacht gefahren sind), muss vorn ein Bit eingeschoben werden (i. d. R. eine 0):

111 → Mitternachtsübergang (+1) → 0111

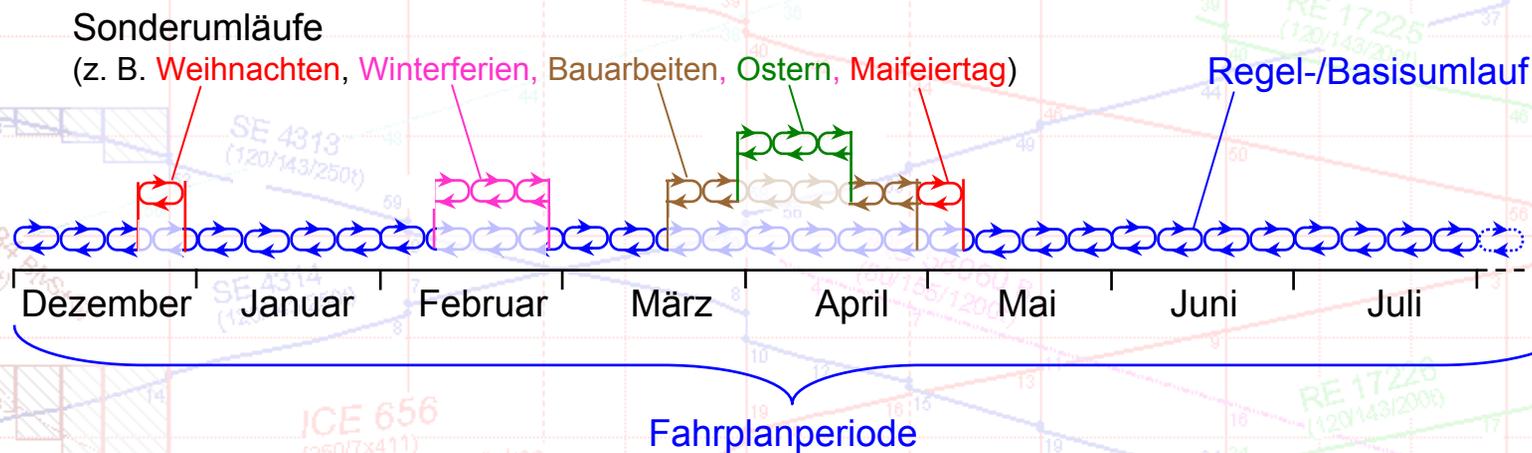
- Damit es keine Bitmasken gibt, die länger sind als die Periode Tage hat, wird das neue Attribut **dayOffset** eingeführt (Stand RailML-Beratung 19.03.2012).
- dayOffset gibt an, um wieviele Stellen=Tage eine Bitmaske verschoben anzuwenden ist. Mit dayOffset>0 ist in Richtung aufsteigenden Datums („nach rechts“) zu verschieben.
- dayOffset ist damit eine Alternative zu arrival/departureDay.
- Mit dayOffset wird ermöglicht, dass arrival/departureDay bei jedem Zug mit 0 zu zählen beginnen können.

```

<operatingPeriods>
  <operatingPeriod id='opp_1' name='täglich' description='verkehrt täglich, auch an Feiertagen' timetablePeriodRef='ttp_
    bitMask='111..111'>
    <operatingDay operatingCode='1111111' startDate='2020-12-13' endDate='2021-12-11' />
  </operatingPeriod>
  <operatingPeriod id='opp_2' name='täglich +1' description='verkehrt täglich nach Mitternacht' timetablePeriodRef='ttp_
    bitMask='111..111' dayOffset='1'>
    <operatingDay operatingCode='1111111' startDate='2020-12-13' endDate='2021-12-11' />
  </operatingPeriod>

```

## Umläufe - Gültigkeit von Umlaufplänen - Umlaufzyklen



- normalerweise bildet ein Regel-/Basisumlauf den (idealisierten) Sollzustand über die gesamte Fahrplanperiode ab
- der Regelumlauf ist „geschlossen“, d. h. kann theoretisch beliebig oft wiederholt werden nach einer Umlaufperiode von nicht mehr als einer Woche
- der Regelumlauf kann durch einen oder mehrere Sonderumläufe zeitweise „überschrieben“ (außer Kraft gesetzt) werden
- Sonderumläufe werden oft erst nachträglich erstellt / exportiert

## Zum Begriff „Umlauf“

- Im Allgemeinen beschreibt ein Umlauf eine Abfolge von Fahrten und sonstigen Einsätzen.
- Ein Umlauf *kann*, muss aber nicht wiederholbar sein (d. h. „umlaufen“ können). Ein Umlauf muss nur dann wiederholbar sein, wenn seine Gültigkeit länger als seine Zykluszeit ist. (Auf den Regelumlauf trifft das normalerweise immer zu.)
- Wenn ein Umlauf wiederholbar ist, liegt seine Zykluszeit (eine Umlaufperiode) in der Praxis zwischen einem Tag und einer Woche.
- „Klassische“ Umlaufpläne können einen Teilzyklus enthalten, d. h. bilden eine Woche in 1-5 Umlauf Tagen ab, z. B. Mo, Di-Do, Fr, Sa, So.
- Diese Umlauf Tage lassen sich verallgemeinern zu „Arbeitstage nach arbeitsfreien Tagen“ (A<sup>n</sup>C), „Arbeitstage vor arbeitsfreien Tagen“ (A<sup>v</sup>C), „arbeitsfreie Tage vor Arbeitstagen“ (C<sup>v</sup>A) usw. (6 Kombinationen).  
*(relevant für strategische Planungen ohne konkrete Fahrplanperiode)*
- Ein Umlaufplan kann mehrere Umlaufgruppen enthalten.

## Umläufe - Modellierung der Elemente und Abfolge (1)

- Das RailML-Element *rostering* enthält einen Umlaufplan für eine Fahrzeugbaureihe (welche über *vehicleRef* oder *formationRef* referenziert wird).

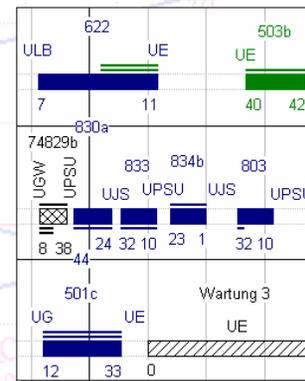
```
<rostering id='rost_642' name='Tfz. BR 642' description='Umlaufplan 642' vehicleRef='veh_DB.642' ...>
```

- Das RailML-Element *blockPart* enthält einen Einsatz (eine Fahrt oder ein sonstiger Einsatz - z. B. Tanken, Wartung, Rangieren, Vorheizen usw.).

```
<blockPart id='bp_95016DBW' begin='22:18:39' startOcpRef='ocp_DBW' mission='timetable',
  trainPartRef='tp_95016_DBW-DH' end='22:51:24' endOcpRef='ocp_DH' runLength='41.010' />
<blockPart id='bp_Tanken_1_Mo-Fr' begin='21:30:00' end='22:00:00' startOcpRef='ocp_DH'
  endOcpRef='ocp_DH', operatingPeriodRef='opp_10' mission='refuel' runLength='0.000' />
```

- Das RailML-Element *block* verkettet *blockParts* zu (willkürlichen) Abfolgen, die aus FBS i. d. R. den üblicherweise dargestellten Blöcken entsprechen.

```
<block id='bl_95016' name='OBE 95016' >
  <blockPartSequence sequence='1' >
    <blockPartRef ref='bp_95016DZ' />
  </blockPartSequence>
  <blockPartSequence sequence='2' >
    <blockPartRef ref='bp_95016DBW' />
  </blockPartSequence>
</block>
<block id='bl_Tanken_1_Mo-Fr' name='Tanken 1 Mo-Fr' >
  <blockPartSequence sequence='1' preProcessingTime='PT0S' postProcessing... >
    <blockPartRef ref='bp_Tanken_1_Mo-Fr' />
  </blockPartSequence>
</block>
```



## Umläufe - Modellierung der Elemente und Abfolge (2)

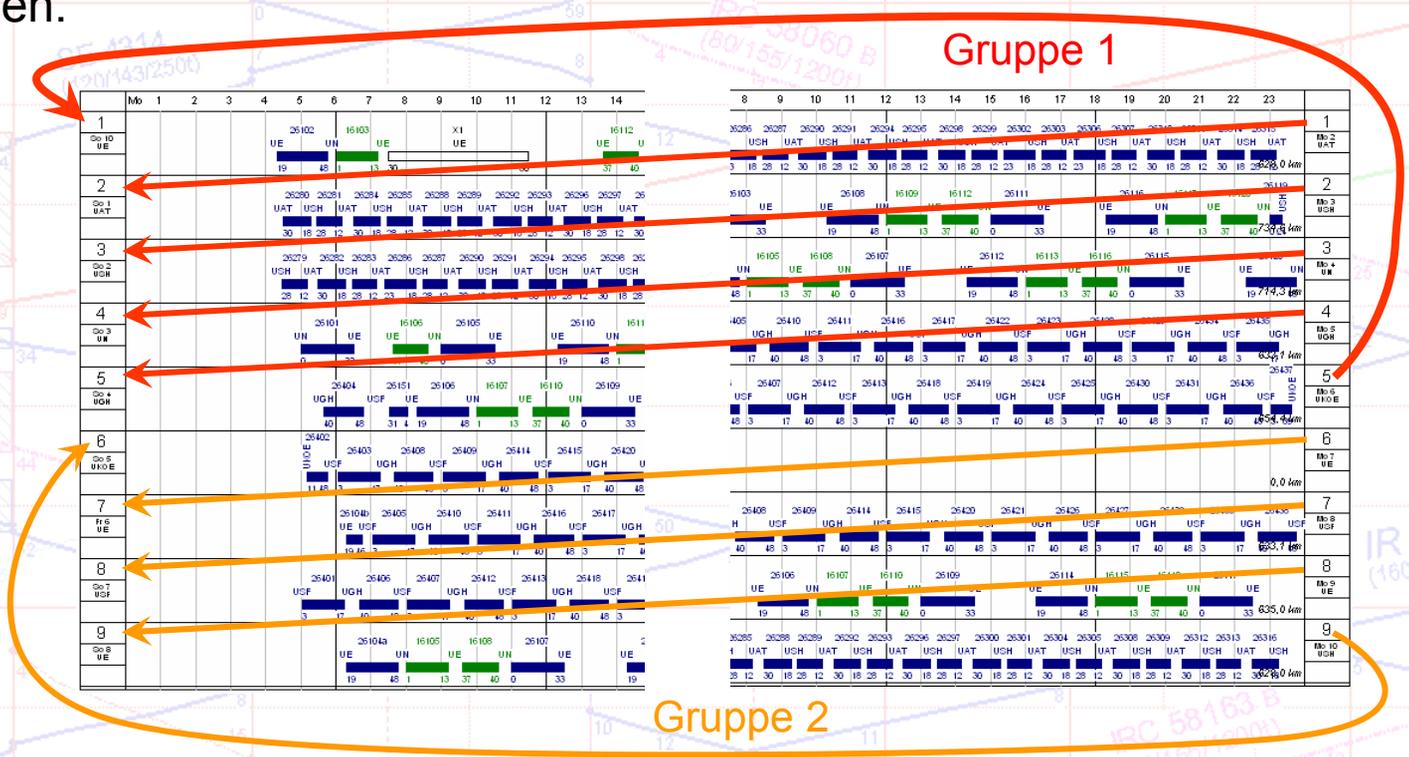
- Das RailML-Element *circulation* definiert die Reihenfolge der „Blöcke“ in Bezug auf Verkehrstage über *operatingPeriodRef*.

```
<circulation blockRef='bl_20016a' operatingPeriodRef='opp_12' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20019b' nextOperatingPeriodRef='opp_12'/>  
<circulation blockRef='bl_20019b' operatingPeriodRef='opp_12' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20000a_W_Sa_' nextOp.Per.Ref='opp_13'/>  
<circulation blockRef='bl_20000a_W_Sa_' operatingPeriodRef='opp_13' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20003b' nextOp.Per.Ref='opp_13'/>  
<circulation blockRef='bl_20003b' operatingPeriodRef='opp_13' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20004a' nextOperatingPeriodRef='opp_13'/>  
<circulation blockRef='bl_20004a' operatingPeriodRef='opp_13' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20007b' nextOperatingPeriodRef='opp_13'/>  
<circulation blockRef='bl_20007b' operatingPeriodRef='opp_13' vehicleIdx='1' groupIdx='1' nextBlockRef='bl_20008a' nextOperatingPeriodRef='opp_13'/>
```

- Es ist je ein *circulation*-Element für jeden Block an jedem Umlauftag enthalten.
- Durch die Attribute *nextBlockRef* und *nextOperatingPeriodRef* jedes *circulation*-Elements ergibt sich die Verkettung der Blöcke und damit die eigentliche Umlaufabfolge. Das Attribut *operatingPeriodRef* jedes *circulation*-Elements gibt über den Inhalt der referenzierten Verkehrsstagerelung (indirekt) an, ob und wie oft ein Umlauftag wiederholt werden muss, um die gesamte Gültigkeit des Umlaufplans abzudecken.

## Umlaufgruppen

- Jede Umlaufgruppe stellt für sich einen eigentlichen Umlauf dar.
- Jede Umlaufgruppe könnte theoretisch in einem eigenen Umlaufplan abgebildet werden. Dies wird in der Praxis aber oft so nicht vorgenommen, weil man u. U. bemüht ist, *alle* Fahrzeuge einer Baureihe in *einem* Umlaufplan darzustellen.



## Eventuelle einsatzfallabhängige Bedingungen

*typische zu bedenkende Mindestanforderungen weiterverarbeitender Programme:*

- Ist eine Linienbezeichnung der Züge/Zugteile notwendig?
- Sind Betriebsstellenummern aller Betriebsstellen / aller relevanter Betriebsstellen / nur der Halte notwendig?
- Müssen Zug-/Zugteil-/Liniennummern usw. rein numerisch sein?
- Müssen zu exportierende Zuggattungen/Produkte in FBS definiert sein? Wie werden Reise-, Güter- und Dienstzüge unterschieden?
- Der Inhalt und Einheit einiger (optionaler) Datenfelder ist in FBS nicht eindeutig spezifiziert (z. B. Koordinaten in WGS84 oder GK).
- Problematik der Mehrfachverwendung von Zugnummern - was ist der Primärschlüssel der Züge im Zielprogramm?
- Problem identischer Betriebsstellen(abkürzungen) an parallelen Strecken - Abbildung als *eine* oder *zwei* Betriebsstellen?
- Muss die Gültigkeit von Umlaufplänen (je Baureihe) disjunkt sein?

## Problematik Mehrfachverwendung von Zugnummern

- Zugnummern sind international bzw. bei großen Eisenbahninfrastrukturunternehmen traditionell auf 5 numerische Stellen begrenzt
- In einigen Fällen ist es verkehrlich und betrieblich-organisatorisch nicht gewünscht (nicht günstig), z. B. bei geringen Differenzen zweier Lagen eines Zuges zwei unterschiedliche Zugnummern zu verwenden.
- Durch konsequentere Einführung von Rechentechnik (elektronische Stellwerke, Bestellung und Abrechnung) müssen früher intuitiv gehandhabte Zusammenhänge heute oft „numerisch sauber“ abgebildet werden (z. B. Abstellfahrten, die eigene Zugnummern benötigen).

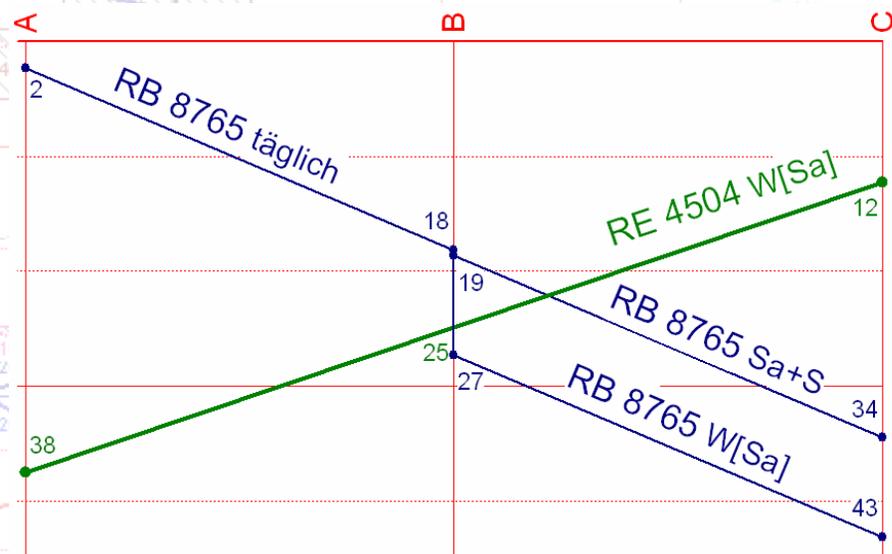
→ Traditionell 5stellige Zugnummern reichen zunehmend nicht mehr aus für den steigenden Bedarf.

→ Zugnummern dürfen unter gewissen Bedingungen mehrfach verwendet werden und werden dazu um eine *Mehrfachzuglaufkennung* ergänzt.

Eine Mehrfachzuglaufkennung besteht aus einer Mehrfachzuglaufart und einem (optionalen) Index. Es gibt die Mehrfachzuglaufarten **Hauptlauf** sowie **Vor-, Zwischen- und Nachnebenlauf**.

## Problematik Mehrfachverwendung von Zugnummern

- typisches Beispiel für „Nachnebenlauf“ -



Abbildungsmöglichkeiten:

**RB 8765 Hauptlauf:**

- verkehrt täglich im Abschnitt A-B
- verkehrt Sa+S im Abschnitt B-C

**RB 8765 Nachnebenlauf:**

- verkehrt nur im Abschnitt B-C und nur an W[Sa]

oder:

**RB 8765 Hauptlauf:**

- verkehrt täglich im Abschnitt A-B
- verkehrt W[Sa] im Abschnitt B-C

**RB 8765 Nachnebenlauf:**

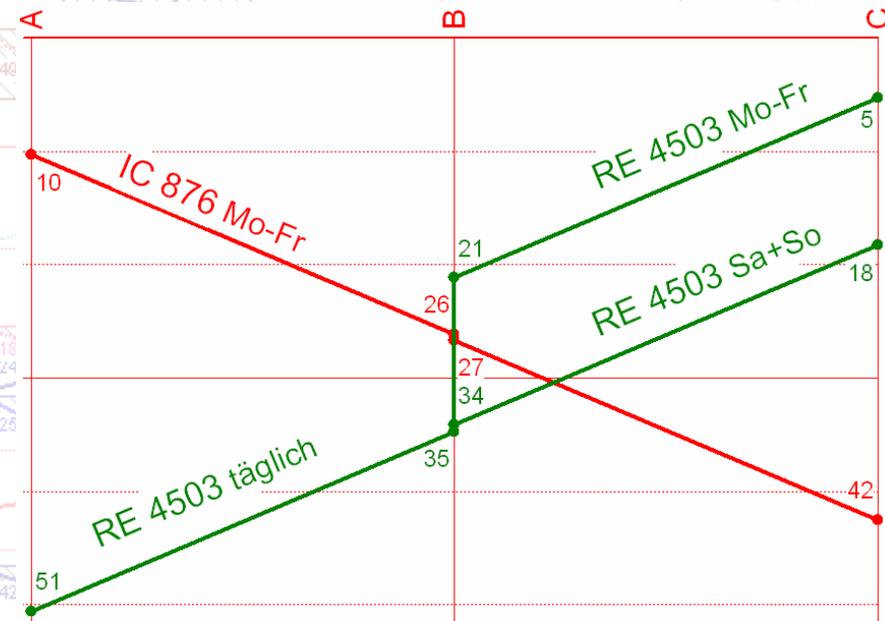
- verkehrt nur im Abschnitt B-C und nur an Sa+S

oder:

- RB 8765 **1. Hauptlauf** verkehrt W[Sa] A-B-C
- RB 8765 **2. Hauptlauf** verkehrt Sa+S A-B-C
- beide Hauptläufe überlagern sich im Abschnitt A-B

## Problematik Mehrfachverwendung von Zugnummern

- typisches Beispiel für „Vornebenlauf“ -



Abbildungsmöglichkeiten:

RE 4503 **Hauptlauf:**

- verkehrt Mo-Fr im Abschnitt C-B
- verkehrt täglich im Abschnitt B-A

RE 4503 **Vornebenlauf:**

- verkehrt nur im Abschnitt C-B und nur an Sa+So

oder:

RE 4503 **Hauptlauf:**

- verkehrt Sa+So im Abschnitt C-B
- verkehrt täglich im Abschnitt B-A

RE 4503 **Vornebenlauf:**

- verkehrt nur im Abschnitt C-B und nur an Mo-Fr

oder:

- RE 4503 **1. Hauptlauf** verkehrt Mo-Fr C-B-A
- RE 4503 **2. Hauptlauf** verkehrt Sa+So C-B-A
- beide Hauptläufe überlagern sich im Abschnitt B-A

## Problematik Mehrfachverwendung von Zugnummern

- Beispiel für „Zwischennebenlauf“ -



Abbildungsmöglichkeiten:

**RE 4503 Hauptlauf:**

- verkehrt täglich im Abschnitt D-C
- verkehrt Mo-Fr im Abschnitt C-B
- verkehrt täglich im Abschnitt B-A

**RE 4503 Zwischennebenlauf:**

- verkehrt nur im Abschnitt C-B und nur an Sa+So

oder:

- Hauptlauf und Zwischennebenlauf im Abschnitt C-B getauscht

oder:

- RE 4503 **1. Hauptlauf** verkehrt Mo-Fr D-A
- RE 4503 **2. Hauptlauf** verkehrt Sa+So D-A
- beide Hauptläufe überlagern sich in den Abschnitten D-C und B-A

Anmerkung:

In allen Beispielen sind vereinfachend nur zeitliche Abweichungen der Mehrfachzugläufe bei gleicher Strecke gezeigt. Ebenso sind jedoch auch räumliche Abweichungen (andere Laufwege / Strecken) möglich.

## Problematik Mehrfachverwendung von Zugnummern

Grundsätzliche Regeln (hier insbesondere im Anwendungsfall DB Netz):

- Bei jeder Zugnummer darf jede Mehrfachzuglaufkennung nur einmal verwendet werden, d. h. Zugnummer und Mehrfachzuglaufkennung müssen innerhalb des Netzes eindeutig sein.
- Die Verkehrstage mehrerer Hauptläufe müssen disjunkt sein.
- Vor- und Zwischennebenläufe müssen an ihrem Endbahnhof mit einem Hauptlauf zusammentreffen. An diesem Bahnhof müssen die Verkehrstage des Nebenlaufs
  - disjunkt sein zu den Verkehrstagen des Hauptlaufs bei dessen Ankunft,
  - eine Untermenge der Verkehrstage des Hauptlaufs bei dessen Abfahrt sein.
- Nach- und Zwischennebenläufe müssen an ihrem Ausgangsbahnhof mit einem Hauptlauf zusammentreffen. An diesem Bahnhof müssen die Verkehrstage des Nebenlaufs
  - disjunkt sein zu den Verkehrstagen des Hauptlaufs bei dessen Abfahrt,
  - eine Untermenge der Verkehrstage des Hauptlaufs bei dessen Ankunft sein.