

---

Subject: Difference between 'Train coupling & sharing' and 'scope'  
Posted by on Tue, 03 Apr 2012 15:16:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

- GERMAN TEXT FOLLOWS BELOW IN RELATIVE BRAKING DISTANCE -

There has been some discussion about the difference between the 'scope' (attribute) of a train in RailML and the principle of "coupling & sharing of trains". Since the discussions took place by E-Mail, I give an excerpt statement here in the forum.

Especially, there has been the question whether both are two possibilities dealing with the same problem. May be these misunderstandings are caused by a mixture of terms in German:

'Train coupling & sharing' = "Flügel von Zügen, Flügelzugbetrieb"  
train.scope=secondaryStart/End = „Startflügel/Zielflügel“ at Deutsche Bahn.

Even though the usage of the term „Flügel“ at Deutsche Bahn (but there only!) both have nothing to do with each other! The terms „Startflügel/Zielflügel“ are expressions of the conservatism of German railwayman since they have been used for a much longer time than 'Train coupling & sharing' is common. They come from and rely to freight traffic only.

The essential difference is:

- The principle of 'Train coupling & sharing' requires the same operating days of two trains.
- "Ergänzungsfahrpläne" (which is the overall DB term to 'train.scope' in RailML) implies disjunctive operating days.

So, this is the sharp contrast! A 'secondaryStart' (Startflügel) would have not one operating day in common with it's 'primary' (Stammzug). To do 'Train coupling & sharing', both trains would need at least one operating day in common!

I think we all know what 'Train coupling & sharing' means so I do not further explain it.

The intention of 'scope' is only to allow that one train (to be exactly: one train number) may have different routes/slots at different operating days. That's it, nothing more! Further explanations are available here:

[www.irfp.de/download/railml\\_beispiel\\_mzl.pdf](http://www.irfp.de/download/railml_beispiel_mzl.pdf)

Dirk.

---

Es gab bisher einige Verwirrungen bezüglich der Bedeutung von „train.scope“ und Zugeilen bzw. Flügeln von Zügen. Da diese Diskussionen per E-Mail stattfanden, gebe ich hier einmal eine kurze Zusammenfassung für Jedermann:

Insbesondere tauchte die Frage auf, ob „train.scope“ nicht nur eine (redundante) Alternative zur Abbildung des Flügelns von Zügen wäre. Dem ist aber nicht so.

Diese Missverständnisse werden vermutlich auch dadurch geschürt, dass die Begrifflichkeiten hier leider nicht sauber getrennt sind: Die Deutsche Bahn spricht bei ihren Ergänzungsfahrplänen auch von „Startflügel“ und „Zielflügel“. Diese haben aber überhaupt nichts mit dem gemeinhin als „Flügeln“ oder „Flügelzugbetrieb“ bezeichneten Phänomen zu tun.

Die Bezeichnungen „Startflügel“ und „Zielflügel“ sind wieder einmal Ausdruck der Konservativität deutscher Eisenbahner. Sie kommen aus dem Güterverkehr und wurden dort schon verwendet lange bevor das „Flügeln“ im Reiseverkehr im heutigen Umfang üblich wurde.

Der entscheidende Unterschied ist:

- Das Flügeln zweier Züge erfordert gemeinsame Verkehrstage beider Züge.
- Zumindest die Deutsche Bahn fordert für ihre „Ergänzungsfahrpläne“ (in RailML: train.scope) disjunkte Verkehrstage.

Das ist also hinsichtlich der Verkehrstage das ganze Gegenteil. Ein „Startflügel“ der DB (in RailML: secondaryStart) hätte garantiert nicht einen gemeinsamen Verkehrstag mit seinem Stammzug im betreffenden Abschnitt. Um zwei Züge zu flügeln, müssten beide aber durchaus wenigstens einen gemeinsamen Verkehrstag haben...

Da sicherlich allen bewusst ist, was mit „Flügeln“ (von Reisezügen) gemeint ist, spare ich mir hier weitere Beispiele.

Der Sinn von „train.scope“ ist hingegen einzig und allein zu erlauben, dass ein Zug (genau: eine Zugnummer) an unterschiedlichen Verkehrstagen unterschiedliche Trassen (Zeiten oder Laufwege) haben darf.. Erläuternde Beispiele hierzu sind zu finden unter:

[www.irfp.de/download/railml\\_beispiel\\_mzl.pdf](http://www.irfp.de/download/railml_beispiel_mzl.pdf)

Dirk.

---