

---

Subject: Re: Extensions to railML for passenger information at stations

Posted by [Milan Wölke](#) on Tue, 27 Aug 2019 15:01:19 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Während unserer letzten timetable Entwickler Telco haben wir folgendes Thema diskutiert:

Quote:

Quote:

\* passengerInfoLanguages: was gilt wenn keine passengerInfoLanguages auf einem StopDescription definiert ist? Ist es den aus dem TrainPart? Ich würde hier gut dokumentieren, wie es zu interpretieren/Vortrittsregeln ist.

Prinzipiell würde ich erwarten, dass wenn keine expliziten Sprachen beim Datenaustausch angegeben wurden, die Systemdefaults des empfangenden Systems gelten. Man könnte aber auch darüber nachdenken, ob das nicht eher für die zweite Art der Abbildung, die ich im Entwurf beschreibe (outputLanguages an annotationRef und announcementRef) spricht. Sollten wir in der Entwicklergruppe noch diskutieren.

Wir haben uns unter Berücksichtigung der oben stehenden Anmerkungen für den zweiten Implementierungsansatz entschieden. D. h. daß für jede annotationRef und jedes announcementRef separate Ausgabesprachen angegeben werden können. Das führt dazu, dass die Beschreibung direkter ist und der Bezug leichter herzustellen ist. Außerdem erlaubt es eine größere Flexibilität, weil so wichtige Information mehrsprachig kommuniziert werden können, wogegen weniger wichtige Infos mit einer reduzierten Mehrsprachigkeit ausgegeben werden können.

-----  
During our last timetableable developers phone conference we discussed the following topic:

Quote:

Quote:

\* passengerInfoLanguages: what applies when no passengerInfoLanguages is defined on a StopDescription? Is it the one from the TrainPart? I would write a good documentation about the way to interpret this and about the priorities.

In principle, I would expect that if no explicit languages were specified during data exchange, the system defaults of the receiving system would apply. One could also consider whether this is more in favor of the second approach (outputLanguages an annotationRef and announcementRef) I describe in the draft. We should discuss this in the developer group.

We have chosen the second implementation approach, taking into account the above comments. This means that separate output languages can be specified for each annotationRef and each announcementRef. This makes the description more direct and the reference easier to make. It also allows greater flexibility because this allows important information to be communicated multilingually, while less important information can be output with reduced multilingualism.

Best regards, Milan

---