
Subject: [railML2] SemCon TT:011 - Consistency of <times> with same @scope
Posted by [Milan Wölke](#) on Mon, 04 Jul 2022 14:44:12 GMT

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

Hi,

in one of our last TT-developer meeting we discussed the proposed semantic constraint TT:011 which is concerned with the consistency of <times> elements of a <trainPart>. So far the constraint stated that all instances of <times> with the same @scope of the same <trainPart> would need to be consistent with each other, meaning that specified times would need to be continuously increasing following the trains path. This at first sound reasonable, however in the discussion we found that there may be a few cases where this actually may not be true. In particular if the @scope values 'earliest', 'latest' and 'published' are used. For example it may be desirable to express that a certain operational requirement constitutes a physical impossibility. For this reason we changed the constraint to exclude the 3 @scope values 'earliest', 'latest' and 'published' from the constraint. The developer group wanted this to be shared with the community in order to get feedback or calls for veto before approving the semantic constrain in this wording. Therefore I would like to ask you to voice your objections if any regarding this change of scope of the semantic constrain until the 1st of August 2022.

The exact wording of the semantic constraint can be reviewed here: [WIKI LINK](#)

Bei einem unserer letzten TT-Entwickler-Treffen haben wir das vorgeschlagene semantische Constraint TT:011 diskutiert, das sich mit der Konsistenz von <times>-Elementen eines <trainPart> befasst. Bisher besagte die Einschränkung, dass alle Instanzen von <times> mit demselben @scope desselben <trainPart> miteinander konsistent sein müssen, was bedeutet, dass die angegebenen Zeiten kontinuierlich ansteigend sein müssen, wenn man dem Weg des Zuges folgt. Das hört sich zunächst vernünftig an, in der Diskussion haben wir jedoch festgestellt, dass es einige Fälle geben kann, in denen dies nicht der Fall ist. Insbesondere, wenn die @scope-Werte 'earliest', 'latest' und 'published' verwendet werden. So kann es beispielsweise wünschenswert sein, auszudrücken, dass eine bestimmte betriebliche Anforderung eine physikalische Unmöglichkeit darstellt.

Aus diesem Grund haben wir das Constraint so geändert, dass die 3 @scope-Werte 'earliest', 'latest' und 'published' aus dem Constraint ausgeschlossen werden. Die Entwicklergruppe wollte, dass dies mit der Community diskutiert wird, um Feedback zu erhalten bzw. ein Vetorecht einzuräumen, bevor die semantische Einschränkung in diesem Wortlaut angenommen wird. Daher möchte ich Sie bitten, Ihre Einwände gegen diese Änderung des Anwendungsbereiches der Semantic Constraint bis zum 1. August 2022 zu äußern, sofern vorhanden.

Der genaue Wortlaut des Semantic Constraints kann hier eingesehen werden: [WIKI LINK](#)

Thanks in advance.

Subject: Re: [railML2] SemCon TT:011 - Consistency of <times> with same @scope
Posted by [David Lichti](#) on Tue, 19 Jul 2022 13:03:58 GMT

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

In our last TT Dev meeting, we revisited this semantic constraint question and finally opted to drop it entirely for several reasons:

earliest and latest are used in early planning stages to model constraints to possible train runs. These requirements may actually be contradictory, indicating that such a train run would not be feasible.

published is used for publication timetables which may differ from actual operational times. Depending on rounding rules and buffer times at different stations, these times may not be strictly increasing in real world examples.

Some tools allow the user to enter non-increasing or inconsistent times. Putting a semantic constraint on scheduled times would prohibit the exchange of these schedules, even for planning purposes with different tools. There are even real world examples, where impossible schedules are used in production, even though it makes operational delays unavoidable.
