
Subject: [railML2] Usage of //ocpTT/times/@scope
Posted by [Milan Wölke](#) on Tue, 14 Jul 2020 13:37:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi community,

the railML's TT developer group has decided to clarify the documentation on <trainPart/ocpsTT/ocpTT/times>@scope (<https://wiki2.railML.org/index.php?title=TT:times>). In that regard we first collected how <times> is used among the members of the developer group. The following list describes the common understanding as of today:

"calculated": Times resulting from the running time simulation; not yet communicated to a party.

"scheduled": Operational times agreed between IM and RU; the arrival and departure time as operationally scheduled

"published": Customer related times communicated from the RU to the end customer (passenger/freight forwarder/...; e.g. an earlier departure time than @scheduled which is communicated to the passengers). Sometimes they could be also less precise than the @scheduled times, e.g. scheduled times are precise down to the second, while published times could be limited to minutes. Also published times are often provided only for the ocpTTs of a train, where a difference between @scheduled and @published occurs (to be confirmed by the community!).

"actual": Describes measured times that the train generated by its run.

So far we do not know of usage for the other enumeration values of <trainPart/ocpsTT/ocpTT/times>@scope ("earliest" and "latest"). Which of the 6 values are you using in your tools and systems? What is the current semantics in place?

You would help us a lot, if you could provide your understanding of <times> in your tools, so we can see if we have a consensus and thus can sharpen the documentation to improve understanding for new people using railML. This also allows us to choose correct and clear modelling when developing the TT section of railML 3.x.

Thanks in advance.

Best regards, Milan

--

Milan Wölke - Timetable scheme coordinator
railML.org (Registry of Associations: VR 5750)
Altplauen 19h; 01187 Dresden; Germany

Subject: Re: [railML2] Usage of //ocpTT/times/@scope
Posted by [Stefan Jugelt](#) on Thu, 16 Jul 2020 14:11:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

for the TAF TSI is a similar attribute "TimingQualifierCode" in use to describe the scope of a given timing. The following 8 values are possible:

PLA = Public Location Arrival
ELA = Earliest Location Arrival
ALA = Actual Location Arrival
LLA = Latest Location Arrival
PLD = Public Location Departure
ELD = Earliest Location Departure
ALD = Actual Location Departure
LLD = Latest Location Departure

It would be useful to align the values proposed for railML with those in use for the TAF TSI.

Best regards,

Stefan Jugelt

Subject: Re: [railML2] Usage of //ocpTT/times/@scope
Posted by [Cathleen Ramson](#) on Thu, 16 Jul 2020 14:53:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

We (IVU) only use only the scope scheduled and for us the meaning of this time is the type of type of time that is used during timetable construction. So the description for scheduled provided by Milan work fine for us.

Subject: Re: [railML2] Usage of //ocpTT/times/@scope
Posted by [Mario Kröplin](#) on Thu, 16 Jul 2020 15:50:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

For the purposes of passenger information, we distinguish the following types of times (inspired by SIRI and Transmodel):

- scheduled (long-term planning)
- aimed (short-term planning)
- expected (forecast for delays)
- actual (detected arrival or departure time)

Currently, our use of railML ends with (very) short-term planning.

This means that we do not transmit expected times and actual times via railML.

The aimed times are the scheduled times of the corresponding train in a separate timetable (short-term planning).

By the way: extensions are used to express cancellations and extra stops explicitly in this separate timetable. A comparison with the annual timetable is therefore only needed to determine the aimed times and the aimed tracks.

The published times are usually rounded scheduled times. But since we also need rounded expected times, we ignore published times and do the rounding on our own.

Subject: Re: [railML2] Usage of //ocpTT/times/@scope
Posted by [Cédric Lavanchy](#) on Fri, 17 Jul 2020 05:08:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

At the SBB, we use the values "published" (commercial aspect for the customer) and "scheduled" (operational aspect used by the IM) as they are described above in the post of Milan.

Subject: Re: [railML2] Usage of //ocpTT/times/@scope
Posted by on Thu, 23 Jul 2020 13:07:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Milan,

here the reply on the usage of <times>@scope in our ex- and imports:

We use

- <times>.scope=scheduled for the operational times, obligatory
- <times>.scope=published for possible different traffic (customer) times, optional
- <times>.arrival+arrivalDay for arrival times, not existing where trains run-through and at the first <ocp> of a train's route
- <times>.departure+departureDay for departure or run-through times, not existing at the last <ocp> of a train's route

So, in our exports, there is at least one and maximum two such elements <times> at each <ocpTT>.

In the same sense, we expect for imports

- that there is at least one element <times>.scope=scheduled at each <ocp> and
- that, in case there are more than one elements <times>, their .scope differs (is disjunct).

- > "scheduled": Operational times agreed between IM and RU; the arrival and
- > departure time as operationally scheduled

No, that is not the documented usage in our railML files. We use "scheduled" for all planning

stages including slot orders (which are naturally not yet agreed by the IM). We understand "scheduled" simply as a synonym for "geplant".

I think that the original intended difference between "calculated" and "scheduled" was that "calculated" only refers to the pure run time calculation and "scheduled" to the actual planning. A planning includes practically always more than the pure (physical) run time. In that sense, "calculated" would be the same as what can be expressed by `<sectionTT>.<runTimes @minimalTime>`.

- > "calculated": Times resulting from the running time simulation; not yet
- > communicated to a party.

I think this could be a bit misleading since the occurrence of these times in a railML file is already a kind of communication to a party. I think the end of the sentence should only be "not yet published". But we could possibly declare "calculated" as deprecated.

-
- > "published": Customer related times communicated from the RU to the end
 - > customer (passenger/freight forwarder/...; e.g an earlier departure time
 - > than @scheduled which is communicated to the passengers).
 - > Sometimes they could be also less precise than the @scheduled times,
 - > e.g. scheduled times are precise down to the second, while published
 - > times could be limited to minutes.

I agree.

-
- > Also published times are often
 - > provided only for the ocpTTs of a train, where a difference between
 - > @scheduled and @published occurs (to be confirmed by the community!).

I even would declare this to a general rule: As long as there are no different "published times", the "scheduled times" are also valid for passenger information.

-
- > So far we do not know of usage for the other enumeration values of
 - > `<trainPart/ocpsTT/ocpTT/times>@scope` ("earliest" and "latest").

I remember that there once was a discussion to use "earliest" and "latest" to define the "slot" which a train could have in it's timetable. But this would be redundant to reserves (supplements etc.) which already have their attributes. I think there should be a Wiki post clarifying that the timetable slot of a train has to be expressed by reserves and not by earliest and latest.

I think the original meaning behind earliest and latest was to express a rather large allowance in slot orders (and only in slot orders), so before the actual timetabling takes place. Such as "I want to arrive between 8 and 10 am." This happens in freight traffic orders, Deutsche Bahn explicitly allows such orders. So, this should be a question for the use case "slot ordering" (@Joachim?). If there is no need for "earliest" and "latest" in this already documented use case, we should declare

it deprecated. Otherwise, we should limit it to that use case.

Best regards,
Dirk.
