Subject: Re: constraints for OperatingPeriod
Posted by Andreas Tanner on Fri, 09 Nov 2012 09:50:47 GMT
View Forum Message <> Reply to Message

Dear all,
as far as railML 2.x is concerned, my suggestion was just to enhance the documentation. The discussion on redundancy has some philosophical traits and therefore, absolute truth cannot be expected to be found.

For railML 3.0, however, I would like to keep the discussion open and let us elaborate a model of validity with a more formally defined semantics that allows to address specific instances of trains from a timetable.

Best, Andreas.

Am 09.11.2012 10:27, schrieb Susanne Wunsch:
> Hi Dirk, Andreas and others,
>
> Dirk Bräuer <dirk.braeuer@irfp.de> writes:
>>> With the redundancy, I do have a problem. It does allow to be
>>> lenient when /writing/ railML, but the costs incur at the import
>>> side. For serious import software, one has to
>>> - extend the customer-specific specification as to disallow
>>> inconsistencies between bitmask and rule
>>> - code a check of the resulting precondition
>>> - add a test case for the software.
>>
>> Please consider:
>>   When writing a RailML file, the software does normally not know for
>> which purpose the RailML file will be used. It has to create a RailML
>> file which is most possibly general.
>>   When reading a RailML file, the software can exactly know the
>> requirements of the target system and therefore can decide which
>> elements and attributes are relevant and which additional rules
>> apply. From my opinion, there always will be additional (semantical)
>> rules which are out of the scope of RailML.
>
> Thank you, Dirk, for the above described explanations.
>
> Andreas, are you convinced by them?
>
>   How about the wanted re-structuring of 'operatingPeriod' for the next
>   major release? Can we drop it? Or do you propose another easier to
>   implement/validate/understand structure?
>
>> Anyway, I agree with you: We would need a possibility to identify
>> "instances" within a period to describe 'actual' information

>> additionally to 'timetable' information - either in 'timetable'
>> schema or somewhere else.
>>
>>>> Additionally, I would prefer to allow an abstract operating period to
>>>> refer to a 'real' operating period. In my opinion, any abstract
>>>> operating period earlier or later becomes real.
>>>
>>> Here I understand that you aim at the mapping of different stages in
>>> the planning process.
>>
>> Yes. I exactly "aimed" on
>>
>>> a concept of unrolling rule-based operating periods onto a calendar
>>
>> So far, this should have been the bit mask.
>
> I filed a ticket for the issue of an "abstract operating period":
>
> http://trac.assembla.com/railML/ticket/187
>
> Kind regards...
> Susanne
>