Subject: Re: constraints for OperatingPeriod
Posted by Andreas Tanner on Mon, 15 Oct 2012 07:35:28 GMT
View Forum Message <> Reply to Message

Am 02.10.2012 17:34, schrieb Dirk Bräuer:
> Dear Andreas,
>
> when the current operatingPeriod type was defined, the aim was to
> explicitly allow a 'describing' definition and a bitMask redundantly.
> The background is that there are many 'describing' expressions for the
> same bitmask. For instance: "Mo-Fr, 01.08-31.08 only" and "[Sa+So], not
> until 31.07 and not from 01.09." lead to the same bitmask.
>

This is two different issues:
1) Redundancy
2) "describing" expressions.
With the latter, I agree, but I intended them to be covered in the
"standard week" case. It does include the specialService etc.
constructions. It should be named more appropriately, though, maybe
"rule based" or something like that.

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.

So please, let us abstain from casually sprinkle some redundancy through
the standard for "easer and clearer reading".

>
> ---
> Concerning a 'third' combination
>> - abstract operating period
>> -- name or code are mandatory
>> -- bitmask, operatingDay, specialService are not allowed.
>
> I would welcome such a possibility. But, from my side we need a
> definition whether different 'abstract operating periods' are
> disjunctive or not. Either we define a matrix or, at least, we define
> that different 'abstract operating periods' _always_ have to be
> disjunctive.

This is an important point. In
http://www.irfp.de/download/railml_doku_beispiele.pdf, you write on the

issue of /keys/ for a train. In another thread in this forum, delay
infos are discussed. There, for instance, one may want to state that
"that instance of the 12345 train with operating period "mo-fr except
holiday" that runs on oct 13, is delayed". So we need even more than an
understanding just of disjoint operating period, namely, a concept of
unrolling rule-based operating periods onto a calendar, and identify
"instances" within a period (which is easy once we have bitmasks
/resulting from unrolling/).

>
> 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. I agree that this is important. I think that this
topic would need more analysis.

Maybe there could be a slot at the upcoming railML conference to discuss
these issues?

Best, --Andreas.