Subject: Re: More detailed 'speed change' definitions
Posted by Susanne Wunsch railML on Fri, 13 Jan 2012 23:06:49 GMT
View Forum Message <> Reply to Message

Hi,

The current thread evolves to some really fundamental discussion about
future infrastructure (track layout) definitions in railML.

Anybody who already uses railML infrastructure or plans to implement
it, please feel personally invited for taking part in this
discussion. Any questions, comments, opinions are highly appreciated.

```
"Carsten Weber" <weber@irfp.de> writes:
>  "Susanne Wunsch" <coord@common.railml.org> schrieb im Newsbeitrag
>  news:bb28vlg7rym.fsf@remi.heep.sax.de...
>>  Hi Carsten, and all who are interested,
>>
>>  "Carsten Weber" <weber@irfp.de> writes:
>>>  "Susanne Wunsch" <coord@common.railml.org> schrieb im Newsbeitrag
>>>  news:bb2ty4a74xn.fsf@remi.heep.sax.de...
>>>>  "Carsten Weber" <weber@irfp.de> writes:
>>
>>  If your track may be operated in both directions - yes. I try to put a
>>  small example here, hope it helps.
>>
>>  <speedChange pos="10" dir="up" vMax="40"/>
>>  <speedChange pos="10" dir="down" vMax="60"/>
>>  <speedChange pos="200" dir="both" vMax="80"/>
>>
>>  pos        10                              200
>>  track dir   ----------------------------------------------------------- >
>>  vMax ->     40->                                80->
>>  vMax <-     <-?                               <-60           <-80
>>
>>  The <speedChange> information defines the maximum speed aspect for the
>>  next track section, means up to the next <speedChange> element in the
>>  track definition direction. A <speedChange> for both directions means,
>>  that the speed aspect at the next track section is the same for both
>>  train running directions. It is _not meant_ to be the same speed aspect
>>  from this point in both directions - that is really misleading!
>>
>  So it becomes tricky to use this structure.
```

Yes. That may be current practice.

Nevertheless I asked some current railML IS user for their current
practice and opinion about the above issue. Most of them appreciated a

clearer structure without the possibility to define speed aspects for
both directions changing at a certain "point". That's really too much
confusing.

If nobody disagrees with good reasons I would file a Track ticket for
"deprecating" the "both" enumeration value from the "dir" attribute in
all "*Change" elements.

New or already used practice is to define seperate elements for each
running direction. The above example would be:

```
<speedChange pos="10"  dir="up" vMax="40"/>
<speedChange pos="200" dir="up" vMax="80"/>
<speedChange pos="2xx" dir="down" vMax="80"/>
<speedChange pos="200" dir="down" vMax="60"/>
```

That means that the semantic for the same XML content changes. That is a
very hard cut that can't be recognized by any parser!

>> The above definition does not include the newly <profileRef>, that comes
>> in addition. It shows the current usage of <speedChange> elements. If
>> your <speedProfile> includes a "direction" attribute, you have to split
>> the <speedChange> elements with current 'dir="both"' attributes.
>>
>> Are there any speed profiles that apply only in one direction?
>
> Yes. For example if you have a railway crossing which is operated by sight
> (without any technical system) you may have speed restrictions only for one
> direction caused by the sight to the track.

```
>           |    |
>  A============B
>     xxxx |    |
```

>
> So you need a speed restriction in direction A to B but not in direction B
> to A. Such cases exist.

Ack.

>> If you have a speed restriction along a bridge, you may define different
>> <speedChange> elements in each direction refering to the same speed
>> profile with different speed aspects. The <speedChange> elements already
>> need the direction attribute. Why do we need to duplicate it?
>>
>
> Because of more clearance.
> In my sight a speedChange should be a child of a speedProfile.
> <speedProfile direction="up" ...>
>   <speedChange position="0" speed="60" .../>

```
>     <speedChange position="5" speed="120" .../>
>     <speedChange position="123" speed="100" .../>
>     <.../>
>  </speedProfile>
>
>  So you can see: the speedChanges if you run the track in one direction in a
>  line.
>  But this structure requires a break in downward compatibility which is not
>  leagl at the moment.
```

The compatibility break is only one side effect of the above mentioned example. Another aspect is that speed profiles are much more general definitions than the detailed speed changes along the tracks.

You may define some speed profiles for different axle weights that are referenced at multiple tracks. If you would define these speed profiles at each track you get heavy redundancies and may "lose track of the big picture".

What's the difference between these two approaches from the timetable point of view?

Let's define a train (part sequence) that goes along three tracks deploying two speed profiles.

```
  (pos)    0   150   345    468   768   1035
  track    t1->      t2->   t3->
  vMax sp1 80->      80->   80->
  vMax sp2    40->  40->   40->  end
```

That means the train (part sequence) deploying both speed profiles is restricted to the following speed profile:

```
  (pos)    0   150   345    468   768   1035
  vMax     80->40->  40->   40->  80->
```

Now let's have a look at your approach (how I understood):

```
  <track id="t1">
    <trackTopology>...(pos=0..345)
    <trackElements>
      <speedProfile id="sp1" description="Default" dir="up"
                   influence="increasing">
        <speedChange id="sc1" pos="0" vMax="80"/>
      </speedProfile>
      <speedProfile id="sp2" description="temporary restriction" dir="up"
                   influence="reducing">
        <speedChange id="sc2" pos="150" vMax="40"/>
      </speedProfile>
    </trackElements>
  </track>
  <track id="t2">
    <trackTopology>...(pos=0..123)
    <trackElements>
      <speedProfile id="sp3" description="Default" dir="up"
                   influence="increasing">
        <speedChange id="sc3" pos="0" vMax="80"/>
      </speedProfile>
      <speedProfile id="sp4" description="temporary restriction" dir="up"
                   influence="reducing">
        <speedChange id="sc4" pos="0" vMax="40"/>
      </speedProfile>
    </trackElements>
  </track>
  <track id="t3">
    <trackTopology>...(pos="0..567)
    <trackElements>
      <speedProfile id="sp5" description="Default" dir="up"
                   influence="increasing">
        <speedChange id="sc5" pos="0" vMax="80"/>
      </speedProfile>
      <speedProfile id="sp6" description="temporary restriction" dir="up"
                   influence="reducing">
        <speedChange id="sc6" pos="0" vMax="40"/>
        <speedChange id="sc7" pos="300" vMax="end"/>
      </speedProfile>
    </trackElements>
  </track>
```

Reference from within the timetable:

```
<trainPartSequence>
  <trainPartRef ref="tp1"/>
  <trainPartRef ref="tp2"/>
  <trainPartRef ref="tp3"/>
  <speedProfileRef ref="sp1"/>
  <speedProfileRef ref="sp2"/>
  <speedProfileRef ref="sp3"/>
  <speedProfileRef ref="sp4"/>
  <speedProfileRef ref="sp5"/>
  <speedProfileRef ref="sp6"/>
</trainPartSequence>

<trainPart id="tp1">
...
  <sectionTT>
```

```
      <trackRef ref="t1" dir="up"/>
  ...
  <trainPart id="tp2">
  ...
    <sectionTT>
      <trackRef ref="t2" dir="up"/>
  ...
  <trainPart id="tp3">
  ...
    <sectionTT>
      <trackRef ref="t3" dir="up"/>
```

Now let's have a look at my approach:

```
  <speedProfile id="sp1" description="Default" influence="increasing"/>
  <speedProfile id="sp2" description="temporary restriction"
                                              influence="reducing"/>
  ...
  <track id="t1">
    <trackTopology>...(pos=0..345)
    <trackElements>
      <speedChange id="sc1" pos="0" dir="up" profileRef="sp1" vMax="80"/>
      <speedChange id="sc2" pos="150" dir="up" profileRef="sp2" vMax="40"/>
    </trackElements>
  </track>
  <track id="t2">
    <trackTopology>...(pos=0..123)
    <trackElements>
      <speedChange id="sc3" pos="0" dir="up" profileRef="sp1" vMax="80"/>
      <speedChange id="sc4" pos="0" dir="up" profileRef="sp2" vMax="40"/>
    </trackElements>
  </track>
  <track id="t3">
    <trackTopology>...(pos="0..567)
    <trackElements>
      <speedChange id="sc5" pos="0" dir="up" profileRef="sp1" vMax="80"/>
      <speedChange id="sc6" pos="0" dir="up" profileRef="sp2" vMax="40"/>
      <speedChange id="sc7" pos="300" dir="up" profileRef="sp2" vMax="end"/>
    </trackElements>
  </track>
```

Suppose the following reference from within the timetable.

```
  <trainPartSequence>
    <trainPartRef ref="tp1"/>
    <trainPartRef ref="tp2"/>
    <trainPartRef ref="tp3"/>
    <speedProfileRef ref="sp1"/>
    <speedProfileRef ref="sp2"/>
  </trainPartSequence>
```

The <trainPart>... definitions are the same as in your approach.

>> In order to calculate the maximum speed aspect for a train along a track
>> section you have to follow the speedChange elements along the track
>> (referenced in the sectionTT) and take only the elements refering to the
>> specified speed profiles. If you get "overlapping" speed aspect you take
>> the highest "increasing" minus the deepest "decreasing" speed aspect.
>>
>> This way you don't need "both". "unknown" would be a fixed value that
>> can't be used for calculating with other overlapping speed aspects.
>>
> Ok. So what should we do if the keyword unknown appears?

Ok. Let's drop the "unknown"! ;-)

```
>>>> >    <xs:element name="tiltingAngle" minOccurs="0" maxOccurs="1">
>>>> >     <xs:annotation>
>>>> >      <xs:documentation>Tilting parameters for which this speed profile
>>>> > is
>>>> > calculated.</xs:documentation>
>>>> >     </xs:annotation>
>>>> >    </xs:element>
>>>>
```

>>>> Good idea. I would spend a child element <tilting> for this and the
>>>> following kind of information. The attribute "angle" could be bound to
>>>> the railML type "tAngleDegQuadrant" for allowing 0° to 90°.
>>>>
>>>> Does a speedProfile covers only one tilting angle value or a range of
>>>> values or some single values?
>>>>
>>> It is a question of definition. In my sight this value means: your
>>> vehicle
>>> has to be able to reach a tilting angle of ... (or more).
>>
>> If your vehicle tilts up to 3.5° it has to pass the curve slower than
>> at an tilting angle of 8°. Maybe you have a speed aspect for a range of
>> tilting angle like:
>>
>>   0°   .. 3.4° 80 km/h
>>   3.5° .. 5.4° 90 km/h
>>   5.5° .. 8°   100 km/h
>>
> Yes. So you would have to define 3 different speed profiles. A speed profile
> without any data to tiliting or a tilting angle of "0", a second one with
> tilting angles higher than 3.5° and a third one for tilting angles higher

> than 5.5°.

If that definition satifies all needs of railML users - let's do it this
way. Prior to fixing this issue to your clarification I would like to
ask for some Swiss, Austrian... experiences with tilting speed profiles.

>>>> >   <xs:element name="tiltingSpeed" minOccurs="0" maxOccurs="1">
>>>> >   <xs:annotation>
>>>> >   <xs:documentation>Tilting parameters for which this speed profile
>>>> > is
>>>> > calculated.</xs:documentation>
>>>> >   </xs:annotation>
>>>> >   </xs:element>
>>>>
>>>> The terminus "speed" may be a bit misleading. I suppose, that is not
>>>> related to the "train speed" but to the "rate/speed of tilting", that
>>>> means the value of tilting degrees per second. I would call this
>>>> attribute "rate". Are there any other ideas?
>>>>
>>>> This attribute may be bound to the railML type "tSpeedDegreesPerSecond".
>>>>
>>>> There is another kind of information related to the tilting that comes
>>>> to my mind: the method of tilting. It could go into an attribute
>>>> "method" that is bound to an enumeration of "active", "passive",
>>>> "rollCompensation", "unknown", "other:anything".
>>>>
>>> I know this. But is it really important here? Do I need to know which way
>>> the vehicle tilts?
>>
>> I thought that it is another type of speed restriction. If you go with
>> "roll compensation" you may increase your speed, if you support "active
>> tilting", you have to switch the system off and go with default speed
>> aspect. This may be a Swiss use case, I mean.
>>
> Yes and no. So you need a speed profile for tilting angles up to 2° (or
> something like that) and you may have a second one for higher tilting
> angles. So if your vehicle is only able to do a tilting up to 2° you can
> only run the "slow" profile. The way you reach the tilting angle is not
> really important. It is more a question of the tilting speed but it is also
> not important there.

Are there any different speed profiles for the same tilting angle
depending on the tilting method?

>>> By the way: We need to define an "endOfSpeedList". So for example if you
>>> have to run slow with a train e.g. above a bridge the allowed speed goes
>>> back to track speed after you left the bridge. So you need an option to
>>> say:

>>> the speed profile ends here. Other speed profiles are now in progress. So
>>> you can use speed profiles along the whole train run but they have one
>>> one
>>> speed limitation at a short bridge or something like this.
>>
>> No, that use case is covered by allowing multiple speed profiles that
>> overlap each other. The software export has to ensure that there are
>> valid speed definitions for all sections of the track.
>>
> This maybe a missunderstanding.
> For example the bridge with there speed limit. So you define a speedProfile
> for a high axleLoad or something like this and say at the beginning of the
> bridge: maxSpeed="60". So what do you write at the end of the bridge where
> the speed raises up to a minimum of all the other speedProfiles?

+1

I really missed this issue. We need some "end of speed restriction"
indication. That may be defined as additional fixed string allowed for
the "vMax" attribute, e.g.

```
  <speedChange id="sc7" pos="300" dir="up" profileRef="sp2" vMax="end"/>
```

Thanks for the ongoing lively discussion.

Kind regards...
Susanne

--
Susanne Wunsch
Schema Coordinator: railML.common