

- > 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.

- > 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.

- >>> Suppose you define some restricted speed aspects for a construction
- >>> area. You may define one <speedProfile> and refer to it from within
- >>> various affected <track>s. Their <speedChange> elements contain this
- >>> reference and the effective direction of the restricted speed aspect.
- >>>
- >> Well a speedChange would be a child element of a speedProfile. But to
- >> bring
- >> this into RailML you need to change structure. The suggestion is based on
- >> the current structure without any change inside.
- >
- > No. A <speedProfile> element goes as a grand-child element to the

```

> <infrastructure> element.
>
> <infrastructure>
> <infraAttrGroups>...
> <tracks>
>   <track id="t1">
>     <trackTopology>...
>     <trackElements>
>       <speedChanges>
>         <speedChange id="sc1" profileRef="sp1" vMax="40"/>
>       </speedChanges>
>     </trackElements>
>   </track>
> </tracks>
> <trackGroups>...
> <operationControlPoints>...
> <controllers>...
> <speedProfiles>
>   <speedProfile id="sp1" name="constructionArea"/>
> </speedProfiles>
> </infrastructure>
>

```

As written above. The solution here is usefull for the current scheme.

```

> 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?

```

>>>> <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

```

```

>>>

```

```

>>> 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).

>

> tilting angle like:

>

>

Yes. So you would have to define 3 different speed profiles. A speed profile without any data to tilting or a tilting angle of "0", a second one with

> I don't know, how the different speed aspects are defined in real

> life. Do you know how it is managed in Germany and Switzerland?

>

So far as I know you can transmitt only one speed profile for tilting vehicles with current known systems.

But it would not be a problem to have several speed lists for different tilting angles.

```
>>>> <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


```

> vMax -> 40-> 80->
> train rel -> head-> end->
> vMax <- <-? <-60 <-80
> train rel <- <-? <-mid
> <-end
>
> The "trainRelation" attribute applies to the starting point of the "speed
> aspect track section".
>
[...]
```

> Thanks for the explanation. Maybe we could shorten the name to "stop"?

>

Maybe, but obligatory stresses the importance of this stop.

```

>>>> <xs:attribute name="vMax" type="rail:tSpeedKmPerHour"
>>>> use="required"/>
>>>> </xs:attributeGroup>
>>>
>>> I would like to rename this attribute to "maxSpeed", but that only comes
>>> with next major release. (Trac ticket #108) ;-)
```

>>>

>> Hm. You told me no abbreviations inside RailML?!

>

> We allow 'max' for 'maximum' and 'min' for 'minimum'. They are not

> misleading and widely known.

>

[no comment]

>> 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 misunderstanding.

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?

>>> Thanks for your suggestions, they are really helpful for further

>>> development. Let's take them up to railML style XML structures for
>>> providing sustainable XML schemas. :-)
>>>
>> Just do it. ;)
>
> I think we are on the right way. ;-)
>
I hope so.

Best regards.

Carsten
