

Dear Susanne and other railML users,

>> - A <baliseGroup> fulfills a certain function, which will be defined  
>> in the parameter "type" with the possible values 'infill', 'signal',  
>> 'technicalFixed' and 'technicalSwitchable'.  
>  
> That above mentioned types are of different kind. A "signal" balise  
> group is always "technicalSwitchable". The "infill" balise group is also  
> always "technicalSwitchable". A "technicalFixed" balise group may be one  
> for odometry or for track conditions ...  
>  
> Thus I would prefer the enumeration values "infill", "signal" and  
> "fixed" for the "type" attribute.

Ok, I modified the trac ticket [1] accordingly.

>> - A <baliseGroup> may have a reference to a <signal>, which will be  
>> defined in the optional parameter "signalRef".  
>  
> On another thread we currently discuss the reference from a signal to  
> its "train protection element". I would prefer to go the same way here.  
>  
> Thus there could be a reference from a signal to its "protecting" balise  
> group with a "baliseGroupRef" attribute in <signal> or <signalAspect>.

Ok. See the trac ticket changes in [1].

>> - The reference from a <baliseGroup> to up to eight single <balise>  
>> elements remains with the sequence of <baliseRef> objects.  
>  
> As mentioned at the beginning of this thread the main idea was to define  
> the up to eight balises inside the balise group not referring them  
> outside. A balise of a balise group cannot be used otherwise by another  
> balise group. Some attributes of the current <balise> element should  
> move to the <baliseGroup> element in order to reduce not-needed  
> redundancy.

Yes, it is useful to group the balise objects in <baliseGroup> than  
grouping references there. However, this will be a major change if you  
do not want to have two (legal) places for defining <balise> elements.  
Therefore, I prefer to change this with railML 3.0.

[1] <https://trac.assembla.com/railML/ticket/174>

Regards

--  
Christian Rahmig  
railML.infrastructure coordinator

---