

Alfonso Gonzalez <alfaeco@gmail.com> writes:

Hello,

- > I am modelling a sample station using the infrastructure schema or
- > Railml, but I am having problems with the balise/baliseGroup element
- > and with the orientation attribute of the connection element.

Please, describe your problem with orientation attribute in a different thread, maybe somebody can help out.

- > In ETCS a balise group is composed by balises, but the baliseGroup
- > element only allow references to already existing balises. Each balise
- > has to be defined individually but that implies that information that
- > belongs to the group has to be repeated for each balise (!?)

>

- > For example:
- > LinkReactionAscending/Descending
- > LinkingAccuracy
- > GroupID
- > CountryID

IMHO in ETCS each balise has to report all these information independent from its grouping. These values have to be the same for the whole balise group, but they are repeated in each balise telegram.

railML follows this view. Despite, some of the above attributes are repeatedly encoded in the "static telegram".

- > Shouldn't it be better to have the balise contained in side the
- > baliseGroup element and define at baliseGroup level the attributes
- > with information related to the balise group?

That would be possible. But there are also "single balises" with the same attributes (without GroupID). How would you structure the data for both cases?

We can't change the code now, earliest change in structure comes with the next major release (up to 3.0?). Up to now, we may add some attribute or element if necessary.

Is there some data missing, which you need only for balise groups but not for balises?

- > If I define a multiple balise group do I have to repeat the informaton
- > which belongs to the group on each single balise?

Yes. It is redundant information which can be used for checking.

- > I my attempt I have modelled a balise group as follows:

```
>
> <ocsElements>
> <signals>
> <signal id="E2" pos="120.000" type="main" dir="up"/>
> </signals>
> <balises>
> <balise id="BG01_0" countryID="357" groupID="01"
> pos="100.000" ndx="0" dir="up"/>
> <balise id="BG01_1" countryID="357" groupID="01"
> pos="100.003" ndx="1" dir="up"/>
> <balise id="BG01_2" countryID="357" groupID="01"
> pos="100.006" ndx="2" dir="up"/>
> <baliseGroup>
> <baliseRef ref="BG01_0"/>
> <baliseRef ref="BG01_1"/>
> <baliseRef ref="BG01_2"/>
> </baliseGroup>
> </balises>
> </ocsElements>
```

Looks like correct railML code. :-)

- > I find it cumbersome to repeat for each balise the attributes
- > ndx, dir, groupID and countryID.

Sorry for the blown up code, we missed to recognize this detail before releasing railML 2.0. Nobody sent us a hint about it. Let's take it to the next major release, I will make a ticket for not losing it.

Before opening the ticket, I would develop the better structure together with you and others, who are interested in. Please post an XML snippet of your idea.

read you ...  
Susanne

Susanne Wunsch  
Schema Coordinator: railML.common