
Subject: Balise / baliseGroups : structure & attributes
Posted by [pierre.simon](#) on Wed, 04 Jul 2012 17:23:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

Conceptually a balise belongs to one and only one balise group. So why not create the parent node <baliseGroup> having a list of 1..8 balises ?

On top of that, attributes like countryID which today is attached to <balise> could be move the parent node <baliseGroup> (+ some additional attributes like

- idBaliseGroup
- its type (in Belgium we have Infill Balise Group / Signal Balise Group / Technical Fixed Balise Group / Technical Switchable Balise Group)
- its reference to the signal (xs:IDRef)

Plus a balise group will have some function(s) and textMessage(s)

For instance we can have:

```
<baliseGroup id="id1" name="d" pos="4" dir="up" absPos="4"
application="ETCS" nidBg="1" nidC="251" type="IBG" signalRef="id12873">
  <balise baliseType="Fixed" nPig="1"/>
  <balise baliseType="Switchable" nPig="0"/>
</baliseGroup>
```

Is it possible to review the model of the <baliseGroup> element in the next railML version ?

[de: Eine Balise gehoert immer zu einer Balisengrppe. Somit koennte man Attribute der Balisen in die Balisengruppe verschieben, um sie nicht in der Balise zu wiederholen. Es koennen bis zu 8 Balisen pro Balisengruppe erscheinen.]

--
----- posted via PHP Headliner -----

Subject: Re: Balise / baliseGroups : structure & attributes
Posted by [Christian Rahmig](#) on Thu, 05 Jul 2012 04:12:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Pierre,

- > Conceptually a balise belongs to one and only one balise group. So why not
- > create the parent node <baliseGroup> having a list of 1..8 balises ?
- > On top of that, attributes like countryID which today is attached to
- > <balise> could be move the parent node <baliseGroup> (+ some additional
- > attributes like
- > - idBaliseGroup
- > - its type (in Belgium we have Infill Balise Group / Signal Balise Group /

- > Technical Fixed Balise Group / Technical Switchable Balise Group)
- > - its reference to the signal (xs:IDRef)

the grouping of <balise> elements within a <baliseGroup> instead of referencing the balises from the balise group is a change that is only possible with a next major release 3.0. However, it is very useful to define the attributes of a <balise> element within a <baliseGroup> as well. And in case we define these attributes being optional, it will be possible to implement them with railML 2.2.

- > Plus a balise group will have some function(s) and textMessage(s)

What kind of functions do you think about? Can you please give an example here?

- > For instance we can have:
- > <baliseGroup id="id1" name="d" pos="4" dir="up" absPos="4"
- > application="ETCS" nidBg="1" nidC="251" type="IBG" signalRef="id12873">
- > <balise baliseType="Fixed" nPig="1"/>
- > <balise baliseType="Switchable" nPig="0"/>
- > </baliseGroup>

Can you please provide some description for the parameters you used within this example? It would be helpful for the readers to understand the actual problem. Thank you very much.

Regards

Christian Rahmig
railML.infrastructure coordinator

Subject: Re: Balise / baliseGroups : structure & attributes
Posted by [Christian Rahmig](#) on Sat, 27 Oct 2012 11:00:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Pierre and other railML users,

- >> Conceptually a balise belongs to one and only one balise group. So why
- >> not
- >> create the parent node <baliseGroup> having a list of 1..8 balises ?
- >> On top of that, attributes like countryID which today is attached to
- >> <balise> could be move the parent node <baliseGroup> (+ some additional
- >> attributes like
- >> - idBaliseGroup
- >> - its type (in Belgium we have Infill Balise Group / Signal Balise
- >> Group /

>> Technical Fixed Balise Group / Technical Switchable Balise Group)
>> - its reference to the signal (xs:IDRef)
>
> the grouping of <balise> elements within a <baliseGroup> instead of
> referencing the balises from the balise group is a change that is only
> possible with a next major release 3.0. However, it is very useful to
> define the attributes of a <balise> element within a <baliseGroup> as
> well. And in case we define these attributes being optional, it will be
> possible to implement them with railML 2.2.

based on your forum entry, I created a trac ticket [1] summarizing the proposed modification of the <baliseGroup> for railML 2.2. Please have a look at it and reply here whether the following attributes fulfill your requirements:

- A <baliseGroup> is modelled as an element with ID and name and therefore inherits the parameters id, name and code.

- A <baliseGroup> fulfills a certain function, which will be defined in the parameter "type" with the possible values 'infill', 'signal', 'technicalFixed' and 'technicalSwitchable'.

- A <baliseGroup> may have a reference to a <signal>, which will be defined in the optional parameter "signalRef".

- The reference from a <baliseGroup> to up to eight single <balise> elements remains with the sequence of <baliseRef> objects.

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

Regards

--

Christian Rahmig
railML.infrastructure coordinator

Subject: Re: Balise / baliseGroups : structure & attributes
Posted by [Susanne Wunsch railML](#) on Wed, 31 Oct 2012 22:09:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Christian, Pierre and other railML users,

Christian Rahmig <coord@infrastructure.railml.org> writes:

>>> Conceptually a balise belongs to one and only one balise group. So why
>>> not
>>> create the parent node <baliseGroup> having a list of 1..8 balises ?
>>> On top of that, attributes like countryID which today is attached to

>>> <balise> could be move the parent node <baliseGroup> (+ some additional
>>> attributes like
>>> - idBaliseGroup
>>> - its type (in Belgium we have Infill Balise Group / Signal Balise
>>> Group /
>>> Technical Fixed Balise Group / Technical Switchable Balise Group)
>>> - its reference to the signal (xs:IDRef)

> - A <baliseGroup> is modelled as an element with ID and name and
> therefore inherits the parameters id, name and code.

+1

> - 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.

> - 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>.

> - 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.

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

Kind regards...
Susanne

--

Susanne Wunsch
Schema Coordinator: railML.common

Subject: Re: Balise / baliseGroups : structure & attributes
Posted by [Christian Rahmig](#) on Fri, 09 Nov 2012 23:15:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

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

Subject: Re: Balise / baliseGroups : structure & attributes
Posted by [Christian Rahmig](#) on Mon, 04 Feb 2013 11:21:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear Susanne and other railML users,

Am 10.11.2012 00:15, schrieb Christian Rahmig:

>>> - 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].

With the latest commit the new optional attribute "baliseGroupRef" has been implemented for the <signal> element. In parallel the attribute "signalRef" has been removed from the <baliseGroup> element.

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

Regards

--

Christian Rahmig
railML.infrastructure coordinator
