

---

Subject: [railML2] Enhancing the <lock> element  
Posted by [christian.rahmig](#) on Mon, 16 Nov 2020 13:32:40 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Dear all,

after the very basic implementation of the signaling element <lock> in railML 2.4 (see [1]), it has been suggested to extend it with upcoming version railML 2.5.

In particular, three modifications are proposed in Trac ticket #428 [2]:

- \* new attribute @positionAtTrack to describe the location of the lock left or right of the track
- \* new attribute @controllerRef to link the lock with the controller from where it is controlled
- \* new attribute @keyStorageRef to link the lock with the OCP where the key is stored in case it is not inserted in the lock.

Further, the question about "types of locks" remains: until further generalization of lock types enabling the introduction of an enumeration attribute, it is suggested to use the free text attribute @description for it.

Are there any objections against that railML 2.5 proposal?

[1] [https://www.railml.org/forum/index.php?t=msg&th=484&goto=1941&#msg\\_1941](https://www.railml.org/forum/index.php?t=msg&th=484&goto=1941&#msg_1941)

[2] <https://trac.railml.org/ticket/428>

Best regards  
Christian

---

---

Subject: Re: [railML2] Enhancing the <lock> element  
Posted by [Torben Brand](#) on Thu, 25 Feb 2021 07:57:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

The Norwegian railway sector appreciates the suggestion for introduction of generic descriptions of a locks properties. We have no objections to the suggested implementation in railML2. We suggest the coordinators double check if the suggested attributes in the previous and in this posting are also contained in [railML3].

We would like to suggest further adding two important attributes of the lock: its type and a reference to what it locks.

With the above suggested attributes, we now have a generic internationally clearly understandable general type description. But we have an UC to unambiguous designate a national type to a national rule. Here we suggest to use the attribute @ruleCode in railML2.5. An example for the rule code would be the Norwegian rule book for locks:

[https://orv.banenor.no/orv/doku.php?id=Brukerveiledninger:personale\\_som\\_skal\\_betjene\\_signalanlegg:kontroll%C3%A5ser\\_og\\_saml%C3%A5ser#lasetyper](https://orv.banenor.no/orv/doku.php?id=Brukerveiledninger:personale_som_skal_betjene_signalanlegg:kontroll%C3%A5ser_og_saml%C3%A5ser#lasetyper)  
Listing and describing the different types and their operational rules for the Norwegian lock types.

For railML3 I leave it to the coordinators to describe the type with either: @ruleCode, @kind, @type, @system or <designator>.

Reference to what the lock actually locks can be done either from the lock or to the lock. In railML2.4nor extensions we have the attribute @lockRef on the element <switch>, <crossing> and the <derailer>. But you could also have a subelement <takesControlOf> with an @ref attribute like in railML3. What does the community prefer?

---

Subject: Re: [railML2] Enhancing the <lock> element  
Posted by [Michael Gruschwitz](#) on Thu, 25 Mar 2021 12:52:32 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Dear all!

Am 25.02.2021 um 08:57 schrieb Torben Brand:  
> Reference to what the lock actually locks can be done either  
> from the lock or to the lock. In railML2.4nor extensions we  
> have the attribute @lockRef on the element <switch>,  
> <crossing> and the <derailer>. But you could also have a  
> subelement <takesControlOf> with an @ref attribute like in  
> railML3. What does the community prefer?

We prefer the solution to have the attribute @lockRef on the element <switch>, <crossing> and the <derailer> as it is implemented in the same manner on the <controllerRef>.

Best regards,

--  
Michael Gruschwitz  
Bahnkonzept Dresden/Germany

---

Subject: Re: [railML2] Enhancing the <lock> element  
Posted by [Jörg von Lingen](#) on Sun, 11 Apr 2021 04:08:35 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thanks for your feedback. We will the attribute @lockRef as suggested.

Best regards,  
Joerg v. Lingen - Interlocking Coordinator  
Am 25.03.2021 um 13:52 schrieb Michael Gruschwitz:  
> Dear all!  
>  
> Am 25.02.2021 um 08:57 schrieb Torben Brand:

>> Reference to what the lock actually locks can be done either  
>> from the lock or to the lock. In railML2.4nor extensions we  
>> have the attribute @lockRef on the element <switch>,  
>> <crossing> and the <derailer>. But you could also have a  
>> subelement <takesControlOf> with an @ref attribute like in  
>> railML3. What does the community prefer?

>  
> We prefer the solution to have the attribute @lockRef on the element <switch>,  
> <crossing> and the <derailer> as it is implemented in the same manner on the  
> <controllerRef>.

>  
> Best regards,

---