

---

Subject: Possession Mgmt Use-Case

Posted by [Stefan Wegele](#) on Fri, 29 Oct 2021 15:14:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Dear all,

in IS-workgroup we collect use-cases for TrafficManagementSystems. One of them is "management of possessions" with the description below.

Here we are interested in feedback on how to organize this in context of RailML: ignore, IS, or a dedicated domain.

Best regards

Stefan

#### Use-Case description

Possession is a takeover of responsibility for a part of railway network from the "train operator" to the PICOP (Person In Charge Of Possession).

The objective of a possession is to ensure safe construction/maintenance works on the railway infrastructure. The safety is ensured by a set of safety measures:

- Temporal speed restrictions around the construction work (including neighbour tracks if needed)
- Closed tracks for most of the trains except the specific once.
- Specific position of the switches (similar to flank protection)

Possession management is safety relevant as any failure could result e. g. in a collision of passenger train with a construction train with > 1000 involved people.

Possession undergoes a specific life cycle (here the default "path"):

- It is planned by the maintenance system, defining elements to be worked on and additional definitions (e. g. used machines) which could influence the required safety measures.
- The safety measures are planned by a signalling specialist.
- Timetable planning department defines time intervals for activation, as well as rules for disturbed case (e. g. let Train 1002 pass if delayed less than 5 minutes).
- Train operator modifies planned time intervals according to the expected traffic situation, e. g. by postponing start of possession.
- When the PICOP and his team arrive at possession area
  - o He safely identifies his location, to prevent activation of wrong possession
  - o Requests the activation of Possession
  - o The train operator verifies,
    - \* That timetable requirements for disturbed case are implemented (train 1002 has passed)
    - \* No unexpected trains are inside of the possession area
  - o Train operator activates the speed restrictions defined in Possession.SafetyMeasures
  - o Train operator puts switches in positions defined in Possession.SafetyMeasures
  - o Train operator verifies all the safety requirements defined in Possession are fulfilled
  - o Train operator notifies the PICOP about the possession activation
  - o PICOP and his team start working
- After PICOP finished the work
  - o He ensures, that his team has left the possession area and it is ready for operations
  - o He requests the train operator to finish possession
  - o Train operator

- \* releases blocked switches
- \* removes temporal speed restrictions
- \* closes the possession

To make the life more complicated, the lifecycle of possessions can vary:

- possession can be stopped and reinstated later
- two possessions can be assigned to one PICOP
- one big possession could be split into two small once, without deactivation/reactivation phase
- PICOP could require to move possessed switches to check their proper function.

---

Subject: Re: Possession Mgmt Use-Case

Posted by [Joerg von Lingen](#) on Tue, 02 Nov 2021 01:24:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Dear all,

we have already some kind of "local transferred responsibility" for a particular area in IL. These are the RestrictedAreas with the types <WorkZone>, <LocalOperationArea> and <ShuntingZone>.

There is the option to introduce a new element <PossessionZone> similar to the ones above for defining the extent of the possession and additional functions.

The other option would be to add required functions/features to the existing element <WorkZone>, which is basically a possession in the traditional interlocking world.

So beside the extent of the possession we would need:

- status and validity dates (start/end) of possession in life cycle
- related speed restrictions
- additional rules for related timetable

Best regards,

Joerg v. Lingen - Rollingstock Coordinator

On 29.10.2021 17:14, Stefan Wegele wrote:

- > Dear all,
- >
- > in IS-workgroup we collect use-cases for
- > TrafficManagementSystems. One of them is "management of
- > possessions" with the description below. Here we are interested in feedback on
- > how to organize this
- > in context of RailML: ignore, IS, or a dedicated domain.
- >
- > Best regards
- > Stefan

- >
- > Use-Case description
- > Possession is a takeover of responsibility for a part of
- > railway network from the "train operator" to the PICOP
- > (Person In Charge Of Possession).
- > The objective of a possession is to ensure safe
- > construction/maintenance works on the railway
- > infrastructure. The safety is ensured by a set of safety
- > measures:
- > - Temporal speed restrictions around the construction work
- > (including neighbour tracks if needed)
- > - Closed tracks for most of the trains except the specific
- > once.
- > - Specific position of the switches (similar to flank
- > protection)
- > Possession management is safety relevant as any failure
- > could result e. g. in a collision of passenger train with a
- > construction train with > 1000 involved people.
- >
- > Possession undergoes a specific life cycle (here the default
- > "path"):
- > - It is planned by the maintenance system, defining elements
- > to be worked on and additional definitions (e. g. used
- > machines) which could influence the required safety
- > measures.
- > - The safety measures are planned by a signalling
- > specialist.
- > - Timetable planning department defines time intervals for
- > activation, as well as rules for disturbed case (e. g. let
- > Train 1002 pass if delayed less than 5 minutes).
- > - Train operator modifies planned time intervals according
- > to the expected traffic situation, e. g. by postponing start
- > of possession.
- > - When the PICOP and his team arrive at possession area
- > o He safely identifies his location, to prevent
- > activation of wrong possession
- > o Requests the activation of Possession
- > o The train operator verifies,
- > \* That timetable requirements for disturbed case are
- > implemented (train 1002 has passed)
- > \* No unexpected trains are inside of the possession
- > area
- > o Train operator activates the speed restrictions defined
- > in Possession.SafetyMeasures
- > o Train operator puts switches in positions defined in
- > Possession.SafetyMeasures
- > o Train operator verifies all the safety requirements
- > defined in Possession are fulfilled

- > o Train operator notifies the PICOP about the possession
- > activation
- > o PICOP and his team start working
- > - After PICOP finished the work o He ensures, that his team has left the
- > possession area
- > and it is ready for operations
- > o He requests the train operator to finish possession
- > o Train operator \* releases blocked switches
- > \* removes temporal speed restrictions
- > \* closes the possession
- >
- > To make the life more complicated, the lifecycle of
- > possessions can vary:
- > - possession can be stopped and reinstated later
- > - two possessions can be assigned to one PICOP
- > - one big possession could be split into two small once,
- > without deactivation/reactivation phase
- > - PICOP could require to move possessed switches to check
- > their proper function.
- >

---

Subject: Re: Possession Mgmt Use-Case  
Posted by [David Lichti](#) on Fri, 14 Jan 2022 08:00:37 GMT  
[View Forum Message](#) <> [Reply to Message](#)

We discussed this in the TT Developer group. While possession management is not a prime concern for all TT users, it is still relevant in some degree to most. For some, it is enough to have a basic object that can be referenced from train schedules. So it should be possible to include possessions without too much details about activities. Others do detailed possession management throughout their entire planning and/or implementation phase.

While it may not be reasonable to integrate all possible life cycle states of all capacity objects into one single, common enumeration, it would definitely be good to use similar wordings and definitions wherever these use cases have similar stages. It should be kept in mind, that the life cycle of a capacity object is not always a linear process. Capacity requests may be cancelled, rejected or amended, resulting in an iterative process.

for the TT Dev Group

David Lichti

---

Subject: Re: Possession Mgmt Use-Case  
Posted by [Thomas Langkamm](#) on Wed, 19 Jan 2022 10:00:04 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Maybe this should be done in a way that allows a generic support of workflows. I can see two different ways to categorize this:

Ownership: We assign something to a person or a group that is responsible.

A track plan (infrastructure) detailing changes/construction work could be assigned to a person or organization for review/corrections.

In operations, certain regions of the network may be assigned to operators or workstations (time dependent, maybe one interlocking is operated by train controllers during the day but remotely from another interlocking during the night).

A weekly plan for a duty roster could be assigned to the workers council for clearance.

Planning stage: Define a level of completion. For example, in timetable planning we may have stages

Weekly planning (long term). This is a weekly roster for all train journeys that serves as template for a timetable period (a year).

Date based planning (a few days ahead). Timetable for one specific date including deviations, but still managed by timetable planners.

Today's timetable (constantly changing due to external circumstances).

My suggestion would be to create some basic framework for this (both planning stages and responsibility), while the different subschemas could use different values. For example, having a "today" planning stage makes a lot of sense for timetable planning but may be less useful for infrastructure and interlocking. There are some commonalities though, for example in planning stage we have typically long term/mid term/short term. Perhaps this is enough so that we don't need to define schema specific values?