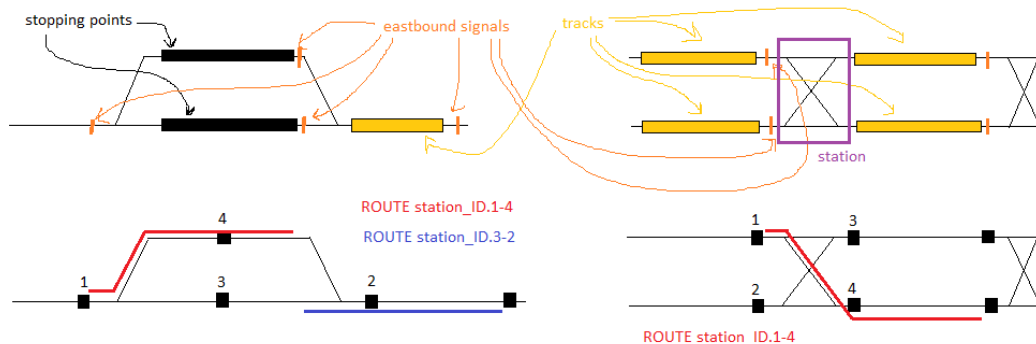


## Instances description

### Status.json

Here you can find the current status of trains (Note: in this file there are fewer train IDs than in the other files – you should take into account the information on these trains only, and skip all the other train IDs)

- “now”: current time
- For each train:
  - Train ID
  - Current position, made of TRACK or STATION\_STOPPING\_POINT or STATION\_ROUTE or a sequence of these, with the entry time (time\_in)
  - Train\_mode (all AUTO)
  - Train\_hold\_main: a particular train which should always keep the main, unless if it is absolutely necessary to be rerouted
  - Current\_length: length of the train
  - Train\_category: category, it is needed for trains’ priority
- Lined routes: for each train, the set of already issued lined routes. This means that the resources are already reserved for the train, which is the first one to use them. Notice that the lined routes are written as <station\_ID>.<starting\_line\_point>-<ending\_line\_point>. See the examples in the figure:



- Blocks:
  - Start\_time: if the block is of type ROLLING, the start\_time is equal to the “now”. If the block is COUNTDOWN, it has a specific start time previous of “now”
  - Duration: the duration of the block from the start\_time
  - Block\_type: either ROLLING or COUNTDOWN
  - Blocked resources
  - Long\_term: if true, this flag is used to remove the resource from the movement graphs, together with the resources that only lead to the blocked resource (this is already taken into account in the movement graphs given)
- Slowdowns: modifications of the allowed speed
  - ID
  - Start\_time
  - Duration
  - Speed: Maximum allowed speed
  - Description (not relevant)
- dispatcher\_solved\_conflicts: empty (not relevant at the moment)

### TrainInfo

For each train:

- train category
- train priority
- default length (length in status or planned changes in length in the movement graphs win over this length)
- speed: default speed (but actually we use the running times recorded in the movement graphs)
- line\_point\_headway : amount of seconds that the follower trains need to wait before reaching a line point, since the train's **head** traversed it
- followers: list of trains in the same direction
- crossings\_ list of trains in the opposite direction

(I advise you not to rely on this information, as when it was added it seemed to be correct, but later on we found out that the same pair of trains can be followers and crossings along their path)

### LineMovements

The set of feasible line movements (the file without "all" or "orig" is the graph already filtered according to actual activities or blocked resources. I left the other ones only for debug purposes)

For each train, and for each usable TRACK:

- the station\_ID of the station reachable at the end of the track
- the set of reachable tracks
- the information about runtimes: note that there is the cumulative running time for the whole track and also for the individual track circuit, whenever a track is splitted into more than one track circuits.
- Line\_headway: amount of seconds that the follower trains need to wait before reaching a line point, since the train's **head** traversed it
- Min\_cumulative\_runtime: the minimum time needed for the train to reach its destination (either FINAL or INTERMEDIATE or PSEUDO\_FINAL)
- Best\_out\_track\_id: the ID of the track that allows the arrival time of the train to be in line with the min\_cumulative\_runtime
- Correct\_path\_id: the ID of the reachable track that does not require taking a reverse switch (it almost always coincides with the best\_out\_track\_id)
- Min\_reverse\_switches: minimum number of reverse switches that will for sure be traversed to reach the train's destination (both for the current track, and for each one of the reachable tracks)
- Preferred\_out\_track\_id: the preferred track usable to exit from the station\_ID
- Min\_non\_preferred: minimum number of non-preferred routes that will for sure be traversed to reach the train's destination (both for the current track, and for each one of the reachable tracks)
- information about preferred tracks, activities on tracks
- available\_mask: not relevant

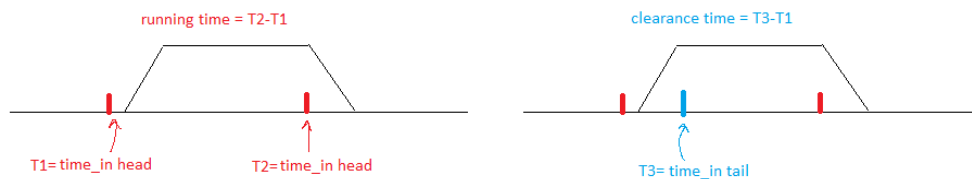
### StationMovements

The set of feasible station movements (also in this case, refer to the file without "all" or "orig")

For each train, and for each usable STATION:

- the list of all "station nodes", which include both the entry and exit tracks, and all the station routes
- for each track:
  - the min\_cumulative\_runtime, as for the LineMovements
  - previous\_edges: the station routes which lead to the EXIT track
  - next\_edges: the station routes which stem from the ENTRY track
  - reachable\_stopping\_points: the stopping points reachable from the ENTRY track

- min\_clearance: the minimum time needed to clear the switches
- available mask: not relevant
- information about preferred tracks, activities on stopping points
- for each station route:
  - correct\_path: either true or false
  - reachable\_stopping\_points: the stopping points reachable from the ENTRY line point
  - default and current min\_cumulative\_runtime
  - runtime infos: running time and clearance time. The **running** time is the time needed by the train's head to go from a signal to the next one. The **clearance** time is the time interval between the train's head traverses a signal and the train's tail traverses the next signal in the opposite direction (=it frees the switch)



- next\_edges: station routes (if the current route ends on a stopping point) or track (if the route ends at a line point)
- previous\_edges: track (if the current route starts from a line point) or station routes (if the current route starts at a stopping point)

### Forbidden Positions

Precalculated relative positions between pair of trains that identify bound-to-deadlock situations. I don't know if it may be useful for you, if not you can avoid to consider this file.

(you can bypass the meta\_index\_map)

- For each Train\_ID1
  - For each Train\_ID2
    - Position of train\_ID1 (from head to tail)
    - List of positions of train\_ID2 which identify bound-to-deadlock situations (from head to tail)

### PlannerLineChoiceSequence

List of decisions taken by the planner during the previous planning cycle. Not sure you need this either.

- Routing\_choice:
  - Train\_ID
  - Routing point: it's a station ID
  - Out\_resource: out track from the station ID above
  - In\_resource: resource from which the station was reached
- Conflict\_choice:
  - Train\_ID1
  - Train\_ID2
  - Loser\_train: train that waits for the other train to depart the meet location
  - Waiting resource: resource where the loser train should wait
  - waiting\_resource\_is\_track (Boolean)