Demonstration scenario 1– International: cross border

Grant Agreement N°: FP7 - SCP1-GA-2011-285243

Project Acronym: **ON-TIME**

Project Title: **O**ptimal **N**etworks for **T**rain Integration **M**anagement across **E**urope

Funding scheme: Collaborative Project

Project start: 1 November 2011

Project duration: 3 Years

Work package no.: WP8

Deliverable no.: WP08-DEL-002; Rev 0

Status/date of document: Final, 31/10/2014

Due date of document: 31/05/2014

Actual submission date: 31/10/2014

Lead contractor for this document: Ansaldo STS
Genoa, Italy

Project website: www.ontime-project.eu

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	**X**
CO	Confidential, only for members of the consortium (including the Commission Services)	

Revision control / involved partners

Following table gives an overview on elaboration and processed changes of the document:

Revision	Date	Name / Company short name	Changes
1	31/10/2014	Antonio Miano/ASTS, Marco Giaroli/ASTS	First issue

Following project partners have been involved in the elaboration of this document:

Partner No.	Company short name	Involved experts

Executive Summary

The overall aim of the ON-TIME project is to improve railway customer satisfaction through increased capacity and decreased delays both for passengers and freight. This is achieved through new and enhanced methods, processes and algorithms.

A key objective of the project was to demonstrate the research results with real life examples and situations. Locations were selected to encompass the following aspects

1. International cross-border line
2. Long distance intercity and commuter traffic in a mixed scenario.
3. National Nodes
4. Predominantly freight line.

The locations were in different countries in order to show the adaptability and flexibility of the research results.

The demonstrator shows how resulting algorithms (and processes) could be integrated into large systems. The demonstrator is an enhanced HMI (Human Machine Interfaces) using technology from out with the rail industry. The new interface is a **fully touch** interface These are not in themselves innovations, but add to the ease of use of the systems. This state-of-the-art interface allows the operators to be more focussed and productive.

The results are described in a series of three documents for the different scenarios, of which this is the first. In this document the WP8 graphical results from the ON-TIME demonstration scenario 1, Iron Ore, the cross border line are presented.

Table of contents

1 INTRODUCTION 7

2 SCOPE OF THE DOCUMENT 8

3 INNOVATION 8

4 STANDARDIZATION, INTEGRATION, OPTIMIZATION 8

4.1 RailMl standard 8

4.1.1 Demonstration 1: From RailMl to Train Describer 9

4.1.2 Demonstration 2: From RailMl to Train Graph 10

4.1.3 Demonstration 3: a complete “up and running” Train Graph 11

4.2 Integration 13

4.2.1 Demonstration 7: Integration 13

4.3 Optimization 15

4.3.1 Demonstration 8: Optimization 15

5 SIMULATOR 16

5.1 Demonstration 5: Animation with HERMES simulator 17

6 INTERACTIONS 19

6.1 Demonstration 6: Standard interactions 19

REFERENCES 21

Table of figures

Figure 1-1 General Schema 7
Figure 4-1 - IronOre TD created using the unmodified RailMl infrastructure data 9
Figure 4-2 - IronOre TD created using the modified RailMl infrastructure data 10
Figure 4-3 - Iron Ore TG generated using just RailMl data 11
Figure 4-4 - IronOre TG generated using RailMl/RTTP/events data 12
Figure 4-5 - Iron Ore line - Pre TMS optimization TG 12
Figure 4-6 - Iron Ore line - Post TMS optimization TG 13
Figure 4-7 - TG + Path Selection 14
Figure 4-8 - TG + Track Allocation 14
Figure 4-9 - TG + Path Selection + Track Allocation 15
Figure 4-10 - What If Analysis 16
Figure 5-1 - IronOre TD/TG animated by HERMES simulator (Step 1)......................17
Figure 5-2 - IronOre TD/TG animated by HERMES simulator (Step 2)......................17
Figure 5-3 - IronOre TD/TG animated by HERMES simulator (Step 3)......................17
Figure 6-1 - Train list for station 19
Figure 6-2 - "On screen display" train timetable 20

Table of abbreviations

ONT	ONTIME project code on the repository

1 INTRODUCTION

The ON-TIME project will develop new methods and processes to help maximise the available capacity on the European railway network and to decrease overall delays in order to both increase customer satisfaction and ensure that the railway network can continue to provide a dependable, resilient and green alternative to other modes of transport. In the project, specific emphasis will be placed on approaches for alleviating congestion at bottlenecks. Case studies to be considered will include passenger and freight services along European corridors and on long distance main-line networks and urban commuter railways.

The project is divided into 9 work packages; in particular WP3, WP4, WP5 and WP6 provide modules that process/produce data; WP7 is responsible to create a common data interface. Graffica simulator, finally, is the module that acts as the virtual railway field. The general schema is depicted in the next picture.

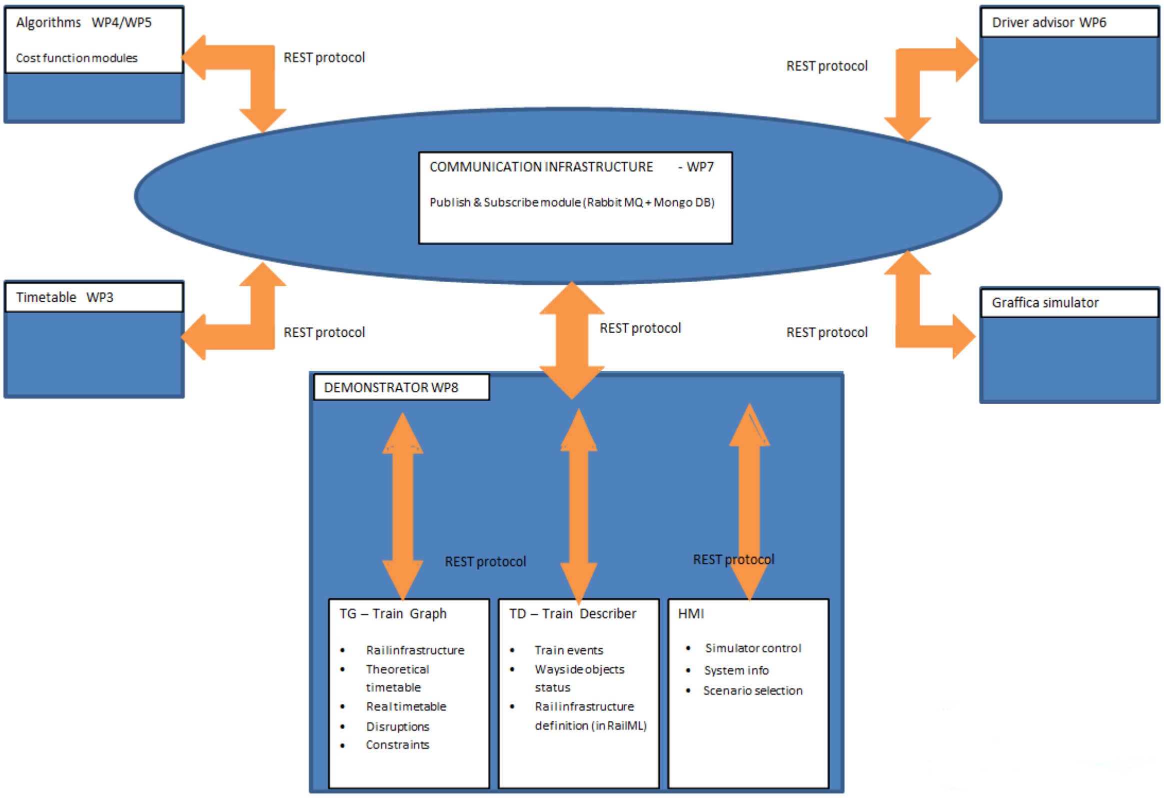

Figure 1-1 General Schema

The complete railway infrastructure can be acquired (requested) from WP7 to the simulator, an optimized and resilient timetable can acquired (requested) from WP3. WP4 and WP5, using infrastructure and timetable are able, by applying the right algorithm, to solve criticalities such as small and big disruption in order to minimize a cost function (total delay for instance). Meanwhile, WP6 is able to inform the driver about the right speed, the traffic flow and bottlenecks if any.

The demonstrator (Work Package 8) collects all the input from all the previous modules and by means of three different graphical interfaces (Train Graph, Train Describer and a general purpose HMI) and is able to give to the final user a complete view of what is going on.

2 SCOPE OF THE DOCUMENT

The demonstrator being mainly a graphical interface, this document will present several screenshots in order to give the reader a full view of the results achieved by this project. Every screenshot will demonstrate a different area of interest in which the project itself reached a particular target.

3 INNOVATION

The key theme of OnTime is innovation and its application. All the results have been achieved keeping in mind that the main target was finding new and effective solutions. The screenshots presented in this document can give a full idea of the proposed innovations but there is something that can't be demonstrated using a simple picture.

This is the case of the **fully touch** approach Ansado STS used to develop the new interface. This result was made possible by the usage of a new Java library, JavaFX, that can be considered the present state-of-the-art in HMI development.
Using the touch functionalities the operators can be faster and more focused on their job with a set of gesture similar to the familiar ones of smartphones.
From a development point of view, using JavaFX is easier for programmers to add new code and to debug a program. A low memory and cpu consumption means workstations less powerful or faster and bigger HMI.
All the previous points lead to sensible cost reductions not only for the developers but also for their customers.

4 STANDARDIZATION, INTEGRATION, OPTIMIZATION

Even though innovation is the central theme, standardization, integration and optimization are three valuable outcomes that are not underestimated. In this chapter, with the assistance of some screenshots, we will expand on these themes as applied to the Iron Ore line demonstration.

4.1 RailMl standard

The railML.org Initiative was founded in early 2002 against the background of the chronic difficulty of connecting different railway IT applications. Its main objective is to enable heterogeneous railway applications to communicate with one another. Today, the connection of various railway software packages is beset with problems. RailMl is the answer to this problem. It can be considered an Xml language, as html for web pages, to model complex railway lines in terms of infrastructures, timetables, rolling stocks, interlockings and to simplify data exchange between railway applications.

RAILML offers:

- Full integration with all Infrastructure Managers that adopt this standard.
- A common method to refer to the wayside objects (signals, switches, etc).
- The contact point between geo mapped and schematic lines representation.

The HMI we developed is one of the first in Europe that is able to read this new data model in a transparent way. No configuration modifications are needed other than those in RailMl. All the mimic interfaces come directly from the data model.

The use of RailMl leads to a dramatic reduction of the configuration phase and reduces the probability of mistakes. This new approach is invaluable also in terms of the service level given to the customer. Fast prototyping, effective information sharing, quick data updating are added values to railwaymen.

4.1.1 Demonstration 1: From RailMl to Train Describer

RailMl contains all the information needed to draw a Train Describer using both the geo mapped and the schematic data. As the latter is not always present in the RailMl flow, Ansaldo developed a tool to convert manually the geo mapped info into a more schematic data set.

The next screenshot shows the TD of the Iron Ore line obtained directly using the RailMl data coming from the simulator are without any modification.

Figure 4-1 - IronOre TD created using unmodified RailMl infrastructure data

Using the tool Ansaldo developed to improve the data set is it possible to obtain something more schematic as shown in the next picture.

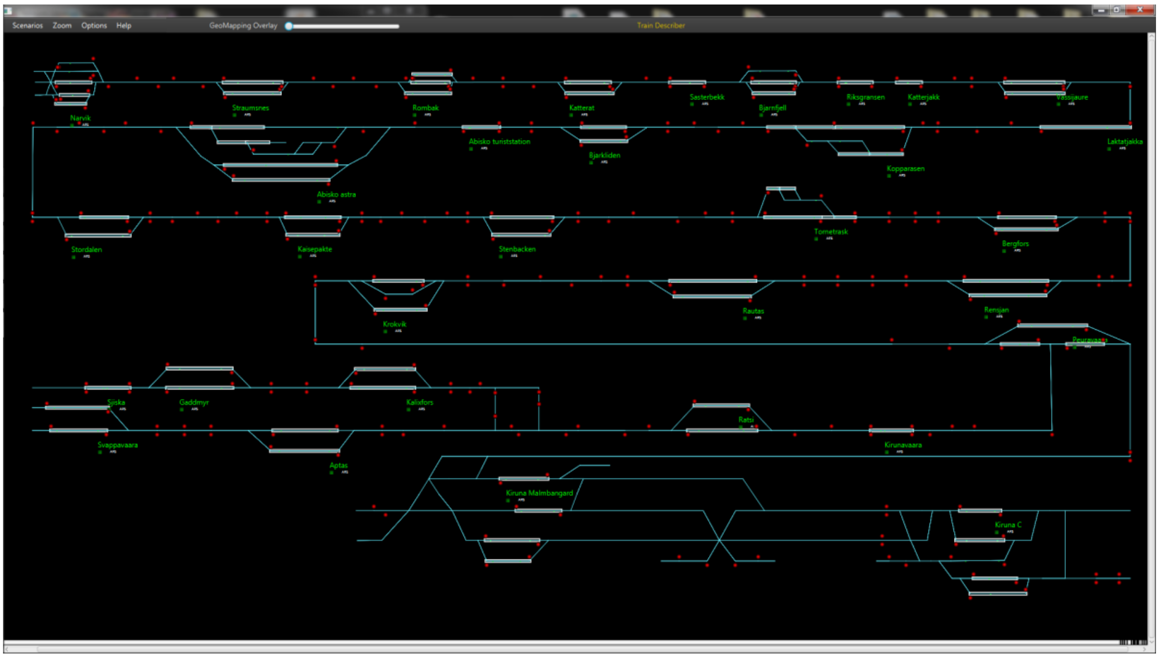

Figure 4-2 - IronOre TD created using modified RailMl infrastructure data

4.1.2 Demonstration 2: From RailMl to Train Graph

RailMl contains also all the information needed to draw a Train Graph. TG needs four different sets of data and the first two are RailMl compliant.

- railway infrastructure. TG shows an ordered list of stations along one axis. Often the space between the stations is proportional to the real distance between them. Station names and distances are inside the infrastructure data.
- timetable. It is the basis to draw all the trains tracks (theoretical time). A section of RailMl contains all the timetables needed.

Infrastructure and timetable are sufficient to draw a "static" TG, without any animation. As shown in the next picture, at this stage the TG only displays the theoretical time of all trains and the stations of the line.

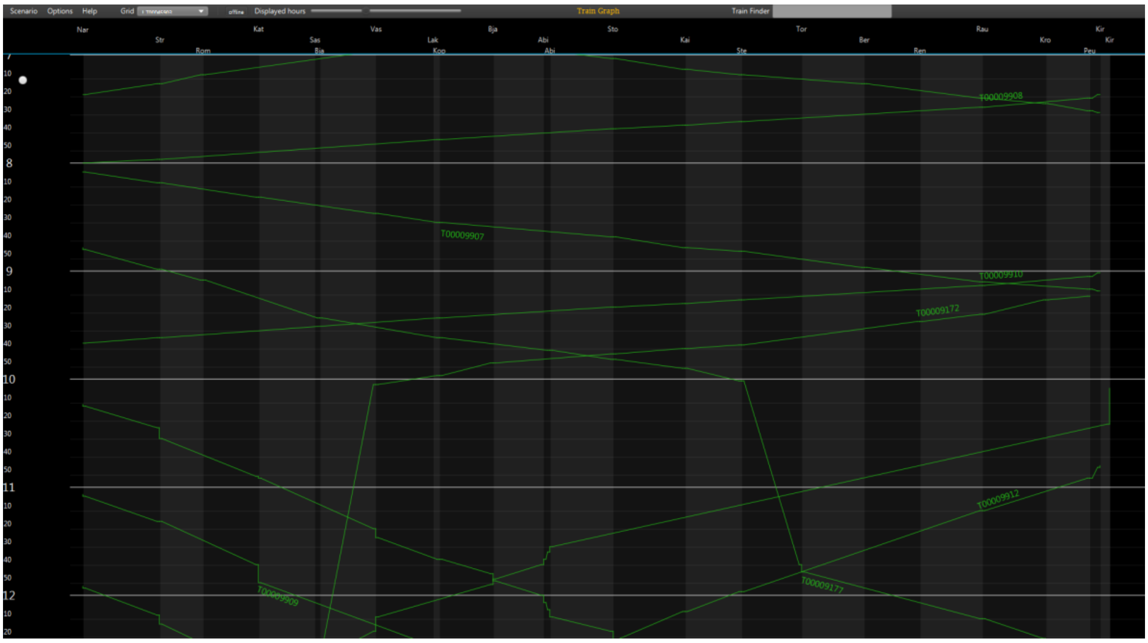

Figure 4-3 - Iron Ore TG generated using just RailMl data

4.1.3 Demonstration 3: a complete "up and running" Train Graph

To draw a complete TG, two more data sets are required.

- RTTP, Real Time Traffic Plan. This is the forecast for every train and comes from the prediction modules: TMS+ROMA or TMS+RECIFE (WP4/WP5). RTTP is not strictly related to RailMl, in fact it is an OnTime internal formalism very similar to RailMl but it doesn't belong to this standard.
  One of the project outcomes is the proposed extension to RailML

- Real events. When a train arrives/departs to/from a station, the corresponding track on TG is highlighted (thickened). This is possible because the simulator generates all the events needed to follows trains along their journey and TG can receive them through the WP7 infrastructure.

Using this information it is possible to "animate" the TG.

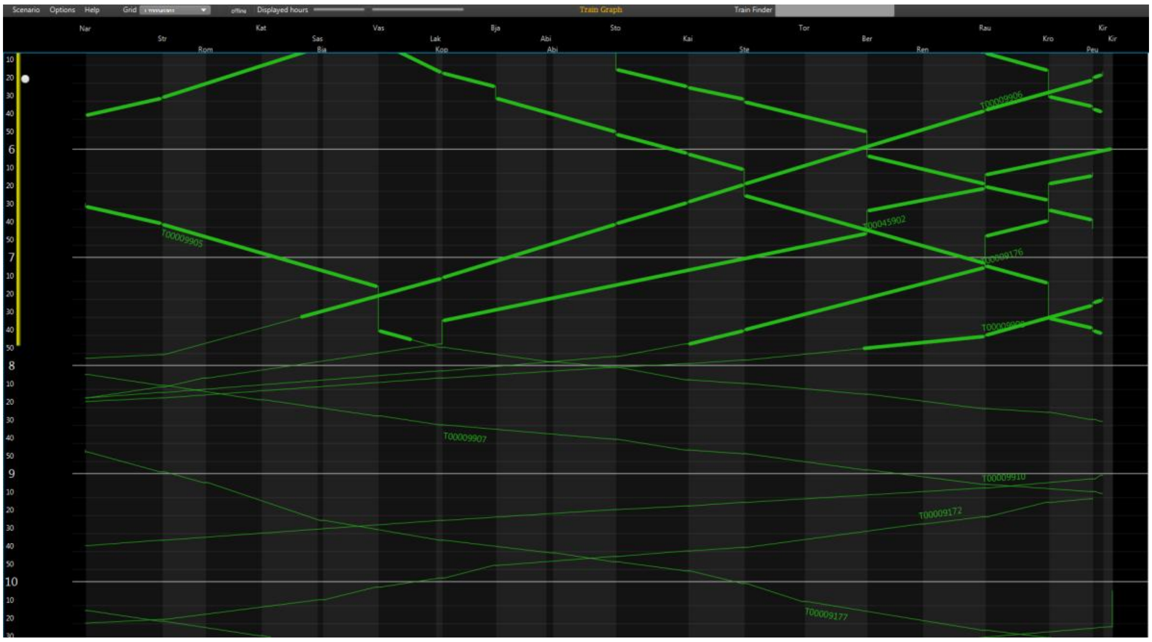

Figure 4-4 - IronOre TG generated using RailMl/RTTP/events data

The following two pictures show how the TMS optimization acts on the railway capacity. This effect is clearly visible in the TG.

Figure 4-5 - Iron Ore line - Pre TMS optimization TG

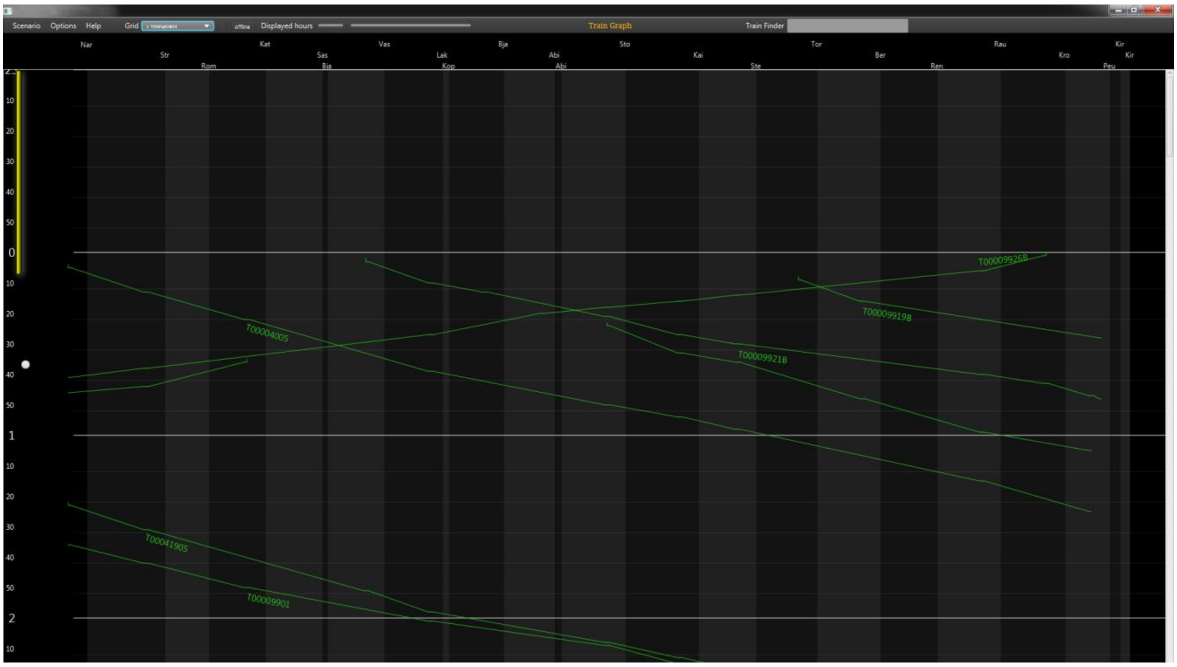

Figure 4-6 - Iron Ore line - Post TMS optimization TG

4.2 Integration

Having a full functional workstation is an essential target to achieve but it is not enough. Many interfaces, even if functional and fast, have the problem of being too complicated: the info needed is somewhere hidden in the system and a considerable effort is required to access it. This sometimes happens because on the same screen there are too many HMI that don't "cooperate" with one another.

To improve this situation, the TG we developed combines three different HMIs: TG (of course) Path Selection and Track Allocation. Adopting this approach, a lot of useful data, usually confined in some dark area, is accessible at once without the need to find it.

4.2.1 Demonstration 7: Integration

For a Train Controller having the possibility to change the path of a train is essential. This functionality can be simplified if the Path Selection is integrated into the TG as shown in the next picture.

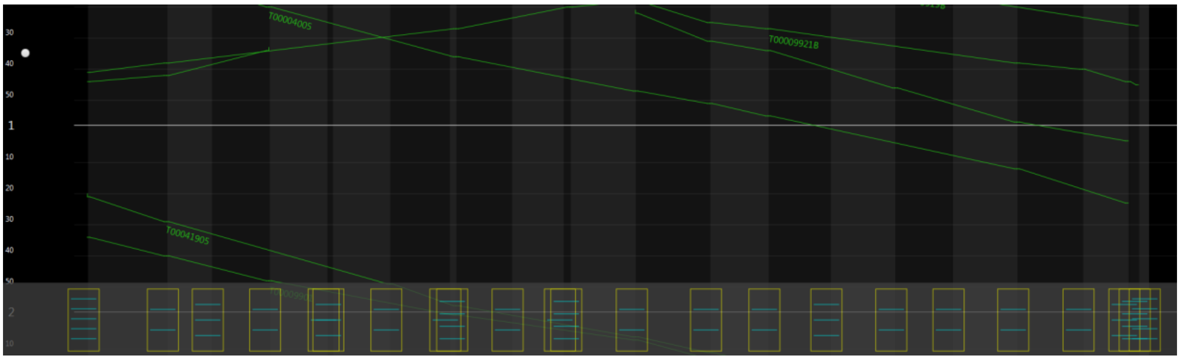

Figure 4-7 - TG + Path Selection

Another important tool to help the Train Controller in making decisions is the Track Allocation interface. Used in the proper way, it allows enhancing the capability of line. An example of the Track Allocation integrated into the TG is shown in the next picture.

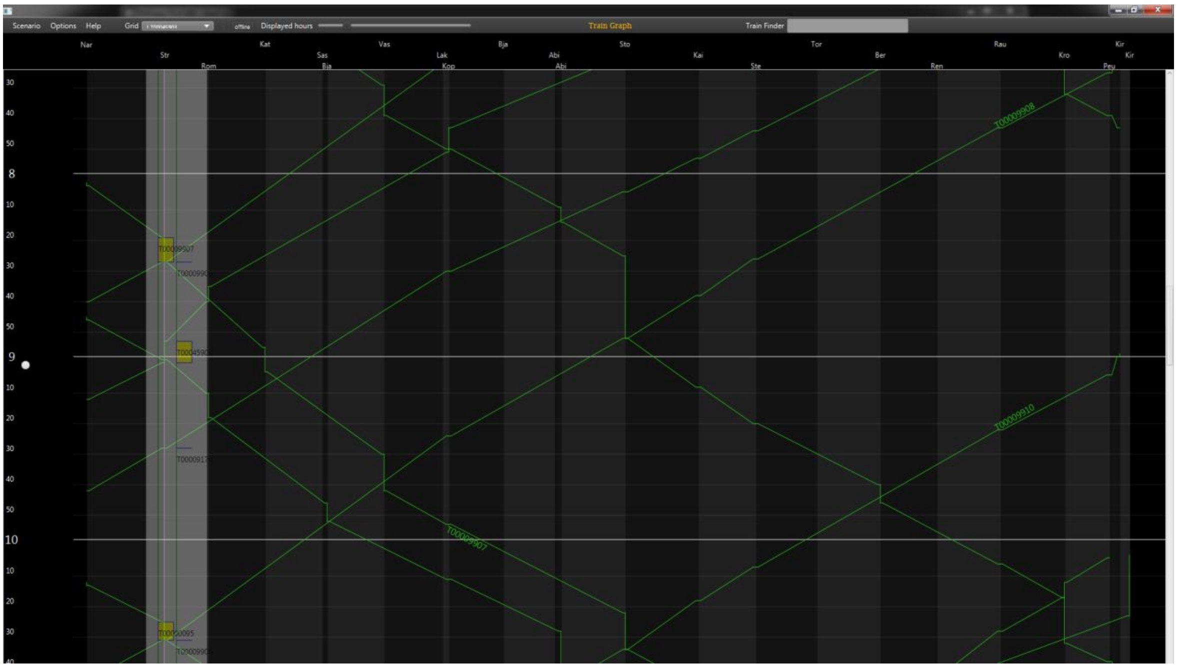

Figure 4-8 - TG + Track Allocation

Ii is possible of course, to have both Path Selection and Track Allocation on the same screen.

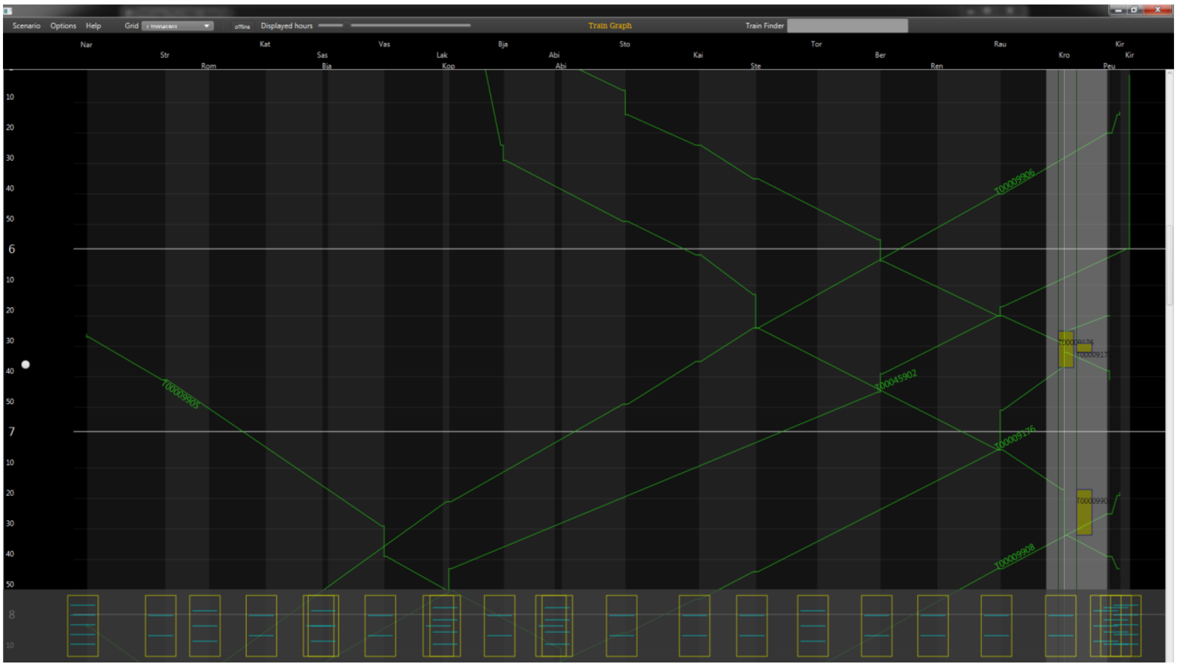

Figure 4-9 - TG + Path Selection + Track Allocation

4.3 Optimization

There aren't good solutions "a priori". A solution can be good in a context but ineffective or even dangerous in another one. This means that an expert system must find several solutions for a single problem and let the user choose between them.

OnTime developed different algorithms that can be used in different scenarios. Every algorithm tries to maximize one or more cost functions. The question now is: which is the best one to apply? The new HMI not only readily allows the user to choose one of them but also to give an immediate feedback about the "goodness" of the choice. This is called "what if analysis".

4.3.1 Demonstration 8: Optimization

Different algorithms, different solutions, different outcomes. Here is how the HMI will ask the user to choose among several solutions.

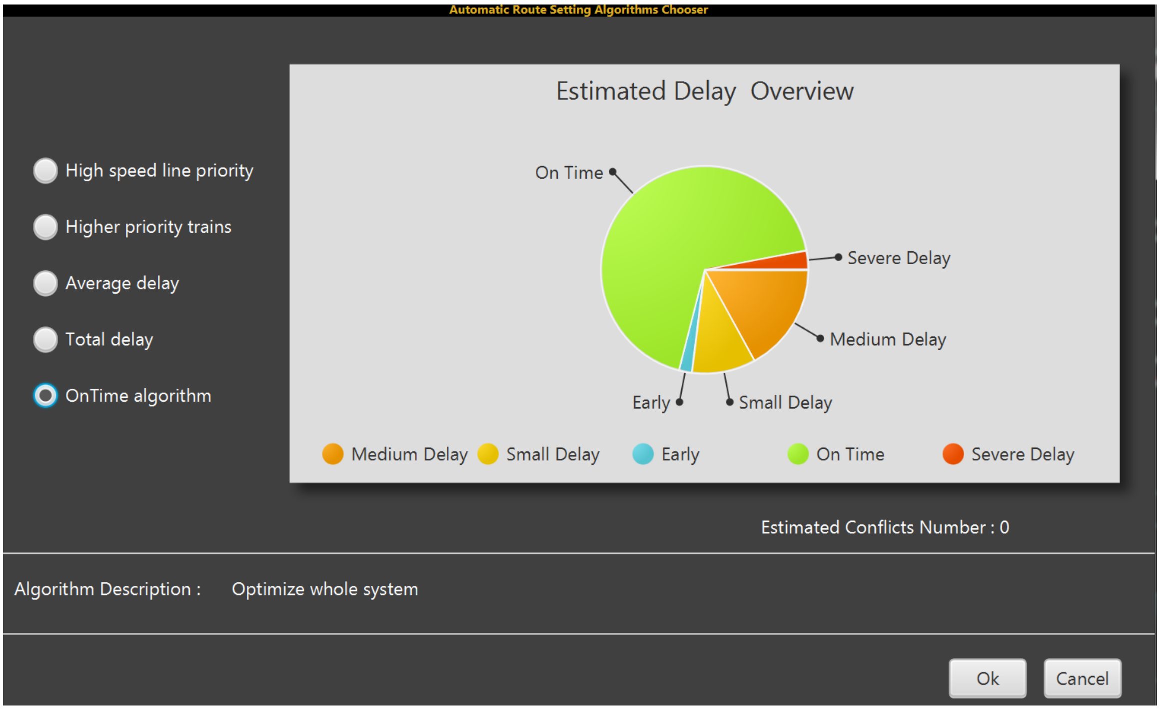

Figure 4-10 - What If Analysis

5 SIMULATOR

Holistic Environment for Rail Modelling and Experimental Simulation (HERMES) is the simulator used by OnTime project. It has a proven GSDK toolkit as its foundation software. HERMES comprises a number of distinct functional components offering rail related services that can be used by algorithmic or presentational components connected into the platform.

The service-based architecture enables future developments and extensions to be made to meet specific client requirements for new concepts, evolutions to existing functionality or to use the facility to generate scenarios or for the development of training systems.

The HERMES platform offers the possibility of designing and validating groundbreaking architectural solutions that exploit new technologies and improved operational processes within the rail industry, coupled with the predictive power of advanced train control algorithms.

The simulator is able to generate a RailMl output describing the railway infrastructure, the events needed by all the modules of OnTime environment (section occupation, train change position and so on) and it is fully integrated in the WP7 communication infrastructure.

5.1 Demonstration 5: Animation with HERMES simulator

In this sub-section there are several screenshots of "animated" TD/TG. Every screenshot represents a particular step in the life cycle of the simulator as stated in its caption.

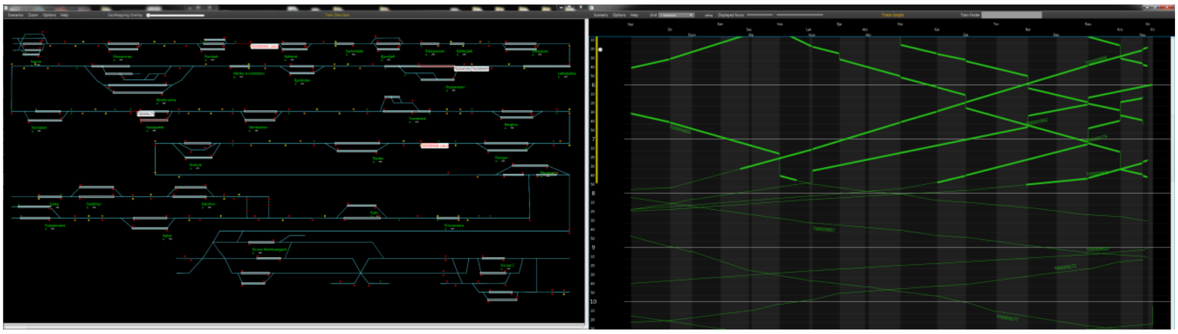

Figure 5-1 - IronOre TD/TG animated by HERMES simulator (Step 1)

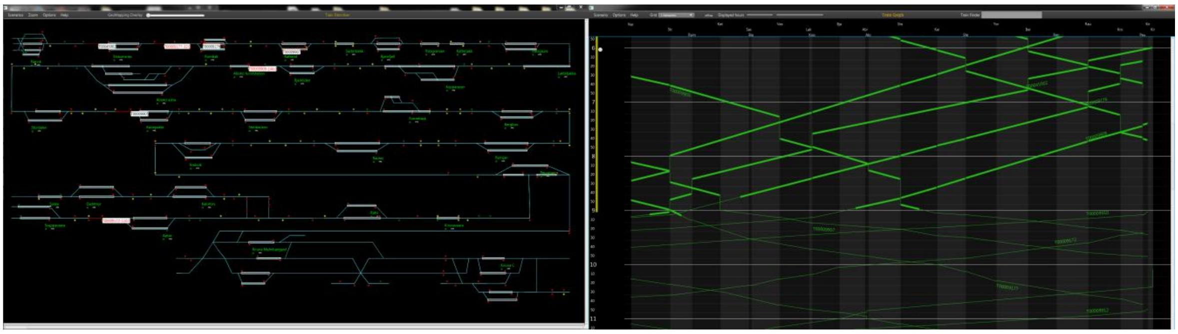

Figure 5-2 - IronOre TD/TG animated by HERMES simulator (Step 2)

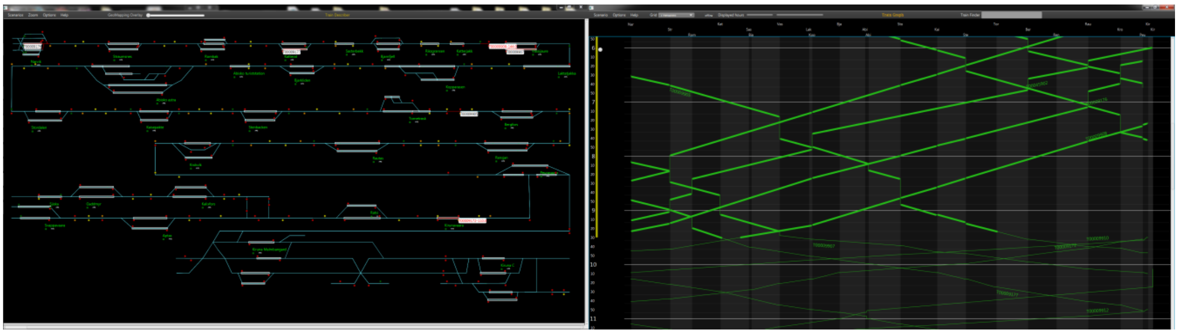

Figure 5-3 - IronOre TD/TG animated by HERMES simulator (Step 3)

In our opinion this other sequence in which a speed restriction (yellow rectangle) generates a conflict between two trains (T9919B and T9926B) is very interesting. The conflict is solved by increasing the dwell time of the first one.

Figure 5-4 - Solving conflict (Step 1)

Figure 5-5 - Solving conflict (Step 2)

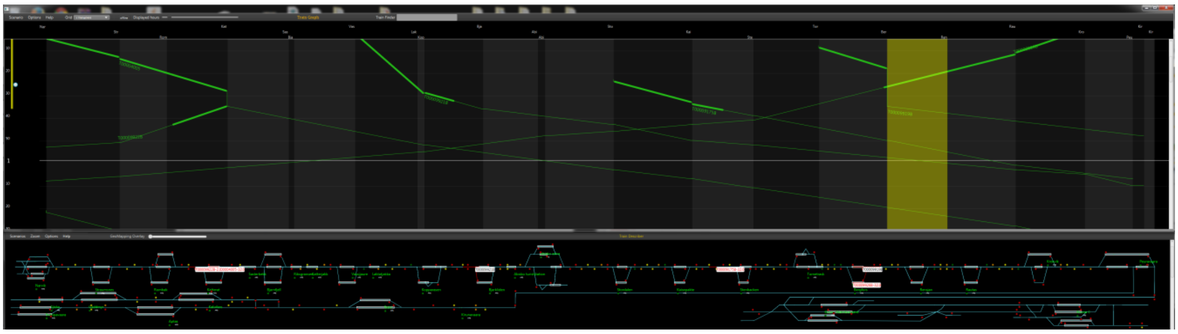

Figure 5-6 - Solving conflict (Step 3)

Figure 5-7 - Solving conflict (Step 4)

6 INTERACTIONS

TG and TD are ready to be used in an interactive environment. In OnTime, unfortunately, the interactions are hidden inside every scenario so it wasn't possible to perform a real test of this functionality. However, the tests we conducted using the Ansaldo TMS simulator gave us the feedback we were looking for.

The full list of interactions (operations) can be found in [1], paragraph 3.3.1.3.

6.1 Demonstration 6: Standard interactions

The next two screenshots have been obtained by interacting with the "object station" and the "object train".

They are just two of the most interesting outcomes.

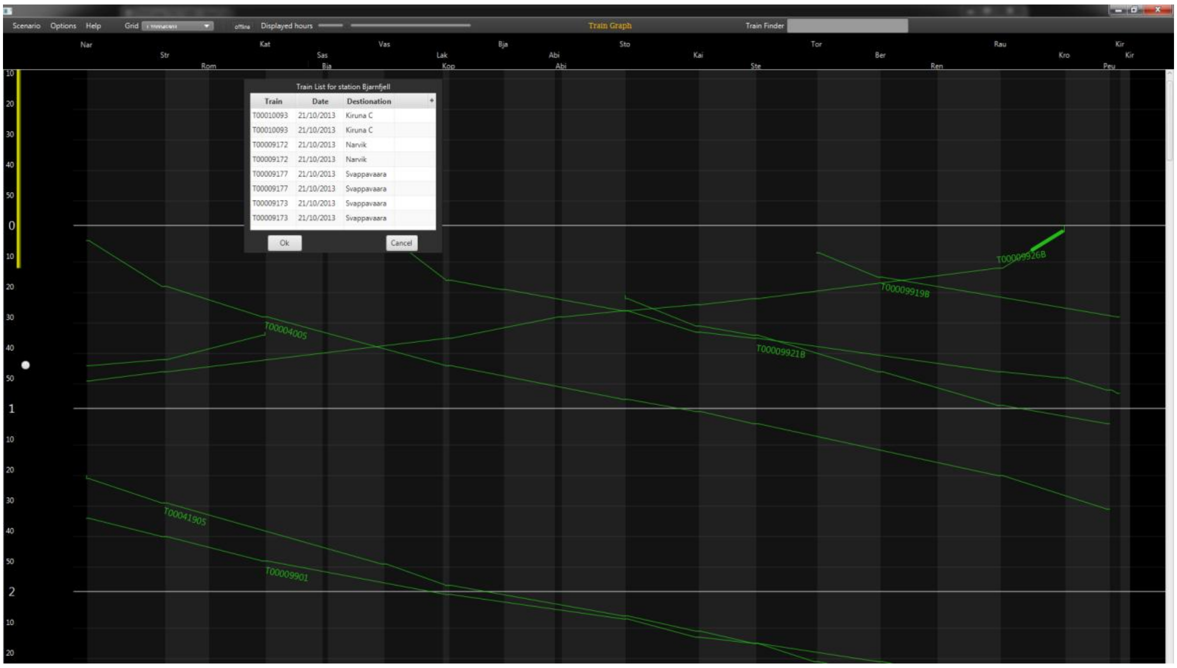

Figure 6-1 - Train list for station

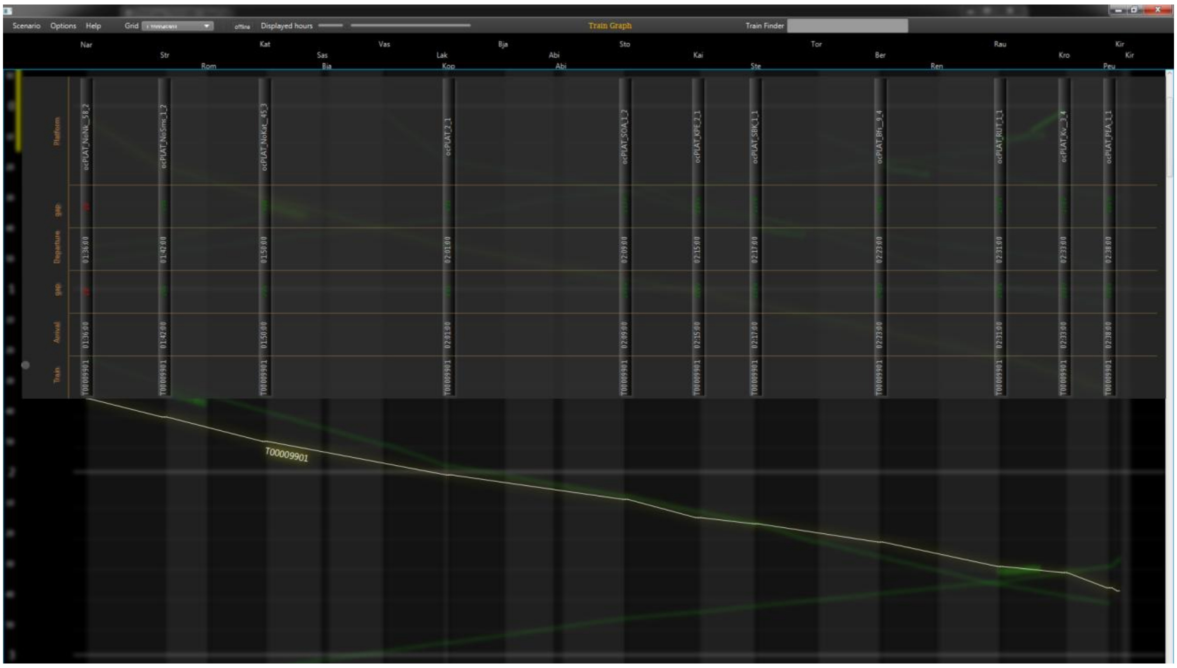

Figure 6-2 - "On screen display" train timetable

REFERENCES

1) Antonio Miano (2013)
D8.1 – WP8 System requirements and specifications