Viriato – Software for railways

Head Office
SMA und Partner AG
Gubelstrasse 28
8050 Zurich, Switzerland

Branch Office
SMA et associés SA
Avenue de la Gare 1
1003 Lausanne, Switzerland

Subsidiaries
SMA Rail Consulting + IT, Corp.
277 Santa Ana Market Place, Suite 825
Santa Ana, CA 92705, USA

SMA (Deutschland) GmbH
Stresemannallee 36
60318 Frankfurt, Germany

SMA (France) SAS
45-47 Rue d’Hauteville
75010 Paris, France

info@sma-partner.com
www.sma-partner.com
Editorial services
SMA und Partner AG, Zurich
Visual concept
Eggmann Design, Tann
Printing
Linkgroup, Zurich

Photo credits
Cover
Large photo: Sebastian Terfloth, Dresden
Small photo: Panthermedia
Page 2
Panthermedia
Page 4
David Kaplan, Schwarzenberg, www.myLandscape.ch
Page 6
Johannes Fer, München, www.feine-fotos.com
Page 8
Panthermedia
Pages 10/11
Panthermedia
Pages 12/13
Ingrid Scherck, Senden
Pages 14/15
Rudolf Ganz, Mainz
Pages 16/17
Panthermedia
Pages 18/19
Ingo Seidritz, Büdelisdorf
Pages 20/21
Sebastian Terfloth, Dresden
Pages 22/23
Panthermedia
Pages 24/25
Panthermedia
Pages 26/27
Panthermedia
Pages 28/29
Klaus Schmückle, Kernen-Stetten
Pages 30/31
Panthermedia
Pages 32/33
Sebastian Terfloth, Dresden
Pages 34/35
Sebastian Terfloth, Dresden

© SMA und Partner AG | September 2016
INTRODUCTION

Viriato is a powerful integrated timetable planning tool that allows users to optimise the railway planning process.

Since Viriato was first introduced in 1996, it has undergone a continuous process of development to add new features and functionality, and has been recently completely rewritten to take advantage of the latest software engineering practices using an up-to-date architecture and technology. Originally focussing on strategic timetabling, its functionality has now expanded to cover the entire planning process, from very long-term timetables through to short-term operational plans.

Viriato integrates into the entire planning process:

- **Strategic**
  Using high-level data to develop initial travel time estimates, rough timetables and rostering plans that optimise vehicle utilisation. Viriato helps planners identify optimal timetables by allowing them to easily compare alternatives.

- **Capacity**
  As timetable implementation approaches, Viriato can work with detailed data to refine and share timetables between stakeholders.

- **Operations**
  Daily timetables can be prepared for use by train operators and infrastructure managers.

Viriato provides timetable planners with a full suite of applications addressing all aspects of the timetable planning process. The Viriato interface consists of the graphic timetable, trip time analysis, conflict detection, netgraphs, platform occupation charts and customer timetables in addition to data manipulation features.

To allow the optimal feature coverage, Viriato has been developed into two parallel products, Viriato Standard for timetables with a high degree of regularity, and Viriato Enterprise allowing greater flexibility as the operational requirements emerge.

Following a major investment project we have recently developed a new integrated vehicle rostering module that allows efficient plans to be made enabling effective use of rolling stock assets while the timetable is being prepared. This module includes automation and optimisation functionality to assist the user in the vehicle planning process.

Viriato allows the user to work with timetable concepts varying in size from aspirational schemes through to fully populated national databases, allowing the creation of efficient plans maximising the usage of network capacity and rolling stock assets, while providing high quality services to customers.

Viriato forms a complete timetable planning suite that helps timetable planners to develop optimal timetables for all levels of operations quickly, accurately and transparently. Thanks to its unique features, it has established itself as an integral part of the timetabling process at more than 90 companies in 15 countries.

Timetable planning at your fingertips
Viriato is the planning tool that covers the full planning process from the strategic through to the operational level. It achieves this through the use of a top-down or “service-centric” methodology where you can freely think about the train services that you want to run before aligning them with the constraints of the network.

Unlike other planning tools, in Viriato you work at a level of data granularity appropriate for the task. A feature-orientated data model lets you concentrate on planning trains and then adding more detailed information later in the process.

Having the appropriate level of infrastructure data for the current task relieves the train planner from maintaining an overly detailed infrastructure model where unnecessary information has to be entered and managed. As the timetable implementation date approaches, Viriato can work with increasingly detailed data to refine the plan. This prevents the planner from being constrained by having to create an unnecessarily detailed model for the task they are working on.

Timetables based on the concept of coordinated interval services provide passengers with excellent services, but are very complicated to develop and evaluate by hand. Viriato was created explicitly for regular-interval timetables and integrates this concept fully throughout the application, in both Viriato Standard and Enterprise versions.

Viriato Enterprise extends this conceptual model further to allow the train planner to make the changes to a highly structured timetable that are needed when faced with commercial variations, engineering works and other sources of modifications to the plan. Trains can have their service patterns altered, be rerouted or use different infrastructure features while retaining their link to the original train.

Viriato consistently follows these principles to enable train planners to develop their plans, and understand any constraints on their services at an earlier stage than would otherwise be possible. This ensures that Viriato provides the maximum possible information to the train planner at each stage in the process, from conceptual design through to production-ready timetables.

Planners can design efficient timetables, allowing them to easily compare alternative scenarios and identify where changes have been made.
Viriato offers the user an efficient working environment in which to create timetables and evaluate their effectiveness.

The program is developed using state-of-the-art software engineering tools and techniques, and using object-oriented architecture offers significantly higher performance and better scalability for very large railway networks. This architecture allows the integration of user-specific functionality and interfaces in a consistent manner.

The latest version of Viriato includes a range of features that significantly increase the efficiency of the timetable planning process:

- The user interface is consistent throughout the entire program, with an efficient workflow with the information required on each screen made directly available to the user.
- The addition or modification of data on any screen is reflected in all views, allowing the planner to visualise and manage at a glance the timetable development process.
- The undo feature allows the planner to experiment with alternatives, and to quickly correct any non-optimal changes made.
- Bulk update functionality speeds up repetitive data entry tasks.
- The train validity model allows each train to have its own individual validity, while user defined templates and update functionality helps to maintain complex patterns easily.
- The graphic timetable can be edited directly, and the user may retime whole or partial trains simply through selecting the relevant portion of a service.
- Conflict detection enables the user to eliminate conflicts dynamically simply by dragging trains on the graphic timetable or platform occupation view.
- User-defined reports allow detailed queries of timetable data to be produced.
- Configurable keyboard shortcuts allowing direct access to features.
- The extensible architecture allows the development of bespoke interface to capacity ordering and management systems.

Viriato can be used with a range of database management systems allowing system installations ranging in size from single users to many hundreds with controlled access rights to data by name and role.
Viriato has been designed with a modular architecture allowing the user to work with the features that are required for the planning task that they are undertaking. As more detail outputs become necessary, additional modules can be seamlessly integrated into the program.

**Viriato Standard features:**
- Netgraph – service-centric view of timetable showing service frequencies and routes
- Graphic timetable – visualisation of train service and interaction between trains
- Customer timetable – tabular presentation of timetable
- Calendar – set running day patterns to allow timetable variants by day
- Running time calculation – calculate precise technical running times
- Platform occupation – view station utilisation and resolve capacity constraints
- Network visualisation map
- railML timetable import/export

**Viriato Enterprise additional features:**
- Infrastructure variants – dated changes to the infrastructure during a timetable period projected onto timetable
- Flexible train model – Different timings, stop patterns and routing possible within trains on specific dates

**Optional features (available in both Viriato Standard and Enterprise):**
- Vehicle rostering – prepare detailed plans of fleet utilisation and maintenance tasks
- Conflict detection – verify compliance of the timetable with infrastructure constraints
- Robustness analysis – verify the resilience of the timetable to perturbations caused by operational incidents
- Trip time analysis – display in tables and graphics the journey time between locations including change times, and export data for deeper analysis
- Passenger assignment – algorithms to assign passengers to specific trains based on journey attractiveness
- Job management – schedule the execution of long running tasks such as railML exports to run automatically in the background

Sometimes the planning workflow process will have specific additional requirements not covered by the standard features of Viriato. For these cases SMA has extensive experience in developing interfaces to other information-management systems. Built-in outputs with standard file formats such as railML offer instant compatibility with a wide range of third-party applications.
The introduction of coordinated ("interval") timetables has led to the ability to represent timetables in a way that shows both the geographical network and the train times in an efficient manner on a “netgraph”. A single line represents each service (and the corresponding journey in the reverse direction) connecting the stations it passes through on its trip, with the arrival and departure times for each station indicated. The cyclical aspect of the timetable means that a whole repeating family of trains can be covered by one line on the diagram. The connection times between services at a station can be seen at a glance, giving a powerful method for improving the coordination between services for the train planner.

Viriato was the first planning tool to successfully combine this netgraph with graphic and tabular timetables to produce a single integrated planning tool.

### Functionality
- Fully integrated with the timetable database – change train data in one location and it changes in all Viriato views
- Step-by-step undo function (to last saved version)
- Powerful snap-to-grid mechanism supports the user in the efficient drawing of netgraphs with a visually compelling layout
- Advanced drawing functions for legends, comments, etc. allow the production of ready-to-print netgraphs
- Grouping of multiple trains on common sections (e.g. to display an hourly service of several different trains with two-hourly frequency over the same route) to maintain visual clarity

### Display/output
- Directly work with trains in the netgraph
- Select trains for analysis and illustration using specific day filters
- Create easy-to-read line maps
- Import and export data between different Viriato databases
- Separation of views for display on screen and for printing
- Copy netgraph to clipboard or export it directly to a graphics file (including *.pdf, *.svg)
Graphic timetables represent train operations and are pivotal when determining the interactions between trains at a network-wide level. Graphic timetables can also be used to identify the likely rolling stock requirements for a line and produce initial utilisation plans.

An experienced timetable planner can assess the feasibility of a timetable and the plan, such as insufficient headways between services or conflicting movements over a single track section simply by looking at a graphic timetable.

Viriato’s highly configurable graphic timetable functionality allows users to represent the trains according to their needs. For example, colours may indicate different train types (e.g. by service group or line), while line thickness may reflect express or regional service. Such customisation gives timetable planners the power to produce highly descriptive graphical timetables, and to identify timetable improvements.
<table>
<thead>
<tr>
<th>S1</th>
<th>IR</th>
<th>IR</th>
<th>S1</th>
<th>S1</th>
<th>S (21)</th>
<th>IR</th>
<th>ICN</th>
<th>S1</th>
<th>S1</th>
<th>S (11)</th>
<th>S1</th>
<th>S1</th>
<th>S (11)</th>
<th>S1</th>
<th>S1</th>
<th>S (11)</th>
<th>S1</th>
<th>S (11)</th>
</tr>
</thead>
<tbody>
<tr>
<td>22123</td>
<td>2305</td>
<td>2257</td>
<td>21119</td>
<td>21123</td>
<td>20115</td>
<td>2307</td>
<td>21125</td>
<td>22129</td>
<td>20119</td>
<td>2309</td>
<td>10013</td>
<td>21125</td>
<td>21129</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>16</td>
<td>16</td>
<td>15</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:14</td>
<td></td>
<td>6:35</td>
<td>6:45</td>
<td></td>
<td>7:05</td>
<td>7:14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:16</td>
<td></td>
<td>6:32</td>
<td>6:48</td>
<td></td>
<td>7:02</td>
<td>7:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:18</td>
<td></td>
<td>6:34</td>
<td>6:50</td>
<td></td>
<td>7:04</td>
<td>7:18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:21</td>
<td></td>
<td>6:37</td>
<td>6:54</td>
<td></td>
<td>7:07</td>
<td>7:21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:25</td>
<td></td>
<td>6:41</td>
<td>6:57</td>
<td></td>
<td>7:11</td>
<td>7:25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:29</td>
<td></td>
<td>6:45</td>
<td>7:01</td>
<td></td>
<td>7:15</td>
<td>7:29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:30</td>
<td></td>
<td>6:47</td>
<td>7:03</td>
<td></td>
<td>7:17</td>
<td>7:31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:34</td>
<td></td>
<td>6:49</td>
<td>7:07</td>
<td>7:09</td>
<td>7:19</td>
<td>7:34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6:49</td>
<td>7:07</td>
<td>7:25</td>
<td>7:38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Locamo
[2] 16

Lugano
[1] 17
The customer timetable is the traditional method for presenting the schedule of trains along a route. Indeed, its structure is reflected in the etymology of the word timetable itself, which refers to the tabular presentation of times. Despite this long history, the customer timetable in Viriato has increased its capabilities. Used in conjunction with the operational modules, such as the graphic timetable, it provides further insights into aspects of the timetable on rail traffic.

Timetables also form a clear legal description of the service in terms of product type, coverage, frequency and/or station calling pattern policy, and are thus indispensable. The functional requirements for a customer timetable vary depending on how it will be used.

Viriato provides flexible filter possibilities and layout settings that permit users to easily display the desired information. Viriato also provides an interface that allows customer timetable data to be analysed and/or displayed using MS Excel.

Facts and figures

<table>
<thead>
<tr>
<th>Functionality</th>
<th>Display/output</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Prepare customer timetables for selected line segments or combinations with branches</td>
<td>■ Display timetables for selected time periods of up to 24 hours</td>
</tr>
<tr>
<td>■ Directly open trains from the customer timetable</td>
<td>■ Select display of arrival, departure and passing times</td>
</tr>
<tr>
<td>■ Step-by-step undo function (to last saved version)</td>
<td>■ Individually show/hide trains in the timetable</td>
</tr>
<tr>
<td>■ Select trains using filters for operating days and validity periods</td>
<td>■ Use colour to distinguish between train type or status</td>
</tr>
<tr>
<td>■ Show the overtaking of trains</td>
<td>■ Freely select display layout, colours, fonts, etc.</td>
</tr>
<tr>
<td>■ Indicate validity changes in a train’s operation</td>
<td>■ High quality PDF printed outputs</td>
</tr>
<tr>
<td>■ Interface with MS Excel</td>
<td></td>
</tr>
</tbody>
</table>
At the conceptual planning level a nominal day can be used to develop and evaluate timetables; however, as the date of timetable introduction draws near, planners must define the timetable in greater detail to include weekend variations, holidays and other service pattern alterations. Actual operations require day-specific timetables to allow responses to short-term changes, such as additional or cancelled trains, to be incorporated in the working plan.

To incorporate this flexibility, Viriato has a train specific validity model allowing the timetable planner to assign any operating pattern to a chosen train, even allowing different validities to trains within a train family. To help manage this process, Viriato allows the definition of template validities that can be modified later on a train specific basis. Viriato generates validity descriptions allowing the user to see at a glance when a train operates.

For the efficient management of train validities within a timetable, the user can modify the running days of multiple selected trains using a bulk update option.

All output views allow the user to filter by days and dates which trains are displayed.

Functionality

- Individual validity information stored at the train level allowing full operational flexibility in the planning process
- Efficient creation and reuse of common operating patterns through the user of saved templates
- Quickly update train validities through batch operations
- Display and analyse operations for any individual calendar date or selection of dates in the netgraph, graphic timetable, customer timetable, station track occupancy plan and arrival/departure tables
- Accurately calculate performance figures based on the calendar-based schedule (e.g. train kilometres, number of departures from a station, etc.)
- Use Viriato’s perpetual calendar to define holidays that change dates every year (e.g. Easter)

Display/output

- Use calendar view to easily define operating/non-operating days by clicking on the required days using the mouse
- From within any train view edit the days that this service operates
- Automatically generate validity descriptions for footnotes in schedules
- Powerful mini-calendar to display effective timings and routing of a train (in Viriato Enterprise version only)
Accurate train planning requires a detailed knowledge of a train's technically feasible running time. The ability to dynamically recalculate the running time during the planning process is essential. This allows changes in rolling stock and infrastructure, such as temporary speed restrictions for engineering works, to be included in the timetable. Viriato makes it possible to calculate precise running times based on a wide range of technical and operational parameters.

The running time calculator in Viriato includes the algorithm used by Swiss Federal Railways (SBB) and by Infrabel to calculate all their running times in Switzerland and Belgium. This calculator allows an unlimited combination of rolling stock types, traction values, dynamic resistances and braking curves to produce highly accurate running times which have been validated by SBB during their daily operations.

The civil engineering characteristics of the infrastructure such as gradients, curvature and speed restrictions are entered to produce a highly detailed model of the railway.

The Viriato running time calculator allows users to easily define their own rolling stock types with associated performance characteristics.

The Viriato running time calculator integrates fully into the train planning process, and the results are instantly reflected in a train's running times.

### Functionality
- Validated industry standard running time calculation engine
- Fully integrated into Viriato
- Gradients, tunnel and curvature information
- Detailed traction and braking system characteristics definition
- Multiple infrastructure speed profiles for different vehicle types
- User-defined station-specific entry and exit speeds
- Train configuration changes during train run (e.g. adding or removing vehicles to the consist)
- User-defined parameters for energy-saving operating techniques
- Significantly increased performance combined with additional calculation parameters

### Display/output
- Diagram illustrating line speed and train speed with optional display of gradients and curve profile
- Train line in diagram showing head, middle or rear of train or both head and rear
- Log file recording all calculated parameters including energy consumption with MS Excel export
- Running time data can be directly transferred to Viriato's train definition
The planning of efficient vehicle rosters is a key activity affecting the economic performance of a railway. The capital cost of rolling stock is high, and the challenges of meeting the service commitment to passengers and customers, while retaining sufficient margin for maintenance activities requires a coordinated planning approach.

The Viriato Vehicle Rostering module is designed to be fully integrated with the timetabling functionality of Viriato, allowing the creation of vehicle rosters during the planning process.

Throughout the life cycle of a timetable there are different requirements for vehicle rostering, and the Viriato Vehicle Rostering module provides the functionality for this entire process.

During the initial long term strategic planning process the user can concentrate on determining the size of the required rolling stock fleet to deliver the service pattern. As more detail becomes available the vehicle rosters can be refined to include detailed rolling stock information, maintenance activities and empty runs.

The Viriato Vehicle Rostering module includes the following features:

- The vehicle rostering process is fully integrated with the timetable. If running times or routes are changed, this is reflected in the vehicle roster too. User definable tolerances allow the planner to decide what level of change is allowed before a revised roster is required.
- The rostering process is undertaken in a graphical environment. The user can drag unplanned trips to a roster. This trip is then automatically linked with the previous trips and suggestions are made for the next trips to link.
- Vehicles to roster can be split by any user defined criterion. This allows plans to be prepared by a range of rules including by rolling stock type, assigned business group, base depot, or other parameters.
- The vehicle rosters can be created and managed in several different types of Gantt-chart type views. The compact view compresses a time period into a single day allowing the planner to quickly see which activities link consistently and which occur less frequently. A rolled-out view shows the roster in a continuous day-by-day format allowing the links to be traced through the plan. Finally, a vehicle-specific view is available allowing the sequence of work for a single unit to be examined.
<table>
<thead>
<tr>
<th>Train Family</th>
<th>Number</th>
<th>From (Name)</th>
<th>To (Name)</th>
<th>Departure</th>
<th>Arrival</th>
<th>Validity</th>
<th>RollStock ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>RVZH S 17</td>
<td>18342</td>
<td>Zürich HB L</td>
<td>Sihlbrugg</td>
<td>15:47.5</td>
<td>16:14.0</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 17</td>
<td>18343</td>
<td>Zürich HB L</td>
<td>Sihlbrugg</td>
<td>16:47.5</td>
<td>17:14.0</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 17</td>
<td>18344</td>
<td>Zürich HB L</td>
<td>Sihlbrugg</td>
<td>17:47.5</td>
<td>18:14.0</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 17</td>
<td>18345</td>
<td>Zürich HB L</td>
<td>Sihlbrugg</td>
<td>18:47.5</td>
<td>19:14.0</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 17</td>
<td>18346</td>
<td>Zürich HB L</td>
<td>Sihlbrugg</td>
<td>19:47.5</td>
<td>20:14.0</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19760</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>05:11.0</td>
<td>05:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19761</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>06:11.0</td>
<td>06:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19762</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>07:11.0</td>
<td>07:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19763</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>08:11.0</td>
<td>08:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19764</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>09:11.0</td>
<td>09:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19765</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>12:11.0</td>
<td>12:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19766</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>13:11.0</td>
<td>13:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19767</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>14:11.0</td>
<td>14:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19768</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>15:11.0</td>
<td>15:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19769</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>16:11.0</td>
<td>16:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19770</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>17:11.0</td>
<td>17:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19771</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>18:11.0</td>
<td>18:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19772</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>22:11.0</td>
<td>22:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19773</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>23:11.0</td>
<td>23:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19774</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>10:11.0</td>
<td>10:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19775</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>10:11.0</td>
<td>10:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19776</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>10:11.0</td>
<td>10:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
<tr>
<td>RVZH S 40.X</td>
<td>19777</td>
<td>Regensdorf-Watt</td>
<td>Uetikon</td>
<td>10:11.0</td>
<td>10:57.3</td>
<td>daily</td>
<td>RE450-1</td>
</tr>
</tbody>
</table>
Unique to Viriato vehicle rostering is the concept for laying out rosters. The user is free to plan the vehicle rosters using the rosters as a form of sketching tool. Once the roster is finalised the program will convert this to a publication ready format based on the vehicle work pattern.

Automation and optimisation to speed up the process of generating and improving rosters. Automation is included to allow the user to quickly find a valid roster plan, or to complete an unfinished one. An interface is available allowing an external optimisation tool to control all aspects of a vehicle roster so that an optimal solution is found respecting vehicle movements, empty runs, maintenance activities and any user specific business rules that must be respected.

The vehicle rosters can be viewed and edited directly on a graphic timetable allowing clear visualisation of the work undertaken by vehicles within the timetable environment.

Rosters can be exported through the railML timetable standard allowing data exchange with other systems.

The sequence of vehicles within a multiple unit is automatically managed in Viriato. Thus the rostering tool determines when a vehicle has changed direction, and alerts the user if a roster plan contains planned activities which are not possible because a vehicle is trapped by another unit.

Built in plausibility checks ensure that the rosters created by the user contain links for all dates within the timetable period ensuring the creation of valid rosters.

Reports showing utilisation of vehicles during each day and vehicles waiting at nodes allowing the planner to manage times of peak demand for rolling stock and storage locations.

Empty runs for stock balancing purposes can be created as either rostering only activities, or trains inside the Viriato database itself which move through the infrastructure as “normal” trains. This allows the planner to see the effects on capacity consumption of the vehicle roster to make the roster more reliable.

Maintenance activities can be planned in the roster, with validation rules based on time interval between activities (e.g. a cleaning activity every four hours) or distance between activity (e.g. refuelling at least every 1000km). Rule sets can be defined specifying which types of activities can occur at each location, allowing flexible planning of vehicle specific tasks.

Functionality
- Create vehicle rosters through drag-and-drop
- Automatically generate rosters efficiently using built-in optimisation engine
- Complete existing partial rosters through automation
- Define empty vehicle movements
- Define maintenance tasks and frequencies
- Integrated within timetabling functionality allowing opening of trains from within roster
- Work with vehicles operating in multiple combinations
- Partition timetable data into efficient and logical sets for rostering
- Manage changes of vehicle direction during journey
- Optimisation interface available for third party systems

Display/output
- Prepare vehicle rosters as Gantt charts
- Display validation rules violations
- Print high quality vehicle rosters in detailed and compact formats
- Output summary statistics and reports
Stations are at the heart of railway operations. Here lines converge, trains arrive and depart, passengers come and go and change trains, and trains are taken out of service, cleaned and maintained. As train operators introduce regular interval timetables and develop major connection points, the operational functions and capacity requirements placed on stations increase significantly.

Given these conditions, every step must be planned in detail. Therefore the analysis of the allocation of platforms, their occupancy and the arrival/departure times are a core part of the timetable production process. The questions that are addressed in this analysis include: Are there sufficient platforms available for the proposed service concept? And what additional tracks or switches are needed to increase the service?

Viriato’s platform occupancy module provides timetable planners with a simple and powerful tool for planning, evaluating and producing platform occupation charts. The planner can drag and drop trains between platforms and review conflicting moves instantly.

Viriato can help the timetable planner to create conflict-free platform workings with the optional conflict detection module displaying planning rule violations and conflicts between services on a given track or between arriving and departing trains from different tracks within the station area.

### Functionality
- Define track-specific occupancy and duration
- Set line-specific standard defaults and individual restriction rules for track allocation
- Set track-specific stopping points and arrival/departure speeds (for use in travel time calculation)
- Interactively allocate track using drag-and-drop within the allocation diagram
- Working area allowing trains without platforms to be kept in a separate pool until graphically assigned a platform
- Step-by-step undo function (to last saved version)
- Use filters for operating day, date range and validity
- Work directly with a train from the allocation diagram

### Display/output
- Display/print station diagrams with tracks, platforms and lines
- Display the operating sequence in the station with slide control
- Adjust display based on operating day, train number, times, directions
- Freely select colours for displaying trains
- Copy platform occupation charts to clipboard or export it to a graphics file format (including *.PDF, *.SVG)
CONFLICT DETECTION

The dark side of capacity

A railway timetable is subject to numerous constraints, including those due to infrastructure (line headways, interlocking systems, etc.), vehicles (performance, train type, etc.) and the service pattern (intermediate stops, connections, etc.). The higher the railway system’s degree of complexity, and the closer to operating at design capacity, the more these constraints must be considered in timetable planning.

In complex cases it can be very difficult to verify that all the constraints on operation have been considered and that the proposed timetable remains conflict-free.

Viriato’s conflict detection module rapidly determines and visualises all conflicts on the graphic timetable. The user can alter a planned train by dragging the path on the graphic timetable, and any remaining conflicts are instantly shown. Viriato produces a tabular summary of conflicts and durations to enable the planner to resolve them quickly.

The Viriato conflict detection module also detects platform allocation issues. Our rigorous mathematical algorithm detects conflicts within a platform occupation plan, and alerts the user if the proposed service pattern is infeasible, providing a valuable early warning to the train planner.

**Functionality**
- Distinguish between single direction and bi-directional service on a line
- Detect crossings on single track lines
- Detect itinerary conflicts in station gridirons
- Detect insufficient headway times based on user-defined headway standards
- Set station-specific separation times for conflicting routes and overlap distances
- Examine simultaneous track allocations, platform lengths and prohibited routes
- Dynamic drag-and-drop conflict resolution

**Display/output**
- Graphic display of conflicts on the track occupancy diagram and graphic timetable
- Tabular conflicts listing including conflict duration and affected trains
Timetable production is an interactive process involving many constraints and conflicting goals. Operational constraints often have impacts that are not initially obvious. Furthermore, the system-wide impacts of local timetable changes are not directly evident in large railway networks.

Viriato’s trip time analysis module enables timetable planners to compare alternative timetables and analyse timetable and service quality. It starts by identifying all meaningful connections based on the timetable data, trip times and user-defined transfer times. Once the connections have been identified, Viriato calculates the alternative’s most important customer and operational qualities.

The trip time analysis module provides many different analysis tools enabling timetable planners to thoroughly evaluate alternative timetables, and permitting the attractive presentation of this data in a variety of graphical formats. This allows the travel opportunities offered by the new timetable to the railway and its customers to be communicated effectively.

### Functionality

- Determine all possible connections with: shortest trip time, fewest transfers and shortest weighted trip time, for any selection of origin/destination pairs
- Compare summary and specific origin/destination pair journey time data between timetables
- Set user-defined transfer weighting values

### Display/output

- Display connection data in a tabular format for a large set of criteria (including trip time, waiting time, total distance travelled, speed, etc.)
- Use filters to display selected data
- Aggregate data based on various criteria
- Graphically view results in the SMA geo-visualisation tool
Understanding passenger flows

The creation of a new timetable and service often influences the choice of route taken by passengers. The Viriato add-on module trip time analysis calculates the routes that can be taken by passengers, including changing trains and indicates any change of service quality between timetables. The Demand Assignment module goes one step further, as it incorporates the passenger demand into trip time analysis and thus allows the determination of the number of passengers travelling on each train in the timetable.

An iterative process is carried out within Viriato which makes transparent to the user the interaction between new timetable service concepts and the associated changes in passenger routing and flows. Within this process occurs the timetable concept creation itself, calculation of the possible connections available, apportionment of the passenger demand on the new routes and thus the loading of trains. As the user modifies the timetable concept this process repeats. This analysis is all undertaken within Viriato using the Trip Time Analysis and Demand Assignment add-on modules.

The algorithm for passenger assignment uses an analogy with the flow of electrical current based on the principle of simultaneous apportionment to calculate routing and loads. The demand from every origin-destination pair is simultaneously distributed in proportion to the resistances for each possible route using all viable journey connections.

### Functionality
- Import demand matrix
- Import daily utilisation curve
- Import connection data (results from Viriato Trip Time Analysis)
- Adjustable resistance exponent in algorithm
- Calculation of part trips
- Calculation of trips parts

### Display/output
- Additional tab showing passenger loading per train journey segment
- Demand per train
In the iterative process of timetable production, evaluation of plans plays an important role. One aspect which is usually especially hard to quantify and assess is the operational robustness of a timetable variant. The determination of the impact of local infrastructure or timetable changes on the robustness of the whole system is a difficult task even for experienced planners when working with large networks.

The robustness analysis module allows the validation of the stability of a timetable and the comparison of the performance of alternative timetables. The user develops delay scenarios containing a set of pre-defined incidents that they wish to test a timetable against. This is then processed by Viriato using the infrastructure data, with the original delays propagated through the timetable until the service pattern returns to normal. Once the calculation is done Viriato produces statistics and the perturbed timetable can be saved as a regular Viriato scenario.

The result of the delay propagation may be reviewed using Viriato, and includes statistics such as the duration for recovery and the total delay minutes due to the original perturbation. These statistics can be exported and used for further analysis and for the presentation of results. Results saved in a new timetable scenario can be displayed and analysed using all the standard Viriato functions. This allows the robustness of any new timetable to be effectively analysed and easily communicated.

**Functionality**
- Define delay scenarios by injecting delays to trains into a timetable
- Determine the behaviour of trains by defining which kind of planned time reserves can be used to catch up delays
- Uses the Viriato conflict detection to model infrastructure capability
- Simulate the propagation of delays in the timetable through a railway network
- Analysis of results and production of aggregated statistics

**Display/output**
- Export the statistics in .csv files for further analysis
- Save the perturbed timetable as a new Viriato scenario
- Display the results using the graphical timetable comparison and all other standard Viriato views
A train timetable goes through many planning stages over its long period of development. The process starts with the preparation of a draft service plan. This is constantly refined until operations of the train commence. Finally, after the introduction of the timetable system performance is analysed. Finding the timetable that best balances the railway’s operational and economic needs requires much iteration.

Viriato’s versatility enables it to support timetable planners at all stages of the development process. Nevertheless data will almost certainly need to be exchanged with other applications at some point in the process, and used either in parallel or sequentially. A central element in the data exchange process is insuring efficient data flow without wasteful and error-prone re-collection of data. The challenge consists of linking different applications over intelligent and standardised interfaces.

The railML initiative was created to improve the data exchange process between railway information technology applications (www.railML.org) through the development of agreed standards. SMA and Partner AG was a founding member of the railML initiative and continues to actively participate in railML development. The goal of railML is to link different applications through the creation of defined interfaces between diverse rail service planning and operations IT applications, and to simplify information exchange with the definition of standardised XML-based schemas.

Open interfaces
- Direct exchange of data between Viriato databases
- Support of the railML standards for timetable import and export
- Direct export to MS Excel

Proprietary interfaces
- PPSFR (DB vehicle rostering planning for passenger service)
- TPN (DB Netz path portal for ordering train paths)
- THOR (timetable system of SNCF-Réseau)
- BookIN (Infrabel slot portal for SNCF to order train paths)
- LIKE (slot portal of Finnish Transport Agency FTA)
- PCS (path coordination system of RailNetEurope)
- NetS (timetable construction system of SBB)
- KIS (customer information system of RhB)
- KSS (DB Netz timetable format for various systems)

Application Server
- Citrix-compatible for operation on an application server
- Windows Terminal Server compatible

Data base system recommendations
- MS Access
  - Recommended for stand-alone installations and small work groups
  - Provides flexibility and simple administration
- Oracle/MS SQL
  - Recommended for multi-user work groups
  - Provides high level of data security and performance
  - WAN-compatible
  - Flexibility for local work on mobile clients under MS Access using data extract and synchronisation functions
- User authorisation concept: Role based multi-level authorisation for Viriato functions (administrator, super-user, user, guest...), user-group based authorisation for timetable data access (write, read, no access)
AMMT Agenzia Mobilità Metropolitana, Turin (IT)
Amt für öffentlichen Verkehr Kanton Zug, Zug (CH)
AB Appenzeller Bahnen AG, Herisau (CH)
assa AG, Rapperswil-Jona (CH)
BART Bay Area Rapid Transit, Oakland (US)
BLS AG, Bern (CH)
Citec Ingénieurs Conseils SA, Geneva (CH)
CNAM Conservatoire national des Arts et Métiers, Paris (FR)
DB Fernverkehr AG, Frankfurt a. M. (DE)
DB Netz AG, Frankfurt a. M. (DE)
DB Regio AG, Frankfurt a. M. (DE)
Egis Rail, Guyancourt (FR)
Ente Autonomo Volturino, Napoli (IT)
Swiss Federal Institute of Technology, IVT, Zurich (CH)
Swiss Federal Institute of Technology, LITEP, Lausanne (CH)
University of Gelsenkirchen, Gelsenkirchen (DE)
FART Ferrovie Autolinee Regionali Ticinesi, Locarno (CH)
FTA Finnish Transport Agency, Helsinki (FI)
University of Nuremberg, Nuremberg (DE)
Herzog Transit Services Inc., Irving (US)
HSR – HRT, Helsinki Regional Transport Authority, Helsinki (FI)
Ifsttar, Villeneuve d’Ascq (FR)
Ingerop Conseil et ingénierie, Courbevoie (FR)
IP Engenharia, Lissabon (PT)
IP Infraestruturas de Portugal, Lisboa (PT)
Jernbaneverket JBV, Oslo (NO)
Jungfrau Railways, Interlaken (CH)
Keolis, Paris (FR)
Kompetenzzentrum ITF NRW, Bielefeld (DE)
LNVG Landesnahverkehrsgesellschaft Niedersachsen mbH, Hanover (DE)
Lucchini – Mariotta e Associati SA, Ponte Capriasca (CH)
NAH SH GmbH, Kiel (DE)
Metron Verkehrsplanung AG, Brugg (CH)
Ministerium für Wirtschaft, Verkehr, Landwirtschaft und Weinbau Rheinland-Pfalz, Mainz (DE)
MOB Golden Pass Services, Montreux (CH)
Nahverkehr Rheinland GmbH, Cologne (DE)
Net Engineering, Monselice (IT)
NSB AS, Oslo (NO)
NVBW Nahverkehrsgesellschaft BadenWürttemberg mbH, Stuttgart (DE)
NVV Nordhessischer Verkehrsverbund, Kassel (DE)
Plateway Pty Ltd, Clyde (AU)
ProRail, Utrecht (NL)
PTN Passenger Transport Networks, York (UK)
Région Rhône-Alpes, Lyon (FR)
RegionAlps SA, Martigny (CH)
RHB Rhaetian Railway, Chur (CH)
RMV Rhein-Main-Verkehrsverbund GmbH, Hofheim a. Ts. (DE)
Rigi Railways, Vitznau (CH)
Roma Servizi per la Mobilità, Rome (IT)
RWTH Aachen, Aachen (DE)
SBB Passenger Traffic, Bern (CH)
SBB Infrastructure, Bern (CH)
Setec Ferroviaire, Paris (FR)
SNCF/NMBS Société Nationale des Chemins de Fer Belges, Brussels (BE)
SNCF Réseau, Paris (FR)
SNCF Mobilités, Paris (FR)
Systra, Paris (FR)
University of Berlin, Berlin (DE)
University of Science and Technology Trondheim, Trondheim (NO)
TMR Transports de Martigny et Régions SA, Martigny (CH)
TPF Transports publics fribourgeois, Fribourg (CH)
TransN Transports publics neuchâtelois, La Chaux-de-Fonds (CH)
TRAVYS SA, Yverdon-les-Bains (CH)
Trenord S.r.l., Milan (IT)
Regionalbahn Thüringen AG, Kreuzlingen (CH)
University of Birmingham, Birmingham (UK)
University of Pardubice, Pardubice (CZ)
University of Stuttgart, Stuttgart (DE)
University of Trondheim, Trondheim (NO)
University of Zilina, Zilina (SK)
VMV Verkehrsgesellschaft Mecklenburg-Vorpommern mbH, Schwerin (DE)
VNR Vietnam Railways, Hanoi (VN)
VR Group Ltd, Helsinki (FI)
WSP Parsons Brinckerhoff, Baltimore (US)
ZHAW University of applied sciences zurich, Winterthur (CH)
Zweckverband SPNV Münsterland, Münster (DE)