|  |
| --- |
|  |
| NEXTA Data Structure for Rail Scheduling, Version 1.0 |
| Prepared for INFORMS 2012 RAS Problem Solving Competition |
| **Movement Planner Algorithm Design for Dispatching on Multi-Track Territories**  http://www.informs.org/Community/RAS/Problem-Solving-Competition/2012-RAS-Problem-Solving-Competition |
| **RAS Toy Network Data Set is prepared by 2012 RAS Problem Solving Competition Organizing Committee**  **NEXTA Document is prepared by Jeffrey Taylor (**[**jeffrey.taylor.d@gmail.com**](mailto:jeffrey.taylor.d@gmail.com)**) and Xuesong Zhou (**[**zhou@eng.utah.edu**](mailto:zhou@eng.utah.edu)**)** |
| **Last Revised: 7/26/2012** |

|  |
| --- |
| Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in [www.gnu.org/licenses/fdl.html](http://www.gnu.org/licenses/fdl.html). |

**If you have any questions about the competition problem, submit your question to RASProblemSolvingCompetition@gmail.com.**

**Please feel free to send any questions, feedback, and corrections to Jeffrey Taylor (**[**jeffrey.taylor.d@gmail.com**](mailto:jeffrey.taylor.d@gmail.com)**) or Dr. Xuesong Zhou (**[**zhou@eng.utah.edu**](mailto:zhou@eng.utah.edu)**) by adding comments in this document and including the file as an attachment.**

Table of Contents

[NEXTA Data Structure 3](#_Toc331998302)

[Simple Step-by-Step User Guide 3](#_Toc331998303)

[Input Files 4](#_Toc331998304)

[1. Network Files 4](#_Toc331998305)

[input\_rail\_node.csv [Essential input data] 4](#_Toc331998306)

[input\_track\_type.csv [Essential input data] 6](#_Toc331998307)

[input\_rail\_arc.csv [Essential input data] 6](#_Toc331998308)

[input\_train\_info.csv [Essential input data] 7](#_Toc331998309)

[input\_MOW.csv [Essential input data] 8](#_Toc331998310)

[2. Output Files 9](#_Toc331998311)

[output\_schedule.xml [Essential output data] 9](#_Toc331998312)

# NEXTA Data Structure

This document describes all input files associated with NEXTA for visualizing rail scheduling output. Each input/output file includes descriptions for all variable names, followed by a short description of their type, purpose, function, interaction with other variables, and the use cases in which the variable is required/not required.

# Simple Step-by-Step User Guide

)      Download the zipped file GUI\_release\_For\_RAS.zip from the Google code site: <http://code.google.com/p/nexta/downloads/list>

2)      Unzip the file to a folder on a Windows machine.

3)      Go to subfolder “RAS\_Toy\_problem”, which has a reformatted input data set. The file output\_schedule.xml follows the exactly same format as specified in the sample data set.

4)      Go back to the installation folder, click NEXTA.exe

5)      File->Open Rail Network Project, Open a train schedule \*.xml in the subfolder RAS\_Toy\_problem.

6)      Use mouse wheeler to zoom in and zoom out, and move network. If the network does not appear initially, click on button  in the tool bar to display the network and train/string diagram.

7)      Click on tool bar  to show timestamps of train entries, by min and by second.

8)      Train trajectories are shown in solid lines when they are running on main tracks, otherwise as dotted lines on switches and sidings.

9) Go to menu tools->Train List, select a train to highlight its corresponding path on the network and schedule on train/string diagram.

10) Go to menu tools-> Check Schedule Feasibility to check the feasibility of train schedule. Currently, only headway, nonconcurrency and MOW constraints are checked.

11)  Similar to using a GIS package, you can select link layer, and click on “” in the tool bar and use mouse to select a link in order to show the corresponding attributes.

# Input Files

The following tables describe the input files used in NEXTA for rail scheduling. Most tables can be defined as either essential input data (indicated by **Essential input data** label) or nonessential input data, while individual variables (columns) in each table may also be considered as optional variables.

## 1. Network Files

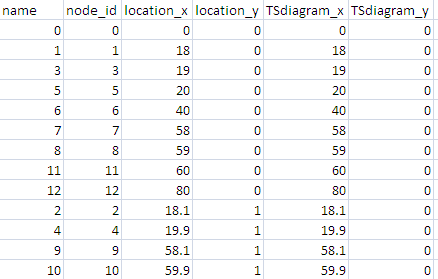
Network input files define the basic node-link structure used in DTALite and NEXTA, along with attributes for each link and node. Additionally, nodes are related to zones and activity locations, which can be used to disaggregate trips from zones to nodes and activity locations.

### input\_rail\_node.csv [Essential input data]

The input\_rail\_node table defines the nodes in the network in terms of names, ID numbers, location/position, and characteristics.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values/ Example Usage** | **Description** |
| Name | String | X |  | Optional: Name label given to node for KML visualization, not currently used in NEXTA |
| node\_id | Integer |  | Value >= 0 | Node identification number |
| location\_x | Double |  |  | describe horizontal coordinate of a node for network visualization |
| location\_y | Double |  |  | describe vertical coordinate of a node for network visualization |
| TSdiagram\_x | Double |  |  | describe horizontal coordinate of a node for space-time diagram visualization, this coordinate can be different from |
| TSdiagram\_y | Double |  |  | describe vertical coordinate of a node for space-time diagram visualization |

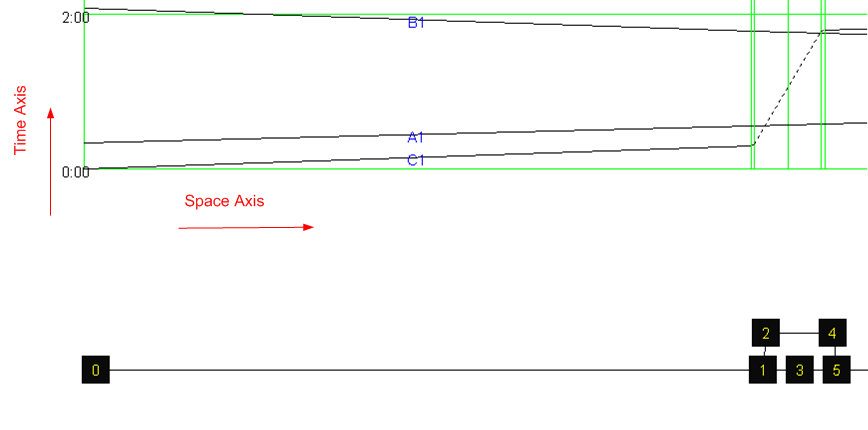
Example from RAS Toy Network:



Why do we use TSdiagram\_x, TS\_diagram\_y, which are different from location coordinates in some cases?

Answer: NEXTA computes the time axis of a time-space diagram by offsetting the TSdiagram x/y coordinates of each node. Thus, a user needs to shift the physical location coordinates of those nodes on switches (e.g., node 3 and 4 in the figure below) so that the time axis of a space time diagram is aligned horizontally or vertically.

Train trajectories are shown in solid lines when they are running on main tracks, otherwise as dotted lines on switches and sidings.



### input\_track\_type.csv [Essential input data]

The input\_track\_type table allows users to define their own specific track types. Link types can also be used to determine how links are visualized in NEXTA.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values** | **Description** |
| track\_type\_code | String |  |  | Optional: Name label assigned to link type in the same row, used for visualization purposes in NEXTA |
| name | String |  | 0 or 1 | Identifies link type as belonging to a freeway class. Only one flag may be used in each row. |
| max\_speed | float |  | >o | Maximum speed for trains running this type of tracks |

Example from RAS Toy Network:

|  |  |  |
| --- | --- | --- |
| track\_type\_code | Name | max\_speed |
| 0 | First Main Track | 80 |
| 1 | Second Main Track | 80 |
| 2 | Third Main Track | 80 |
| SW | Switch | 15 |
| S | Siding | 20 |
| C | Crossover | 15 |

### input\_rail\_arc.csv [Essential input data]

The input\_rail\_arc table defines all links in the network, along with their corresponding characteristics and traffic flow model input data. Several optional fields are included for generating/converting networks for use with microscopic simulation (e.g., VISSIM).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values** | **Description** | **Defined in Table** |
| Name | String | X |  | Optional: Name label assigned to link in current row, used for visualization purposes in NEXTA and KML export |  |
| arc\_id | Integer |  | Value > 0 | Arc identification number |  |
| A\_node\_id | Integer |  | Value > 0 | Identification number corresponding to the node located at the beginning of the link | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| B\_node\_id | Integer |  | Value > 0 | Identification number corresponding to the node located at the end of the link | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| bidirectional\_flag | Integer |  | 1 = single-track  0 or 2= double track | Identifies the direction of travel on the link. When 1, we allow train traverse from A\_node\_id to B\_node\_id , and from B\_node to A\_node |  |
| Length | Double |  | Value ≥ 0.00001 | The length of the link (between end nodes), measured in units of miles or KM. |  |
| track\_type | String |  | 0, 1, 2, …, S, SW, C… | Track type identification code, corresponding to track type (main track, switch, etc.) | [(input\_track\_type.csv)](#_input_link_type.csv_[Essential_inpu) |
| default\_AB\_speed\_per\_hour | Integer |  | Value > 0 mph, kmph | Speed limit on the A-> B direction defined link in units of miles or KM per hour, used to define the free-flow speed. |  |
| default\_BA\_speed\_per\_hour | Integer |  | Value > 0 mph, kmph | Speed limit on the B-> A direction defined link in units of miles or KM per hour, used to define the free-flow speed. |  |

Example from RAS Toy Network:

### 

### input\_train\_info.csv [Essential input data]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values** | **Description** | **Defined in Table** |
| train\_header | string |  |  | Train identification number |  |
| entry\_time | Integer |  | Value ≥ 0 | Time in the schedule at which the train trip begins |  |
| origin\_node\_id | Integer |  | Value > 0 | Departure/origin node identification number | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| destination\_node\_id | Integer |  | Value > 0 | Arrival/destination node identification number | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| direction | string | x |  | Direction which the train trip takes |  |
| speed\_multiplier | double |  | Value > 0 | The train speed on each main track link = speed\_multiplier\* default\_BA\_speed or default\_AB\_speed  e.g. default\_BA\_speed = 80 mph, a train travels through link B to A with a speed multiplier of 0.8, then the actual speed is 80\*0.8 = 0.64.  For non-main tracks, such as switches, sidings, and cross-overs, the speed\_muliplier. E.g. Switch’s default speed 15 mph, the actual speed is also 15 mph for all trains. | This variable is used together with speed value in input\_rail\_arc.csv |
| train\_length | Double |  | Value >= 0; Default: 0 | In output\_schedule.xml, exit time is the exit time of a train’s tail = exit time of the head of a train + train\_length/actual speed on this link. If train\_length is set to 0, then exit time refers to the exit time of a train’s head directly. | This variable is used in output\_schedule.xml |
| tob | Integer |  |  |  | Not used in visualization |
| hazmat | string |  |  |  | Not used in visualization |
| sa\_status\_at\_origin | Integer |  |  |  | Not used in visualization |
| terminal\_want\_time | Integer |  | Value > 0 |  | Not used in visualization |

Example from RAS Toy Network:

### 

### input\_MOW.csv [Essential input data]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values** | **Description** | **Defined in Table** |
| A\_node\_id | Integer |  | Value >= 0 | Identification number corresponding to the node located at the beginning of the link with MOW | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| B\_node\_id | Integer |  | Value >= 0 | Identification number corresponding to the node located at the end of the link with MOW | [(input\_rail\_node.csv)](#_input_node.csv_[Essential_input) |
| start\_time\_in\_min | Integer |  | Value >= 0 | Starting time of MOW in min |  |
| end\_time\_in\_min | Integer |  | Value > 0 | Ending time of MOW in min |  |

Example from RAS Toy Network:

|  |  |  |  |
| --- | --- | --- | --- |
| A\_node\_id | B\_node\_id |  |  |
| 11 | 12 | 0 | 20 |

## 2. Output Files

### output\_schedule.xml [Essential output data]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Variable Name** | **Type** | **Optional** | **Acceptable Values** | **Description** | **Defined in Table** |
| train id | string |  |  | train\_header | input\_train\_info.csv |
| movement arc | string |  | '(%d,%d)' | A node and B node of a link used along the train path | AB or BA direction should be defined in input\_rail\_arac.csv |
| entry | int |  |  | Entry time of a train’s head in seconds |  |
| exit | int |  |  | exit time of a train’s tail in seconds  = exit time of a train’s head + train\_length/actual speed\*3660 seconds/hour |  |

Example from RAS Toy Network:

|  |
| --- |
| <solution territory='RAS DATA SET TOY'>  <trains>  <train id='A1'>  <movements>  <movement arc='(0,1)' entry='1200' exit='2104.500'/>  <movement arc='(1,3)' entry='2010' exit='2149.500'/>  <movement arc='(3,5)' entry='2055' exit='2194.500'/>  <movement arc='(5,6)' entry='2100' exit='3094.500'/>  <movement arc='(6,7)' entry='3000' exit='3904.500'/>  <movement arc='(7,8)' entry='3810' exit='3949.500'/>  <movement arc='(8,11)' entry='3855' exit='3994.500'/>  <movement arc='(11,12)' entry='3900' exit='4894.500'/>  <destination entry='4800'/>  </movements>  </train>  <train id='B1'>  <movements>  <movement arc='(12,11)' entry='1200' exit='2470.588'/>  <movement arc='(11,10)' entry='2410.084' exit='2722.084'/>  <movement arc='(10,9)' entry='2482.084' exit='4129.500'/>  <movement arc='(9,7)' entry='3949.500' exit='4261.500'/>  <movement arc='(7,6)' entry='4021.500' exit='5171.079'/>  <movement arc='(6,5)' entry='5110.575' exit='6381.163'/>  <movement arc='(5,3)' entry='6320.659' exit='6441.668'/>  <movement arc='(3,1)' entry='6381.163' exit='6502.172'/>  <movement arc='(1,0)' entry='6441.668' exit='7591.247'/>  <destination entry='7530.743'/>  </movements>  </train>  <train id='C1'>  <movements>  <movement arc='(0,1)' entry='0' exit='1140'/>  <movement arc='(1,2)' entry='1080' exit='1392'/>  <movement arc='(2,4)' entry='1152' exit='6621.668'/>  <movement arc='(4,5)' entry='6441.668' exit='6753.668'/>  <movement arc='(5,6)' entry='6513.668' exit='7773.668'/>  <movement arc='(6,7)' entry='7713.668' exit='8853.668'/>  <movement arc='(7,8)' entry='8793.668' exit='8913.668'/>  <movement arc='(8,11)' entry='8853.668' exit='8973.668'/>  <movement arc='(11,12)' entry='8913.668'/>  </movements>  </train>  </trains>  </solution> |