
Author: Apane, A. Thabiso (427579)

Due Date: 26 March 2015
DECLARATION

UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

SCHOOL OF MECHANICAL, INDUSTRIAL AND
AERONAUTICAL ENGINEERING

INDIVIDUAL DECLARATION WITH TASK SUBMITTED FOR 
ASSESSMENT

I, Aphane, A. Thabiso (427579), am registered for Course No: BUSE 7005 in the year 2014. I herewith submit the following task, “Development of a Simulation Model of the Town Y Company X Shunting yard.” in partial fulfillment of the requirements of the above course.

I hereby declare the following:

- I am aware that plagiarism (the use of someone else’s work without their permission and/or without acknowledging the original source) is wrong;

- I confirm that the work submitted herewith for assessment in the above course is my own unaided work except where I have explicitly indicated otherwise;

- This task has not been submitted before, either individually or jointly, for any course requirement, examination or degree at this or any other tertiary educational institution;

- I have followed the required conventions in referencing the thoughts and ideas of others;

- I understand that the University of the Witwatersrand may take disciplinary action against me if it can be shown that this task is not my own unaided work or that I have failed to acknowledge the source of the ideas or words in my writing in this task.

Signature: ____________________________ Date: ______________ 20____
Abstract

It was realised that there are inefficiencies at Company X’s plant K shunting yard; service time was long and the idling time of the locomotives was long. Locomotives can be utilised for other purposes in the plant. This has implication in resource planning and productivity in the company.

The study deals with the simulation of the Company X rail network in plant K. The focus is on how shunting and product transportation takes place. A background on the study is given, taking into consideration elements which have been included in the study. These include the locomotives and the Block Train Rail Tanker Cars (RTCs). These containers transport different products from Town L to Town M. The study focuses on the transportation of five products. The study also includes the domestic and international Product E trains arriving at the Product E loading and offloading zone. Simulation model which represents the current-state situation was developed, using SIMIO software package. The study examined how service speed during the process of loading and offloading of products in the plant can be improved. The study also focused on locomotives travelling speed and idling time. Conclusions and recommendations have been made on the model developed. The results obtained were also discussed and analysed.
TABLE OF CONTENTS

Declaration ........................................................................................................................................... i
Abstract ............................................................................................................................................... ii
Definitions of terms .......................................................................................................................... ix
1. INTRODUCTION .......................................................................................................................... 1
   1.1 Background Information ........................................................................................................... 1
   1.2. Problem statement .................................................................................................................. 8
2. OBJECTIVE ..................................................................................................................................... 9
3. LITERATURE REVIEW .................................................................................................................. 10
   3.1. Simulation modelling .............................................................................................................. 10
   3.2. SIMIO .................................................................................................................................... 10
      3.2.1. What is SIMIO? ................................................................................................................. 10
      3.2.2. Benefits of simulation ...................................................................................................... 11
      3.2.3. Definition of objects in SIMIO ......................................................................................... 11
   3.3. Railway Shunting yards ........................................................................................................... 14
      3.3.1 Definitions and descriptions of railway shunting yards.................................................... 14
      3.3.2 Flat yards ............................................................................................................................ 15
      3.3.3 Gravity yards ....................................................................................................................... 15
      3.3.4 Hump yards ......................................................................................................................... 15
   3.4. Internal processes at flat shunting yards ................................................................................. 16
   3.5. Simulation of yards ............................................................................................................... 18
   3.6. Conclusion ............................................................................................................................ 20
4. SIMULATION METHODOLOGY .................................................................................................. 20
5. SIMULATION MODEL INPUT DATA ......................................................................................... 21
6. SIMULATION ................................................................................................................................. 26
   6.1. Description of the model ........................................................................................................ 26
   6.2. Important setup parameters for each object .......................................................................... 29
   6.3. Programmed logic .................................................................................................................. 32
   6.3. Explanation of the model ....................................................................................................... 44
      6.3.1. Entities .............................................................................................................................. 44
      6.3.2. Model ............................................................................................................................... 45
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>EXPERIMENT</td>
<td>47</td>
</tr>
<tr>
<td>8</td>
<td>RESULTS</td>
<td>47</td>
</tr>
<tr>
<td>8.1</td>
<td>Locomotive Transit Times</td>
<td>47</td>
</tr>
<tr>
<td>8.2</td>
<td>Total number of loads entering and exiting the plants in six months</td>
<td>48</td>
</tr>
<tr>
<td>8.3</td>
<td>Utilisation of locomotives</td>
<td>49</td>
</tr>
<tr>
<td>9</td>
<td>VALIDATION</td>
<td>50</td>
</tr>
<tr>
<td>10</td>
<td>DISCUSSION</td>
<td>62</td>
</tr>
<tr>
<td>11</td>
<td>CONCLUSION</td>
<td>63</td>
</tr>
<tr>
<td>12</td>
<td>RECOMMENDATIONS</td>
<td>63</td>
</tr>
<tr>
<td>13</td>
<td>REFERENCES</td>
<td>65</td>
</tr>
<tr>
<td>14</td>
<td>APPENDICES</td>
<td>68</td>
</tr>
<tr>
<td>14.2</td>
<td>Appendix A: Model processes</td>
<td>68</td>
</tr>
<tr>
<td>14.2</td>
<td>Appendix B: Standard Library Documentation</td>
<td>73</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Town L and Town M diagram ................................................................. 1
Figure 2: Company X Plant K rail network layout ............................................. 5
Figure 3: Company X Shunting yards in Plant K .................................................. 7
Figure 4: Scanned depiction of objects used in SIMIO (Pegden. 2009) ............... 13
Figure 5: Scanned depiction of object i.e. path and workstation used in SIMIO (Pegden. 2009) ......................................................................................... 13
Figure 6: Scanned depiction of object i.e. connector used in SIMIO (Pegden. 2009). 13
Figure 7: Scanned depiction of object i.e. TimePath used in SIMIO (Pegden. 2009). 14
Figure 8: Scanned depiction of object i.e. conveyor used in SIMIO (Pegden. 2009). 14
Figure 9: Arrival rate of Local Product E trains monthly ..................................... 23
Figure 10: Inter arrival time (in days) of Local Product E trains ......................... 24
Figure 11: Export Product E trains’ arrival probability ........................................ 25
Figure 12: Block Train arrival .............................................................................. 25
Figure 13: Block Train departures ...................................................................... 26
Figure 14: Company X shunting yard process .................................................... 29
Figure 15: Simulation model general information .............................................. 31
Figure 16: Locomotives ...................................................................................... 32
Figure 17: Properties of the locomotives ............................................................ 32
Figure 18: Block Train Product A (Entity 1) .......................................................... 33
Figure 19: Block Train Product B (Entity 1) ........................................................ 34
Figure 20: Block Train Product C (Entity 1) ........................................................ 34
Figure 21: Arrivals at Shunting Yard 1 ................................................................. 35
Figure 22: Arriving Product A by train at the Source 1 ........................................ 36
Figure 23: Arriving Product B by train at Source 2 .............................................. 36
Figure 24: Arriving Product B by train at Source 2 .............................................. 37
Figure 25: Properties of the Product E train ........................................................ 38
Figure 26: Properties of the transfer nodes ................................................................. 38
Figure 27: Properties of Source 2 ................................................................. 39
Figure 28: Line restriction at single track ................................................................. 39
Figure 29: Matching of products (Coupling) at shunting yard 3: combiner ............... 40
Figure 30: Properties of the separator at Shunting yard 3 (decoupling) .................... 40
Figure 31: Properties of a transfer node in Shunting yard 3 ..................................... 41
Figure 32: Properties of the transfer node in Shunting yard 3: inbound ..................... 41
Figure 33: Properties of the transfer node at Product B and C loading zones ............ 42
Figure 34: Properties of the transfer node at Product A loading zone ...................... 42
Figure 35: Properties of a server at Product B and C loading zones ....................... 43
Figure 36: Properties of the basic node at the locomotive base ................................ 43
Figure 37: Properties of a server at the Product A loading zone .............................. 44
Figure 38: Properties of the resources at Shunting yard 3, Shunt line 2 ................. 44
Figure 39: Company X shunting Process model ..................................................... 46
Figure 40: A 3D model of the rail network in the plant ........................................... 51
Figure 41: Product E train enters the loading zone ................................................. 52
Figure 42: Product E train offloads Product E at the Product E loading zone ............ 52
Figure 43: Product E train exits the Product E loading zone .................................... 53
Figure 44: The locomotive from the plant going to pick up the full train at Shunting yard 1 ................................................................. 53
Figure 45: The locomotive passes shunting yard 2 to pick up the full train at shunting yard 1 ................................................................. 54
Figure 46: A locomotive has stopped due to single track being occupied .................. 54
Figure 47: The locomotive collects the full train at Shunting yard 1 ......................... 55
Figure 48: The locomotive has moved the train from Shunting yard 1 to Shunting yard 3 .................................................................................. 55
Figure 49: The locomotive passes the single track with the full load to Shunting yard 3 .................................................................................. 56
Figure 50: Decoupling of RTCs is takes place at Shunting yard 3 .......................... 56
Figure 51: The locomotive is taking RTCs to the loading zone............................. 57
Figure 52: The locomotive is taking RTCs to the loading zone............................. 57
Figure 53: The locomotive drops off the RTCs at the loading zone....................... 57
Figure 54: The locomotive collect more sets of RTCs at shunting yard 3 to the loading zone in the plant .......................................................... 58
Figure 55: The locomotive collects more sets of RTCs at Shunting yard 3 to drops them off at the loading zone................................................................. 59
Figure 56: The locomotive drops off the RTCs at the loading zone....................... 59
Figure 57: The full train is taken back to Shunting yard 1 after loading coupling has taken place................................................................. 60
Figure 58: The locomotive with full train goes to shunting yard 1 passes a single track section ......................................................................................................... 60
Figure 59: The locomotive has dropped off the full train at Shunting yard 1......... 61
Figure 60: The locomotive makes a drop off at Product A loading zone ............ 61
Figure 61: Locomotive goes to pick drop more RTCs from Shunting yard 3 to the loading zone .............................................................................................................. 62
LIST OF TABLES

Table 1: Definition of terms ................................................................. ix
Table 2: The time RTCs ........................................................................ 48
Table 3: Total number of loads entering and exiting the plants in six months .......... 49
Table 4: Utilisation of locomotives ............................................................. 50
### Definitions of terms

Table 1: Definition of terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIMIO:</td>
<td>Statistical software package used to simulate.</td>
</tr>
<tr>
<td>Block Train:</td>
<td>Rail Tanker Cars or wagons which carry Company X Products from Town M to Town L.</td>
</tr>
<tr>
<td>Product E Trains:</td>
<td>Trains transporting Product E from Company X mine for export purposes.</td>
</tr>
<tr>
<td>RTCs:</td>
<td>Rail tanker cars or wagons used to carry products.</td>
</tr>
<tr>
<td>Shunting yard 1</td>
<td>Site where the Transnet Freight Rail (TFR) and Company X TRCs are exchanged.</td>
</tr>
<tr>
<td>Single line</td>
<td>A line used by Company X Block Train as well as international and local Product E trains.</td>
</tr>
<tr>
<td>TFR</td>
<td>Transnet Freight Rail are Transnet locomotives transporting products to Company X.</td>
</tr>
<tr>
<td>SARS</td>
<td>South African Revenue Services – tax collection.</td>
</tr>
<tr>
<td>Shunting yard 3</td>
<td>An exchange yard in the primary area where shunting occurs.</td>
</tr>
<tr>
<td>Product A</td>
<td>One of the products Company X produces.</td>
</tr>
<tr>
<td>Product B</td>
<td>One of the products Company X produces.</td>
</tr>
<tr>
<td>Product C</td>
<td>One of the products Company X produces.</td>
</tr>
<tr>
<td>Product A Loading and off loading zone</td>
<td>The site where Product A is loaded.</td>
</tr>
<tr>
<td>Product B and Product C loading and Off-loading zone</td>
<td>The site where Products B and C are loaded and offloaded.</td>
</tr>
<tr>
<td>Company X Town M Plant</td>
<td>The Company X production site in Town M.</td>
</tr>
<tr>
<td>Company X locomotives</td>
<td>Locomotives owned and used by Company X to transport products between exchange yard 1 to Shunting yard 3 and to the different loading and off loading zones in the Company X plant.</td>
</tr>
<tr>
<td>Transnet Locomotives</td>
<td>Transnet locomotives owned and used by Transnet to</td>
</tr>
<tr>
<td><strong>SPC</strong></td>
<td>Senior Process Controllers controlling trains as per schedules entering and leaving Company X plant.</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Product E</strong></td>
<td>One of the products produced at Company X.</td>
</tr>
<tr>
<td><strong>Car puller</strong></td>
<td>A small car used in the loading and offloading zones to place the RTCs or wagons between tanks after the locomotives have dropped off the RTCs.</td>
</tr>
<tr>
<td><strong>Pre Admin</strong></td>
<td>Administration done before loading and off-loading of products in the loading zones.</td>
</tr>
<tr>
<td><strong>Export Product E Trains</strong></td>
<td>International Product E trains transporting Product E from other countries to Company X Town M Product E section.</td>
</tr>
<tr>
<td><strong>Standard Library document</strong></td>
<td>Report detailing how the simulation model was developed.</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 Background Information

The study deals with the simulation of the railway system at Company X plant K which is situated in Town M. The shunting yard and product transportation in the plant and out of the plant is included (See figure 1).

Figure 1: Town L and Town M diagram

The study’s focus is on developing a simulation model which will simulate the current situation at Plant K. According to De wet (2008) simulation is a technique for modelling dynamics that can augment static value-stream analysis. For example, simulation can be used to estimate the effectiveness of alternative configurations of a lean business process prior to actual implementation. Simulation can often clarify the exact nature of the trade-off between custom satisfaction and cost-effective delivery of service, and allow the service provider to choose the right level of resources. Given the simulation explanation above simulation was chosen as appropriate tool for the analysis of this situation. In this study SIMIO software will be used in order to simulate Plant K railway networks.

The study includes the locomotives used during shunting, and the so-called Block Train. The Block Train is a train used by Company X to transport products from Town L to Town M; it consists of 40 (RTCs) or wagons. The products transported by the Block Train include Products A, B, C and D. The Block train transport these products between the two towns namely Town L and Town M.

Product A, B and C are used at Town L and M manufacturing plant to manufacture other products including Product D and all these products are transported by Block train. Apart from the Block Train there are other trains which transport other products. The RTCs transport more than one kind of product but each RTC only contains one product. The
products are loaded and offloaded at different locations in the plant (See figure 2 and 3). The RTCs are transported by Transnet locomotives between Town L and Town M. The other products are used to manufacture product A, B, C and D. There are other trains used to transport other products which are used in the manufacturing of product D. There are other products which are not transported by Block train which are being used in other offloading loading zone not included in the study.

The plants in Town L and M complement each other. The other products are being transported to different loading zones in the plant K. That implies that all the products are manufactured in Town L plant and also in Town M plant K depending on the capacity of the plant at a given time. The Block Train travels from Town L to Town M and back, and is pulled by Transnet Freight Rail (TFR) locomotives. Figure 2 shows the Company X rail network layout in the plant K used by Block Train and Product E train. The Transnet Company owned locomotives which drop off the RTCs at plant K shunting yard 1. The Company X locomotives take over from the Transnet locomotives from shunting yard 1 to move the RTCs to the plant Shunting yard 3 and to the different loading zones (See figure 2). The Block Train is divided in sets of ten maximum or less RTCs at shunting yard 3 (Figure 2). The reason for the train to be divided in sets of ten RTCs or less at shunting yard 3 is because the loading zones can only take 10 RTCs or less at a time and due to load destination. After the loading has taken place all the RTCs will be taken back to shunting yard 3 by a Company X locomotive from the different loading zones and then a full train will be assembled and taken to shunting yard 1 (figure 2). Shunting yard 3 has inbound lines and outbound lines for the locomotives and are situated in the plant. This is where the Block Train is assembled and parts (ten sets of RTCs or less) are moved to different offloading and loading zones. This is also where the locomotives park while they are idling (See figure 3).

The railway network in the plant is also used by both international and domestic Product E trains. Product E trains are from other locations nationally and internationally and these trains transport Product E to Company X Product E offloading and loading zone. Product E is used to manufacture product A, B and C and these products are then used to manufacture Product D. Product E trains and the Block Train share single track (figure 3) at the rail network at Company X plant K.
Senior Process Controllers (SPCs) controls the movement of trains in the plant K rail network. The SPCs control the movement of trains from shunting yard 1 to Shunting yard 3, and to the loading and off-loading zones, and back to Shunting yard 3. They also control the trains when they exit the plant (Figure 43).

Locomotives leaving the Company X plant are monitored to avoid collision with oncoming trains by Transnet Senior Process Controllers situated at Shunting yard 1 and Company X Senior Process Controllers (SPC) situated in the Company X plant K. The Transnet locomotives can only take RTCs as far as shunting yard 1 in plant K.

At the single track section trains have to wait for other trains to pass. There are control signals at this point which are also controlled by both Transnet and Company X Senior Process Controllers (SPC). Control measures are in place in the form of automated signals which control the movement of the trains on the single track. These signals use interlocking systems which means that once there is a locomotive using the line the signals lock automatically. Locomotives leaving the Company X plant are monitored to avoid collision with oncoming trains by Transnet Senior Process Controllers (SPC) and Company X Senior Process Controllers (SPC).

Products A, B, C and D are liquid and are transported in RTCs, whereas Product E is granular and is transported in wagons. Product E is transported to Company X Product E offloading zone by Transnet locomotives. Transnet locomotives transport product E wagons directly to the Product E loading zone. A Transnet Freight Rail (TFR) locomotive brings the train in. It then offloads and the train is taken out from the Product E offloading zone by the same locomotive. The Senior Process Controllers (SPC) communicates on a regular basis to control the movements of the locomotives from one site to the other. The SPCs also control the movement of product E trains in and out of the plant K. The SPC communicates with the Transnet Freight Rail shunter/controller located at Shunting yard 1. There is no exchange of locomotives that takes place for Product E trains.

The maintenance of locomotives takes place at the locomotive shed where they are serviced and maintained (See figure 3).

The research is concerned with developing a simulation of the activities of the Block Train, Product E trains arriving and leaving the Company X plant, and the shunting taking place. The restrictions in the model have been used as signals to control the movement of trains at
a single track (See figure 3). Simio object resources have been used to restrict the movement of trains in the single track section.
Figure 2: Company X Plant K rail network layout
The Block train and the shunting areas will be simulated in order to show the exchange of RTCs from the Transnet locomotives to Company X locomotives.

The Block train comes with Product A; B, C from Town L. These products are used to manufacture Product D and other products. All the products are loaded after they have been manufactured.

Two locomotives are used to transport the RTCs from Shunting yard 1 to shunting yard 3 in the primary area of plant K, and back to Shunting yard 1. There are other loading zones in plant K which are used by other trains other than the Block train.

Figure 2 shows the rail network of Company X Plant K where all the loading zones are included. It includes loading zones which are used by the Block Train and other trains delivering other products in the plant. The other trains use the same rail network as Block train to transport other products to the different loading zones in plant K.

The Block Train arrives in Town M on Monday, and leaves on Wednesday. It will return to Town L from Town M on Thursday, and then leave on Sunday. The trains arrive according to schedules, and loading and offloading is controlled by Senior Process Controllers (SPCs). The train schedules are used to enable the smooth transportation of these products. They also assist Transnet in knowing how many locomotives to allocate for that particular day, and at what time. The SPCs control the movement of trains from Shunting yard 1 to shunting yard 3 and to the different loading zones in plant K.

Car-pullers are used in all the loading and offloading zones. They are used because the locomotives are large and will not fit in the required locations. The capacity of the loading and off-loading zones has limitations. The loading zone can only handle ten RTCs at a time. The Car-puller is used in the loading and offloading zones to place the RTCs or wagons next to the tanks after the locomotives have dropped off the RTCs (See figure 3). The product from the tanks will be pumped into the RTCs.

A simulation model has been developed to simulate the loading and unloading of products. SIMIO software has been used to simulate the Company X, Plant K railway network.
Figure 3: Company X Shunting yards in Plant K

- Locomotive sted
- Loading zone for other products
- Loading zone for other Products transported by other Trains
- Loading zone for other Products transported by other Trains
- Loading zone for other Products transported by other Trains
- Shunting yard 1
- Shunting yard 2
- Shunting yard 3 (assembling and disassembling of Block Train)
- Company X locomotives parking
- Loading zone for other Products transported by other Trains
- Product E trains entrance and exit
- Single track
- Product E offloading Zone
- Product E loading and offloading zone with 5 lines (Line 1 to 5)
- Product B and C offloading zone

Pull car location. Maximum 10 RTRs at a time.
Figure 3 shows how the Block Train arrives at Company X plant K. The Block Train arrives at Shunting yard 1. The Transnet Freight Rail (TFR) locomotive detaches the RTCs at shunting yard 1, and then the Company X locomotives take them to Shunting yard 3 in the primary area (See figure 3). The Company X locomotive with the Block Train will pass at the single track section which is also used by Product E trains as illustrated in figures 2 and 3.

It is important to note that safety is one of the important considerations when modelling rail transportation systems focused on controlling the movement of RTCs. Safety precautions have to be in place all the time to avoid collisions. Only one train is allowed to use the line at any given time. The SPCs are there to ensure that the movement of trains in the Company X railway network is well managed. An automated signalling system is used to control access to the line. In the simulation model it will be assumed that there is one train travelling on a certain line at a particular time. No two trains going in the opposite direction are allowed to enter the single track. The transit speed along the track is 15km/h.

1.2. Problem statement

Management were concerned about inefficiencies at Company X’s plant K shunting yard. Service times and the idling times of the locomotives were considered too long. Locomotives can and should also be utilised for other purposes, besides shunting, in the plant. This has implications for resource planning and productivity in the company.

A method to identify inefficiencies and to test alternative system configurations was needed, such that cost-benefit studies could be undertaken. Therefore, a computer simulation model of the system was needed.

Therefore problem under investigation is whether the Plant K rail system can be successfully modelled using simulation. The simulation model will focus on Block Train RTCs coming from Town L to Town M Company X Plant K. Shunting within Plant K will also be included in the model. The model will focus on Block Train RTCs entry into the plant, and how they are transported to the different loading and off-loading zones within the plant and also Product E trains and other trains. Simulation was chosen as the method of analysis as it can provide measurable metrics for assessing the performance of a model and SIMIO software was chosen as the simulation tool for the study. In this case the network is developed by putting together the entities and these entities are joined by defining the behaviour of the nodes in the network and
then connecting the nodes together using defined connections. This is done in order to develop a simulation model that will illustrate what is actually happening in Company X Plant K.

2. OBJECTIVE

The objective of the study is to evaluate whether the Company X Plant K rail network system can be simulated such that the resultant model can be used to compare system design changes.
3. LITERATURE REVIEW


The techniques and tools of simulation modelling have been used to research and investigate various systems in a wide range of areas such as commerce, computer networks, defence, health, manufacturing and transportation (Taylor et al, 2009). Simulation is one of industry's most used operations research techniques. Its uses range from answering questions on work-in-process and production feasibility, to comparing alternative plans for system routing and scheduling. Opinions about its efficacy as an industrial decision support tool vary greatly in spite of (or because of) its broad user base (Cochran et al, 1995).

A number of studies have found simulation to be one of the most effective management science methods used today (Christy and Watson, 1983). According to Hwarng (2001) decision makers often use simulation because of the complexities of their problems. Simulation helps business managers design their business services and processes, just as it has helped engineers to design products and processes.

3.2. SIMIO

3.2.1. What is SIMIO?

Pegden (2009) defines Simio as a tool for building and executing dynamic models of systems in order to see how they perform. It acts out and displays a 3D animation of a system over time. Simio enables a user to see the proposed systems in operation before building and changing them. It makes modelling dramatically easier by providing a new object-based approach. The user selects objects from libraries and places them graphically in the model. Objects represent the physical components in the system, such as workstations, conveyors, and forklift trucks in a manufacturing facility, or the gantry in a hospital emergency room (Pegden 2009). Object-based modelling is a very natural and simple approach to simulation modeling. Simio makes 3D animation a simple and natural part of the modelling process. By using Simio a model can be laid out with realistic spatial relations that depict the real life system accurately (Pegden, 2009).
SIMIO uses processes to define steps. Examples can be found in Appendices A. Long wording in the software will not be stated on full. There software use a shorthand and abbreviations where is necessary.

3.2.2. Benefits of simulation

Pegden (2009) points out the following benefits of using Simio:

- Simio simulation enables the user to see the impact of change (Pegden. 2009).
- With Simio simulations the user can make changes to the model quickly to test out ideas, without disrupting the real life system (Pegden. 2009).
- With Simio simulation the user can make mistakes in the model, and not in the real life situation. Simulation enables the user to separate the improvements from the failures and optimize his/her situation (Pegden. 2009).
- Simio simulation enables the user to validate proposed designs and make the best use of limited capital to focus resources where they will make the most impact on results. Simio simulation brings each individual user’s ideas to life, providing an animated preview of a proposed change (Pegden. 2009).
- It provides records, and graphically displays key performance measures for each system. This helps in analysing proposed changes, as well as communicating the benefits of those changes to the other stakeholders (Pegden. 2009).
- A Simio simulation allows the user to account fully for variations in each system and the impact these will have on the system performance overall (Pegden. 2009).
- Simulation assists the user to avoid the critical problems created by applying traditional static analysis in an attempt to understand and predict the behaviour of a variable and complex dynamic system (Pegden. 2009).

3.2.3. Definition of objects in SIMIO.

According to Pegden (2009) the definitions of objects in Simio are stated below (figure 3).

A source object is used to generate entity objects of any type (Pegden 2009).

A server object is used to define time processing (Pegden 2009).
A **sink object** is used to destroy entity objects which have finished processing in the model (Pegden 2009).

A **combiner object** is used to model a process matching the multiple entity objects, forming those entities into a batch, and then attaching the batched members to a parent entity (Pegden 2009).

A **separator object** is used to model a process matching the multiple entity objects, forming those entities into a batch, and then separating the batched members to a parent entity object or making copies of an entity object (Pegden 2009).

A **entity object** is used to represent an object.

A **vehicle object** is used to transport an entity object between node locations. An “on demand” routing type vehicle can be used as a movable resource that is seized and realised for non transport related tasks by the model process logic (Pegden 2009).

A **worker object** is used as a movable resource which is seized and realised for tasks by the model process logic. A worker can also be used to transport entity objects between node locations (Pegden 2009).

A **resource object** is used to model resource constraint with capacity that can be seized and realised by other objects (Pegden 2009).

A **transfer node** is used as an entity to transfer into and out of the object. The object may also be used to define the intersection point in the network of links. Optional entity destination selection and transport logic is provided (Pegden 2009).

A **basic node** is used by entity to transfer into and out of the object. The object may also be used to define the intersection point in the network of links (Pegden 2009).
A path is used to define a pathway between two node locations where the travel time is determined by the path length and travel entity speed. Passing may or may not be allowed (Pegden 2009). (Figure 5)

A workstation object is used to model a constraint resource which has a single capacity. Each operation processed at workstation is performed as a sequence of three activities, that is, setup, processing and teardown. An optional secondary resource makespan and material requirements may be specified (Pegden 2009) (Figure 5).

A connector is used to define a direct, zero travel distance connection from one node location to another (Pegden 2009). (Figure 6)
A timepath used to define a pathway between two node locations where the travel time is use-specified (Pegden 2009). (Figure 7)

![Figure 7: Scanned depiction of object i.e. TimePath used in SIMIO (Pegden. 2009).](image)

A conveyor is used to move travel entities between two node locations using an accumulating or non-accumulating conveyer (Pegden 2009). (Figure 8)

![Figure 8: Scanned depiction of object i.e. conveyor used in SIMIO (Pegden. 2009).](image)

3.3. Railway Shunting yards

3.3.1 Definitions and descriptions of railway shunting yards

According to Marinov et al. (2009) yards are a very important part of railway freight operation. Depending on the yards’ performance, the quality of the railway freight network operation could be improved. Yards dictate freight train movement on the network. They consume a significant amount of the available resources in order to receive, accommodate and send freight trains as a provided service. In other words, the yards execute reassembly processes with freight trains based on the delivery requirements of the customers. The yards are, however, characterized by a limited number of tracks, meaning that their performances are confined by bounds. These performance bounds identify the yards’ processing capabilities, and when they are stretched, then the yards malfunction. This phenomenon contributes to rail freight services of poor quality (Marinov et al. 2009).
A marshalling yard or shunting yard is generally a place where goods trains and other loads such as wagons coming in from nearby shunting yards are received, sorted according to a plan, and new trains formed and dispatched onwards. Marshalling yards were indispensable in earlier times, and were built at all strategic junctions and intersections. Their need arose because of an inherent feature of goods transportation by rail, namely, that goods arriving at a loading point did not always materialize in such quantities to make up a full train load at a time, and even when a train-load of products was offered, the material might not all be booked for the same destination (Bhalerao. 2008).

According to Bart (2007), average train lengths in Europe currently vary between 600 and 700 meters. This might be increased up to 1,000 meters, but the maximum length is limited by double locomotive power which is rarely deployed. At the hub-exchange facility, either rail wagons (at a shunting yard) or load units (at a terminal) are exchanged. In terminal operations, trains remain as they are all the time, without decoupling, and cranes and an internal transport unit transfer load units from one train to another.

Bart (2007) and Bhalerao (2008) state that four different hub-exchange facilities exist; Flat shunting yards, Hump shunting yards, Road-rail terminals (intermediate terminals), and New-generation terminals.

### 3.3.2 Flat yards

Flat-Shunting Yards are yards in which the tracks lead into a flat shunting neck at one or both ends of the yard, where the freight cars are pushed and pulled by a shunting engine which sorts onto the assigned track (Marinov et al. 2014).

### 3.3.3 Gravity yards

According to Marinov et al. (2014) wagons are shunted with the help of gravity and therefore there is no need for special shunting engines. Only one engine is needed to push the wagons on top of the hump. Gravity yards are cheap because of the natural topography.

### 3.3.4 Hump yards

In this type of yard the wagons of a complete train will roll down an incline to get reassembled for their departure. To achieve this kind of process the yard must have a specific layout. Hump yards
consist of four different components: the receiving yard, hump, classification yard and departure yard (Marinov et al. 2014).

Most hump shunting yards are one-directional meaning that trains can enter the facility from only one direction. The number and length of tracks might vary in each yard, depending on the required capacity. The shunting hill area most often consists of one track leading to the shunting hill. A large yard might have two shunting hills. They have automated switches downhill, leading to the sorting and departure yard. Tracks at the beginning of the sorting yard, and the middle of the sorting yard are equipped with automated braking equipment. Large yards might have installed automated equipment for train assembly between the tracks in the sorting yard. Arrival yards and sorting/departure yards are equipped with one or several locomotives (Bart .2007).

3.4. Internal processes at flat shunting yards.

Flat yards show the three basic aspects of every yard: the reception yard, the classification yard, also known as the sorting yard, and the departure yard. A goods train is received in the reception yard, and the engine is sent to the locomotive shed. Adjacent to the reception yard, is the classification yard, where each line is reserved for wagons going in a particular direction (Bhalerao. 2008).

Marshalling yards are selection and delivery points which include flat yards, stations in which the initial train is split, cargoes are sorted according to their destination, and new trains formed. The cargoes are situated near railway stations, being a direct continuation of them. Hazards found in the yards are associated with the activities in a marshalling yard which originate from the hazardous nature of the substances handled (Christou.1999).

According to Lehnfeld, and Knust (2014) storage of items leads to different kinds of optimization problems. The unloading problems arise if the outgoing items need to be retrieved from the storage area. Therefore it has to be decided which items will leave the storage, and in which specified order. Before marshalling occurs the items have to be sorted inside the storage area so that all of them can be retrieved without any further reshuffling afterwards. If incoming items need to be stored while outgoing items need to be retrieved, combined loading/unloading problems might arise.

In Sendera, and Clausena (2011) flows of single wagons with different origins and destinations are consolidated on their way through the network via at least one shunting yard. These yards are part
of hierarchical network, while their size, function, technical equipment, and shunting process might differ. These factors influence the capacity of the formation yards. The wagons are transported through the network using several railway engines (trains). The capacity of these trains is limited. The presented hub location problem covers a lot of the specific characteristics of wagonload traffic.

Lin (2012) presents a formulation and solution for the train connection services (TCSs) problem in a large-scale rail network in order to determine the optimal freight train services, the frequency of services, and the distribution of classification workload among yards. The TCS problem is modelled as a bi-level programming problem.

A Cozzani et al (2007) state that the risk analysis of marshalling yards is complex, because of the variety of operations to which the railcars are subject to. A first possibility is that a train might simply pass through the marshalling yard, usually at reduced speed. Secondly the train might be subject to a simple stop and stay, leaving after some time without undergoing any other operation. Thirdly the train might be subject to a change-over of locomotive or shunting procedure, which consists of the splitting of the arriving trains and the formation of new ones.

In Motraghi (2013) the focus is on rail yards and terminals in order to ensure the movement of passengers and freight in the railway network system. It is important to note that most of the transit time is spent in transhipment and shunting operations. Continuous improvement of processes is important and should always be optimised, in order to ensure that yards and terminals keep up with changes taking place. These days the materials transported by rail have changed from being heavy goods, to light-weight, palletised goods, and this has had an effect on both trains and terminals.

The optimization of railway systems leads to a variety of planning and scheduling problems. For instance, when several freight trains arrive at their terminal railway stations, it is often necessary to split them, and rearrange their cars into other trains. Of course this should be done quickly and with only a limited number of moves, while observing all the restrictions arising from the local infrastructure. Such train rearrangement problems have been modelled, classified, and surveyed. The main result states that arranging a departing train in a depot is NP-complete, even if each track in the depot contains only two cars. The study proved that even the following two highly restricted special cases of this problem are NP-complete (1) the case where every track contains at most three cars and (2) The case with at most two car types (Eggermont et al 2009).
Javadiana (2011) states that a yard capacity study assesses how many cars, blocks and trains can be handled within the existing infrastructure and resources. An optimisation formulation and a solution procedure for the determination of the capacity of rail yard stations had been developed with respect to the optimal number of shunting and reclassification operations at the nodes of the rail network. In order to resolve this problem, a simulated annealing (SA) algorithm was proposed. This had to work efficiently on a neighbourhood search within the space, acceptance probability, and inferior solutions. It resulted in providing a new managerial tool for planning and analysing yard capacity more effectively and efficiently.

Javadiana (2011) describes a yard analysis tool. The newly developed model provided network information, such as yard capacity, unmet demands, and number of loaded and empty rail car at any given time and location.

In Nel et al (2004) coal is transported by Spoornet, the national rail operator, from approximately 40 collieries in the Mpumalanga and KwaZulu-Natal provinces of South Africa. Trains can only carry a maximum pay-load of 16,800 tonnes, and a standard length of 2.7 km consisting of 200 wagons. Up to 24 trains a day can use the system. When the line was built it was envisaged that the line would carry approximately 2.5 million tonnes of coal per annum; today it carries almost 70 million tonnes to different mines in South Africa and abroad. This figure is projected to rise to over 80 million tonnes from the Richards Bay Coal Terminal (RBCT). It is predicted that this will increase the value of coal exports by US$400 million per year through a joint venture between RBCT and the National Ports Authority. The RBCT is the biggest single customer of the South African rail system, accounting for approximately 50% of gross tonnage moved (Nel et al. 2004). The coal is being exported.

3.5. Simulation of yards.

In Marinov et al (2011) yards play an important role in the quality of rail freight services. These facilities function as reassembly hubs in rail freight networks and incorporate a significant amount of static and dynamic resources. Yards are considered an element which adds little value to the final product from the customer’s perspective. Yards are regarded as a main source of delay and loss of business from the yard operator’s viewpoint. Therefore, running yards inefficiently is not acceptable. Marinov et al (2011) studied a double-ended flat shunted yard using G/G/m queues. The results demonstrated significantly low utilisation levels of the yard subsystems in question.
Therefore, possible improvements through changes in traffic rules and production schemes are discussed.

Marinov et al (2013) focus on organising, planning and managing the train movement in a network where three levels of management for rail planning are introduced: strategic, tactical and operational, and are then followed by decision support systems for rail traffic control. Marinov et al (2013) further discuss train operating forms, railway traffic control and train dispatching problems, rail yard technical schemes that are the operations in the network and performance of terminals, as well as timetable design. A description of simulation techniques and specific computer packages for analysing and evaluating the behaviour of rail systems and networks is also provided (Marinov et al. 2013).

Marinov and Viegas (2009) write about simulation modelling methodology for analysing and evaluating flat-shunted yard operations. A yard simulation model has been created and implemented using a computer package for event-based simulation, SIMUL8. The main idea behind the simulation modelling approach is to simulate the flat-shunted yard operations dividing the yard into segments so that each aspect of each segment can be described and analysed separately.

In Marinov and Viegas (2009) the objectives of the study were to provide an adequate methodology accompanied with a reliable tool for estimating, analysing and evaluating the performance capabilities of rail yards using an appropriate approach for the purposes of a rail freight operator.

Di Stefano (2004) proposes a graph theoretical approach to the problem of train shunting in a railway depot. The focus is at night where the trains have to be parked in a shunting depot in such a way that the operations can start as smoothly as possible the next morning. In general this is a most difficult problem, and includes many sub-problems. The study further focuses on the sub-problem of how to arrange the trains in the correct order on the available tracks to avoid shunting operations for outgoing trains in the morning.

García, and García (2012) describe a simulation-based flexible platform developed to support strategic and tactical decision-making related to terminal design and redesign. The study concentrates on interior terminals which interchange containers between rail and road transport. In addition, these terminals might have a classification yard which could have various uses. It could
be used to exchange railcars between trains to brake long trains which cannot be processed in the loading/unloading tracks of the terminal, or as a waiting area for incoming and outgoing trains.

Boysen et al (2010) state that efficient computerized scheduling procedures are indispensable, in order to enable rapid container transhipment between freight trains in modern rail–rail transhipment yards. A dynamic programming approach is used in this regard. It determines yard areas for gantry cranes, so that the workload is spread evenly among the cranes and, thus, train processing is accelerated. In the simulation of transhipment yard operations, the effect of optimal crane areas vs. equally sized areas was examined. The results indicated a remarkable speed-up in train processing if optimal crane areas were applied (Boysen et al.2010).

3.6. Conclusion

The literature deals with locomotives and shunting areas. Most companies highlighted in the literature review intended to improve the manner in which the railway system, that is, the shunting was being handled by introducing new technologies or improving the current internal processes. The focus was on investigating other ways of improving the process and service speed in different terminal or loading and off-loading zones. Simulation has been successfully employed in a number of studies.

4. SIMULATION METHODOLOGY

Simulation input data was collected by Operations Research and Rail representatives at Company X through observations. The simulation results will help in making informed conclusions and recommendations on how the Company X Plant K railway system works. The simulation model will be used with the intention of improving the process of trains shunting, travelling and loading and offloading products at Company X, plant K. This would help to improve the loading and offloading times while allowing the process to run smoothly without delays. These would include aspects such as waiting time within its operating centres and generally improving the service rate in the work stations. The investigation will focus on simulating shunting yards which include arrival and departure times of RTCs and locomotives at the plant.

Simio Software will be used to simulate primarily the Company X trains and the movement of the products they carry.
By effectively reducing the bottlenecks within the process, the improved Company X railway network would lead to better returns on investment. The railway network process and how the simulation model of this rail network was developed will be explained in this report.

SIMIO software was chosen as the simulation tool for the study. In this case the network was developed by putting the entities together. These entities were joined by defining the behaviour of the nodes in the network and then connecting those nodes using connections. The simulation output data can be captured and evaluated during the simulation. This data can also be monitored during the run. The following aspects of the simulation data will be excluded for the purpose of this study:

A. The loading of Product E trains will only be modelled as a delay of the Block train at the single track section in the rail network.
B. Movement of Transnet Freight Rail (TFR) locomotives
C. Shunting details at Shunting yard 1
D. Production processes and storage of products
E. Failures/accidents of vehicles en route to destinations
F. Workforce absenteeism
G. Breakdown/failures of loading/offloading equipment

5. SIMULATION MODEL INPUT DATA

The data used in this study was collected by the Company X, Operations Research Department in 2013. The data was extracted from observations made on the Plant K rail network system by Operations Research and Rail department.

- Shunting of RTCs transporting Products A, B, C and D takes 15 minutes minimum, average 20 minutes and 30 minutes maximum at all the shunting yards. Product D loading takes place in lanes 1, 2, 3 and 5 in the loading and offloading zone in plant k. Products A, B and C loading and off-loading take place at line 4 of Product A, B and C loading zones.
• Administration (the completion of paper work) takes 35 minutes before the loading of Product D. Administration takes 20 minutes before loading and offloading Products A, B and C.

• Coupling of locomotives to Product A, B C and D RTCs take a minimum of 1 minute and a maximum of 6 minutes. On average this takes 3 minutes. The coupling of RTCs takes place at Shunting yard 3. The RTCs will then be transported to different loading and offloading zones in sets of ten RTCs.

• Detaching locomotives (decoupling) from RTCs for all the products takes 1 minute to 5 minutes. On average detaching of locomotives from the RTCs takes 2 minutes for all the products. The time it takes to detach locomotives is the same in all the loading zones.

• Post-loading administration (paperwork) takes 2 minutes minimum; 2.5 minutes average, and 4 minutes maximum. This takes place at Products A and Product B, C and D loading and off-loading zones (Figure 3).

• Weighing of RTCs takes a minimum of 2.5 minutes; and 4 minutes maximum, and on average 3 minutes. Weighing RTCs takes an extra 20 minutes after all the RTCs have been loaded. Product A is only product that is weighed the RTCs which carries other products are not weighed.

Additional information

• There is no delay associated with the changing of shifts. Working hours for each shift at Company X are from 6:00 to 18:00 Sun-Sun.

• Shunting yard 1 is considered to operate 24/7 because the Transnet Freight Rail (TFR) driver placing trains in the yard should be at the shunting yard every day of the week.

• The distance travelled by the locomotives with the Block Train RTCs from Shunting yard 1 to shunting yard 3 and to Products B, C loading and offloading zones is 8.7 km, and to Product A and D loading and offloading zone it is 8.5 km.

• The Block Train weighs 2000 tons, and has 40 RTCs, each weighing 50 tons.

• The locomotive minimum speed in the primary area is 15 km/h. Maximum speed allowed in the primary area is 20km/h. for the purpose of the study 15 km/h speed has been used in the model.
- Company X uses two 2 locomotives to transport the Block Train RTCs from Shunting yard 3 to the Products A, B, C and D loading and off-loading zones in Plant K (figure 3).
- Product A is the only product that is weighed. It is only weighed at the loading zone.
- The locomotives will drop off the RTCs close to the pump tanks and the Car-pullers will then be used to move the RTCs in the loading zone from one point to another. The pump tanks are the tanks used to store products. These tanks have pumps that are used to fill up the RTCs with the products.
- The loading zones can only take one set of 10 RTCs at a time.

Figure 9 shows observed data for one Product E trains with 30, 35, 38 and 50 RTCs respectively. Ten Product E trains with 40 RTCs were counted a total of 14 Product E trains were observed.

![Bar chart showing number of trains vs Product E trains RTCs]

Figure 9: Arrival rate of Local Product E trains monthly

Figure 10 shows observed frequency of Product E train arrival. Out of a total of 7 observations number of Product E train inter arrival times of 4, 5 and 7 days were observed.
Figure 10: Inter arrival time (in days) of Local Product E trains

Figure 11 shows the frequency of Export Product E arrival. In this graph there is a higher likelihood that an Export Product E train will arrive after the first one has arrived between two days in Town M. If there are 4 to 6 days then the chances of an Export Product E train arriving diminishes. There other product E of a certain quality will be exported. The product E that is been exported is from brought in by local product E Trains from different mines nationally.
Figure 12 states that Block Train arrives on Mondays and Thursday and leaves. One Block Train will arrive and leave plant K weekly. Meaning 4 Block Train loads will arrive every Monday of each month and 4 Block trains will leave every week Thursday of each month. The other 4 Block trains that arrive and leave each month to bring the number to 8 are explained in figure 16.

Figure 13: the Block Train leaves Company X plant situated in Town M on Wednesdays and Sundays. Each load of the Block Train has 40 RTCs. Two Block Train loads will leave every week.
Block train arrives on Monday and Thursday and leaves plant K on Wednesday and Sunday.

6. SIMULATION

6.1. Description of the model

The network was developed defining RTCs as entities which move through nodes, links and other SIMIO objects which define the network. RTCs in the model are represented as entities. In the SIMIO objects, events are defined using properties and add-on processes, e.g. how long the locomotive is going to take before it can collect the RTCs, as well as defining how long the RTCs are going to take before they can be connected to the locomotives at the shunting areas.

The simulation model included the site where the locomotives would be parked when the loading and the offloading took place. The simulation developed evaluated the performance of the model and its effectiveness.

The model developed followed the process in figure 14, beginning with the freight trains’ arrival at plant K, and Shunting yard 1. Figure 40 in page 50 shows a layout of the model under Model validation.

The Block Train arrivals were generated using entities arriving at Shunting yard 1 shown as Source 1 at the arrival point. The entities enter Shunting yard 1 through this source (Figure 40).
These entities represent the Block trains’ arrivals at Shunting yard 1 and are according to the arrival table (Figure 15). Trains arrive on Monday and Thursday and the trains arrive in the morning around 07:00. The source represents the entrance to the shunting yards. A full train or load will arrive in Shunting yard 1 with a Transnet locomotive. Bidirectional paths were used to represent railway lines. They were connected by basic nodes.

As soon as the entities which represent the Block Train arrive at Shunting yard 1, the Company X locomotive parked at the locomotive parking area located in Shunting yard 3 will collect the Block Train, at Shunting yard 1. The locomotives have been depicted by vehicles which were changed to represent the real locomotives in the model.

The rail line used by Product E trains meets the rail line used by the Block Train at basic node 22 in plant K (figure 40). All the lines represented in the shunting yards are used for shunting. Product E trains enter the plant through Source 2 and proceed to Server 1 which represents the Product E offloading area at plant K. After unloading, the Product E trains will leave the plant through Sink 2.

Restrictions in the model were represented by means of resources. These allow the trains to use the line in the rail network without colliding (Figure 40). These resources serve as control systems and control the movement of the locomotives on the single line. The resources ensure that once the train is already on the line, no other train will be allowed to enter the single line. The single track starts at Basic Node 22 which serves as an intersection, and ends at Basic Node 23 a node immediately after basic node 22 (Figure 40). All the railway lines in the model are bidirectional; i.e. they allow movement in either direction.

Server 2 is the loading and offloading zone for the other products which are not being transported by the Block train. The products offloaded and loaded at server 2 are products transported by other trains. The depots in the diagram represent offloading and loading zones.

Server 5 represents the loading and offloading zones used by other trains. These trains are transport other products within plant K. Shunting yard 3 has inbound lines and outbound lines for locomotives. This is where locomotives are divided, and taken to the different off-loading and
loading zones. The locomotive base is represented by a basic node (Loco Base). This is where the locomotives will park while they are idling (Figure 40).

Servers 7 and 8 represent loading and offloading zones used for other products which are not transported by Block train. Server 13 represents the loading and offloading zones for both Products A and D. The transfer nodes in the model are used to indicate locomotives directions and destinations. Processes are used to show how the movement and the model logic should take place (Appendix A).
6.2. Important setup parameters for each object.

Figure 15 shows RTC destinations and refers to the locations where loading and offloading takes place. Input@ProductA loading refers to the location where product A is loaded. Input@ProductA loading is a server located at Product A loading zone. Input@ProductsB and ProductC loading refers to the area where loading of products B and C take place. Input@ProductsB and ProductC loading is achieved using a server located at Product B and C loading zone.
Under coupling destinations in Figure 15, ParentInput@ShuntLine1 Outbound and ParentInput@Shuntline2 ShuntLine2 Outbound Shunting Yard 3 shows the places where coupling takes place after the products have been loaded in shunting yard 3. Triangular distribution has been used in the study in order to control the coupling and decoupling of entities in the shunting yard 3.

Inbounds at Shunting yard 3 depict a place where decoupling of RTCs will take place so that each set of RTCs can be moved to the different loading zones. The shunting of the locomotives takes place at this point in the rail network. The RTCs will be taken to different loading and offloading zones from shunting yard 3 at the inbounds and outbound locations.

Figure 14 shows the arrival table which indicates how the Block train will be arriving for the six month period in which the simulation is run. This is done taking into consideration the number of days in a week in a month. Under products in Figure 15, different products transported by the Block train are shown. These are Products A, B and C. All the products have an equal probability of 0.333 to reach the loading zones and the arrival table figures allow products to arrive as expected during the period when the model will run. Product A load destination is 1, and its product type is 1. Product B load destination is 2, and its product type is 2. Product B load destination is 2 and its product type is captured as 3.
In figure 16 a transfer node setup is shown. BM Loco [Entity1.Which loco] refers to the locomotive to be used at the point when the locomotive is required to transport the load into the primary area. The parameter RideOnTransporter mean the RTCS are pulled by a locomotive which is set to (True) to allow this to happen. EnteredAddOnProcess mean that the RTCs exiting the shunting yard are pulled by a locomotive. This will enable the locomotive to go to the correct destination. ExitedAddOnProcess is the destination in the simulation model where the locomotive will exit the Shunting yard 1 plant K. Path 162 is a path the locomotives use to move to a destination, being basic node 52. Connectors and nodes are used to direct the movements of the locomotive.
6.3. Programmed logic.

Figure 17 shows the properties of the Company X locomotives. The entities in the simulations represent the Block train, and the products the train transports. The transport logic has been set at 1, the loading capacity of the locomotive. This is done in to allow the locomotive to carry one train at a time. The travel speed has been set at 15 kilometres per hour. Routing logic indicates how the locomotives should move. In case the locomotive is not needed, it will park at the Loco Base. The population set is 2, meaning two locomotives have been used for the simulation.
Figure 18 shows the properties of entity 1 (Product A). The travel logic has been set to allow the entities to be able to enter the rail network, and exit the network. The entities are set up to ride on the locomotive to the different destinations in the plant.

Figure 18: Block Train Product A (Entity 1)

Figure 19 shows the properties of another entity 1 (Product B).
Figure 19: Block Train Product B (Entity 1)

Figure 20 shows the properties of another entity 1(Product C).

Figure 20: Block Train Product C (Entity 1)
Figure 21 shows the entity arrival logic where the Block train will arrive according to the arrival table at Shunting yard 1 with the different products. The repeated arrival has been included so that the arrival of the entities might be repeated depending how long the simulation will run. The entities arrive at Shunting yard 1. An explanation of this is captured under products in figure 15 where two Block train arrive in Plant K per week. A total 8 trains arrives in Plant K with expected number of products in a month.

Figure 21: Arrivals at Shunting Yard 1

Figure 22 shows the arrivals of the block train with the different products, in this case the logic concerns Product A. It is set up to allow for the arrival of the Block train with Product A and inter arrival has been set up to be random and exponential 7, given the number of days in a week. Entity arrival (the number of products arriving by Block train) is 1. The model has been setup to cater for 24 Hours in day and a seven day week.
Figure 22: Arriving Product A by train at the Source 1

Figure 23 shows Product B logic. The source is set up to allow the arrival of the Block train with Product B and inter arrival has been set up to be random and exponentially set at 7 days, given the number of days in a week. Entity arrival: the number of products to arrive by Block train is 1.

Figure 23: Arriving Product B by train at Source 2
Figure 24 shows Product C logic. The source is set up to allow the arrival of the Block train with Product C, and inter arrival has been set up to be random and exponentially at 7 given the number of days in a week.

Figure 24: Arriving Product B by train at Source 2

Figure 25 shows the properties of the Product E train that is, the logic. Initial ride capacity is set up to allow one load of Product E to be transported by the Product E train. Product E Locomotive parks at output source node when it is not in demand. Product E trains are transported by TFR locomotives.
Figure 25: Properties of the Product E train

Figure 23 shows the properties of the transfer node at the arrival source at Shunting yard 1 (figure 40). The entity destination has been set up to be specific to allow for the travelling time of the locomotive to a specific destination, that is shunting yard 1 inbound. The entity concerns Product A.

Figure 26: Properties of the transfer nodes

Figure 27 shows the properties of source 2. Product E train enters the plant to deliver product E, at Product E loading zone. The arrival has been set up to be random exponentially distribution of 0.25. The number of entities to arrive at the source is 1, and the maximum arrival is set at 1, meaning not more than one product type can arrive at a time.
Figure 27: Properties of Source 2

Figure 28 shows the line restriction source. The resource in this case is used to restrict the movement of the trains on the single track, where only one train is allowed to use the track at a time. This resembles the automated signals in the rail network.

Figure 28: Line restriction at single track

Figure 29 shows the properties of the combiner at the Shunting yard 3. The combiner is set up in such a way that it will couple entities in this case the RTCs with the different products at Shunting yard 3. The RTCs will be combined until they form a complete load with 40 RTCs. The full load will then be taken to shunting yard 1 by a Company X locomotive. The restriction on the single track follow a first in first out principle, which means only one locomotive entered the single track first will be allowed to use the track at that time no other train will enter.
Figure 29 Matching of products (Coupling) at shunting yard 3: combiner

Figure 30 shows the properties of the separator. The separator has been set up to make copies to enable the train RTCs to be divided into sets of ten RTCs. The RTCs are transported in sets of ten RTCs because of the limited number of RTCs the loading zone can handle at a time. Decoupling the Block train takes place at Shunting yard 3 and this is made possible by the parameters set up in the separator. 3 copies of each entity are formed at the separator which means products A, B and C are formed at each separator (Figure 40).

Figure 30: Properties of the separator at Shunting yard 3 (decoupling)

Figure 31 shows the properties of the transfer node at the separator. The node has been set up to allow the movement of the RTCs and the train to the different loading zones in the plant. The nodes are setup to allow the movement of the locomotives and RTCs to the correct destination together with the RTCs.
Figure 31: Properties of a transfer node in Shunting yard 3

Figure 32 shows the properties of the transfer node at the separator. The node has been set up to allow different RTCs with different products to be transferred to different loading zones in the plant. The destination where the products should go is set up in the transfer node parameters.

Figure 32: Properties of the transfer node in Shunting yard 3: inbound

Figure 33 shows the properties of the transfer node at the product B and C loading zone. It is set in such a way that the locomotives go to a specific location. It also allows the coupling of RTCs to the locomotives to take place at the loading zone. The transfer node stated in figure 31 is set up as a location where the locomotive couples RTCs. This happens after the loading of products has taken place at the loading zone. The (True) parameter allows entities to ride on a locomotive. The
RTCs will then be taken to Shunting yard 3 where the full train will be put together and then taken away by the Company X locomotives to Shunting yard 1.

Figure 33: Properties of the transfer node at Product B and C loading zones

Figure 34 shows the properties of the transfer node at the combiner. The node enables the locomotives to go to the specific destination required. The destination is Product A loading zone. It is set up to allow coupling of the RTCs to the locomotives.

Figure 34 : Properties of the transfer node at Product A loading zone

Figure 35 shows the properties of the transfer node in the combiner. The node enables the locomotives to go to the specific destination required. The destination is in this case Product B and C loading zone. It is set up to allow coupling of the RTCs to the locomotives.
Figure 35: Properties of a server at Product B and C loading zones

Figure 36 shows the properties of the basic node where the Company A locomotives park when they are not in demand.

Figure 36: Properties of the basic node at the locomotive base

Figure 37 shows the properties of the server at the Product A loading zone. The processing time of the entity load has been set at 20 minutes. It takes 20 minutes to complete the loading at this point. The server has been enabled to allow Product A RTCs to enter the loading zone as expected.
Figure 37: Properties of a server at the Product A loading zone

Figure 3 and 39 shows the properties of the resource at the shunt line 2 in Shunting yard 3. The resource helps in allowing the movement of the entities or RTCs to take place in a particular sequence. A first in, first out principle takes place.

Figure 38: Properties of the resources at Shunting yard 3, Shunt line 2

6.3. Explanation of the model

6.3.1. Entities

a) Non Permanent entities:
   - Block Train RTCs
   - Product E trains
   - Transnet Locomotives

b) Permanent entities
• Company X locomotives
• Loading and offloading zones
• Locomotive shed

6.3.2. Model

The model developed simulates the Block trains’ activities. Two shunting locomotives are used in the model (figure 39).
Figure 39: Company X shunting Process model
The model was run for a six month period. Running the model for this time meant that the simulation output was consistent and accurate.

8. RESULTS

8.1. Locomotive Transit Times
Results are shown in table 2. For example, it was found that the RTCs on Block train take 1.27 hours minimum, 1.69 hours maximum and 1.48 hours on average to arrive at the Product A loading and offloading zone, and back to shunting yard 1. The Block train takes 1.3 Hours minimum, 1.59 Hours average 1.89 Hours maximum to travel from Shunting yard 1 to the Product B and C loading zone, and back and back to shunting yard 1.

The times indicated in table 2 include the shunting and waiting time for locomotives to pick up the wagons, as well as the time taken by the train to exit. The time includes the offloading and administration time.
<table>
<thead>
<tr>
<th>Data Source</th>
<th>Statistic Type</th>
<th>Value</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time from arrival to Product A plant K</td>
<td>Average</td>
<td>1.48h</td>
<td>Average Block train travel time from arrival to Product A plant</td>
</tr>
<tr>
<td>Time from arrival to Product A plant</td>
<td>Maximum</td>
<td>1.69h</td>
<td>Maximum time from arrival to Product A plant</td>
</tr>
<tr>
<td>Time from arrival to Product A plant</td>
<td>Minimum</td>
<td>1.27h</td>
<td>Minimum time from arrival to Product A plant</td>
</tr>
<tr>
<td>Time from arrival to Product A plant</td>
<td>Observations</td>
<td>60</td>
<td><strong>Total loads of sets of ten RTCs (60 x 10 RTCs)</strong></td>
</tr>
<tr>
<td>Time from arrival to Product B and C plants</td>
<td>Average</td>
<td>1.59h</td>
<td>Average time from arrival to Product B and C plants.</td>
</tr>
<tr>
<td>Time from arrival to Product B and C plants</td>
<td>Maximum</td>
<td>1.89h</td>
<td>Maximum time from arrival to Product B and C plants.</td>
</tr>
<tr>
<td>Time from arrival to Product B and C plants</td>
<td>Minimum</td>
<td>1.3h</td>
<td>Minimum time from arrival to Product B and C plants.</td>
</tr>
<tr>
<td>Time from arrival to Product B and C plants</td>
<td>Observations</td>
<td>132</td>
<td><strong>Total loads of sets of ten RTCs (132 x 10 RTCs)</strong></td>
</tr>
</tbody>
</table>

Table 2: The time RTCs

8.2. Total number of loads entering and exiting the plants in six months

Table 3 illustrates the number of full trains per product which were offloaded in each loading and offloading zone.
In table 4 the utilisation of the two locomotives is shown for the 24 weeks (6 months).
The total utilisation of locomotive 1 for Product A was 1.06 % of the total time when the
locomotive was available. The total utilisation of locomotive 2 for Products A, B, C and D was
2.66 % of the total time when the locomotive was available. The total utilisation of both
locomotives was 1.86% of the total time when both locomotives were available which added up
to (1.06+2.66)/2=1.86%. The total idle time of locomotive 1 was 98.94 % while the total idle
time of locomotive 2 was 97.34 % when they were available (See table 4).
<table>
<thead>
<tr>
<th>Object Type</th>
<th>Object Name</th>
<th>Data Item</th>
<th>Statistic Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MyVehicle</td>
<td>BMLoco</td>
<td>Scheduled Utilization</td>
<td>Percent</td>
<td>1.86%</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[1]</td>
<td>Scheduled Utilization</td>
<td>Percent</td>
<td>1.06%</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[1]</td>
<td>Idle Time</td>
<td>Percent</td>
<td>98.94%</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[1]</td>
<td>Transporting Time</td>
<td>Percent</td>
<td>1.06%       (42.78h)</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[2]</td>
<td>Scheduled Utilization</td>
<td>Percent</td>
<td>2.66%</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[2]</td>
<td>Idle Time</td>
<td>Percent</td>
<td>97.34%</td>
</tr>
<tr>
<td>MyVehicle</td>
<td>BMLoco[2]</td>
<td>Transporting Time</td>
<td>Percent</td>
<td>2.66%       (107.18h)</td>
</tr>
</tbody>
</table>

Table 4: Utilisation of locomotives

9. VALIDATION.

Limited historical data was available for validation. Therefore validation primarily consisted of examining each simulated process individually, and considering whether it was a true reflection of the real-world process, based on what has been observed on-site, as well as practical considerations.

The Block train arrives twice on a weekly basis in Town M and leaves twice from Company X plant situated in Town M to Town L, and the Block train in the model does the same. These results were captured over 24 weeks (6 months) and confirm that in that period 48 Block train trains entered Company X plant K, and 48 Block train trains exited the plant K (Figure 44). This confirms that the Block train arrives twice weekly in six months (24 weeks).

10 sets of RTCs can be loaded or offloaded in each plant at a time, given the limited capacity of the loading and off-loading zones which resulted in 60 loads of Product A sets of ten, and 132 sets of Products B and C transported. The model depicts the decoupling and coupling of ten sets of RTCs in Shunting yard 3. The speed of locomotives to different locations is set at 15km/h as is the case in the real life situation. The Product E trains have also been included in the model because they share the single line in the network rail with the Block Train when they go to a Product E offloading zone. The Product E trains and Block Train use single line at the rail
network and this line is controlled and only allows one train at a time to occupy the section, which is what occurs in the real life situation.

Figure 40: A 3D model of the rail network in the plant

In figure 41 Product E train enters the Product E loading zone. There is no other train on the track except the coal train. There is no exchange of RTCs which confirms what happens in the real situation.
In figure 42 Product E train offloads Product E in the loading zone. The offloading takes place at this point. The loading of Product E in this case confirms what happens in the real situation.

In figure 43 Product E train has offloaded Product E at the loading zone. The offloading has taken place and the train is going back. The movement of the Product E train confirms what happens in the real situation.
In figure 44 Company X locomotives goes to Shunting yard 1 to pick up the Block train from the locomotive base. As in the real situation, the locomotive movement will take place as simulated.

In figure 45 shows a locomotive is on its way to the Shunting yard 1 to pick up the Block train from the locomotive base. The base is where the locomotives park when they are idling.
Figure 45: The locomotive passes shunting yard 2 to pick up the full train at shunting yard 1

Figure 46 shows a situation where a Product E locomotive has stopped because the Company X locomotive has already occupied the single track section in the rail network.

Figure 46: A locomotive has stopped due to single track being occupied

In figure 47 a locomotive collects the Block train at Shunting yard 1.
Figure 47: The locomotive collects the full train at Shunting yard 1

In figure 48 a locomotive passes the single track to Shunting yard 3.

Figure 48: The locomotive has moved the train from Shunting yard 1 to Shunting yard 3

In figure 49 Company X locomotives pass the single track with the load to the Shunting yard 3. This shows what happens in the real situation.
Figure 49: The locomotive passes the single track with the full load to Shunting yard 3

In figure 50 RTCs are taken to the loading zones in sets of tens after the train has been divided in Shunting yard 3. They will decouple and then move to the front to take the load to the loading zone.

Figure 50: Decoupling of RTCs is takes place at Shunting yard 3

In figure 51 a locomotive takes a set of ten RTCs to the loading zone.
In figure 51 a locomotive goes to the loading zone with RTCs from Shunting yard 3.

Figure 52: The locomotive is taking RTCs to the loading zone.

In figure 52 a locomotive has dropped off the load in the Product B and C loading zones. This confirms what happens in the real situation where the locomotive will take the loads to the loading zones as expected.

Figure 53: The locomotive drops off the RTCs at the loading zone.
In figure 54 Company X the locomotive returns to Shunting yard 3 to collect more loads to be dropped off at the Product B and C loading zones. This shows what happens in the real situation where the locomotive after dropping off the load will go back to Shunting yard 3 to get more loads. At same time the dropping off the loaded loads to Shunting yard 3 takes place confirming the real situation.

![Figure 54: The locomotive collect more sets of RTCs at shunting yard 3 to the loading zone in the plant](image)

In figure 55 a locomotive from the loading zone to Shunting yard 3 to collect more loads and at the same time drop off the loaded RTCs at Shunting yard 3. This reflects what happens in the real situation where the locomotive will take loads to Shunting yard 3 after loading and transport more RTCs to the loading zone as expected.
Figure 55: The locomotive collects more sets of RTCs at Shunting yard 3 to drops them off at the loading zone

In figure 56 a locomotive drop off another load of Product B and C loading zones. This mirrors what happens in the real situation.

Figure 56: The locomotive drops off the RTCs at the loading zone

In figure 57 a locomotive transport the full train after loading has taken place to Shunting yard 1.
Figure 57: The full train is taken back to Shunting yard 1 after loading coupling has taken place

In figure 58 a locomotive passes the single track with the full load to Shunting yard 1.

Figure 58: The locomotive with full train goes to shunting yard 1 passes a single track section

In figure 59 a locomotive drop off the full train at Shunting yard 1.
Figure 59: The locomotive has dropped off the full train at Shunting yard 1
In figure 60 shows a locomotives dropping off a set of ten RTCs at the Product A loading zone.

Figure 60: The locomotive makes a drop off at Product A loading zone
In figure 61 a locomotive goes to Shunting yard 3 to collect more sets of RTCs. At the same time a locomotive takes the loaded RTCS to Shunting yard 3 to be jointed to the other loaded RTCs.
10. DISCUSSION

The objective of the study was to discover whether a simulation model of the Company X, Plant K rail network system can be developed and used to compare system design changes. The model has been developed successfully. It has been shown to accurately simulate activities occurring on-site.

Taking the results into account, Company X could increase productivity by increasing the speed at which the locomotives travel from the Shunting yard 1 to the offloading and loading zones in the primary area. Instead of travelling at 15km/h, the minimum speed could be increased to 20km/h. The train would then reach its destination more quickly than it does currently. One locomotive could also be used instead of using two to transport the Block Train RTCs, provided that the current schedule does not alter the arrival and departure of the Block Train twice weekly.

Improvement of delivery will also depend on increasing the speed at which the Company X locomotives travel from Shunting yard 1 to the loading and offloading zones. The decrease in the time it takes for the Car-puller to take a set of RTC to the loading zones could also contribute to improving service delivery within the plant.

Adding more days in Block Train schedule will increase productivity. Instead of the train coming only twice a week this could be increased to 3 days. It will be a challenge to change processing time as under normal circumstances it takes 20 minutes to load and offload the products at the
loading zones. Therefore the waiting time of locomotives would not be improved because of the processing time.

The total time of the locomotives’ availability could be used for other work, or the locomotives could be serviced in terms of maintenance and repairs. Given the capacity of the locomotives available, the shunting yard could handle more trains. The speed at which the trains travel could be increased in order to improve the service delivery. As a result of the availability of locomotives, the number of loads can be increased per months.

11. CONCLUSION

The focus of the study was to simulate the Company X plant K rail networks and how shunting takes place from Shunting yard 1 to Shunting yard 3, to the offloading and loading zones and back. In conclusion the Company X rail network can be modelled successfully, and this model can be used for evaluating proposed improvements to the rail network.

12. RECOMMENDATIONS

Delivery of quality service requires company X to be efficient in all sectors of its operations. In order to achieve the quality service delivery and economies of scale, the focus should be on cost containment and on increasing production. Cost containment can be achieved by efficient utilisation of the locomotives. Increasing the speed of the locomotive to 20 km/h will increase the travel time of the locomotives from shunting yard 1 to loading zone and back to the shunting yard 1. Decreasing the time it takes for the Car-puller to take a set of RTC to the loading zones could also assist to improving service delivery within the plant.

Increase in customers could also lead to increase in production. The more a product is delivered on time in different loading zones this will lead to increase in production and more volumes can be produced and this could lead to increase in revenue and more customers should be acquired. By adding more days in the Block Train schedule can also assist in increasing the number of loads transported to Plant K in a month. Efficient controlling of signals will lead to safe environment, free of accident and incidents in the rail network. It is therefore imperative that the controllers should be well trained in order to perform their duties properly by managing control rail signals systems efficiently.
Based on the results it is evident that the locomotives are being under-used. The locomotives’ idling time is high. This could be reduced if more trains were added to the schedule. The company could then consider using these locomotives for other purposes, transporting other RTCs with other products within the primary area. One locomotive can be used to transport Block Train.

The Block train can be allowed to come 3 days per week to plant K instead of only two days per week. Processing time in the loading and offloading of the products at the loading zones can be decreased by uplifting employee morale and close supervision at this point. Therefore decrease of processing time could assist in to reduce the time the locomotives spend at the locomotive base. The locomotives that are available can be used for other work in plant K. Once the speed is improved and the number of Block Trains coming to plant K in a week is increased the output in production can also be increased per month. Shunting yards can be used to handle more trains as there will be more space available.
13. REFERENCES


Published by: INFORMS 31(3) pp. 66-75.


14. APPENDICES

14.2. Appendix A: Model processes
SetPlantDestination

Begin

Is.Entity1

Decide

SetNode2

Set Node

True

False

End

End
### 14.2. Appendix B: Standard Library Documentation

**Heading: Town M Rail Simulation Model.**

**Standard Library**
Model10  
**Author:** Thabiso

**Property Overrides** [Show / Hide 14 items]

<table>
<thead>
<tr>
<th>Override</th>
<th>Visible</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(ResourceIdleCostRate)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(ResourceCostPerUse)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(ResourceUsageCostRate)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(LogResourceUsage)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(DisplayCategory)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(DisplayColor)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(CapacityType)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(WorkSchedule)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(WorkDayExceptions)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(WorkPeriodExceptions)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(InitialCapacity)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(RankingRule)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(RankingExpression)</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>(DynamicSelectionRule)</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>

**States** [Show / Hide 7 items]

<table>
<thead>
<tr>
<th>State</th>
<th>Dimensions</th>
<th>InitialValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntegerState (ShuntLines)</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>RealState (DecoupleTime)</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>RealState (CoupleTime)</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
**IntegerState (LoadsWaitToEnter)**

**Description:** R1=ProductA R2=ProductB R3=ProductC R4 =Total

**Dimensions:** 4

**IntegerState (LoadsEnterCount)**

**Description:** R1=PoductA R2=ProductB R3=ProductC R4=Total

**Dimensions:** 4

**Events**

**Show / Hide 2 items**

Event (Decoupled1)
Event (Decoupled2)

**Elements**

**Show / Hide 14 items**

ShuntLine1Waiting (Storage)

**RankingRule:** SmallestValueFirst

**RankingExpression:** Entity1.LoadNumber

ShuntLine2Waiting (Storage)

**RankingRule:** SmallestValueFirst

**RankingExpression:** Entity1.LoadNumber

Shuntline1Status (Monitor)

**StateVariableName:** ShuntLines

Shuntline2Status (Monitor)

**StateVariableName:** ShuntLines

**LoadsEnteredTotal (OutputStatistic)**

**Expression:** LoadsEnterCount[4]

**LoadsEnteredProductA (OutputStatistic)**

**Expression:** LoadsEnterCount[1]

**LoadsEnteredProductB (OutputStatistic)**

**Expression:** LoadsEnterCount[2]

**LoadsEnteredProductC (OutputStatistic)**

**Expression:** LoadsEnterCount[3]

**LoadsExitTotal (OutputStatistic)**

**Expression:** LoadsExitCount[4]

**LoadsExitProductA (OutputStatistic)**

**Expression:** LoadsExitCount[1]

**LoadsExitProductB (OutputStatistic)**

**Expression:** LoadsExitCount[2]

**LoadsExitProductC (OutputStatistic)**

**Expression:** LoadsExitCount[3]

**TimeFromArrivalToProductAPlant (TallyStatistic)**

**UnitType:** Time

**TimeFromArrivalToProductA_ProductBPlant (TallyStatistic)**

**UnitType:** Time

**Processes**

**Show / Hide 10 items**

EnterSingleLine

**TokenActionOnAssociatedObjectDestroyed:** EndProcess

**Steps**
<table>
<thead>
<tr>
<th>Decide1 (Decide)</th>
<th>DecideType: ConditionBased</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression: Is.Vehicle</td>
<td></td>
</tr>
<tr>
<td>Seize1 (Seize)</td>
<td></td>
</tr>
<tr>
<td>ResourceSeizes:</td>
<td></td>
</tr>
<tr>
<td>ObjectName: LineRestriction</td>
<td></td>
</tr>
<tr>
<td>ExitSingleLine</td>
<td></td>
</tr>
<tr>
<td><strong>TokenActionOnAssociatedObjectDestroyed:</strong></td>
<td>EndProcess</td>
</tr>
</tbody>
</table>

**Steps**

<table>
<thead>
<tr>
<th>Decide1 (Decide)</th>
<th>DecideType: ConditionBased</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression: Is.Vehicle</td>
<td></td>
</tr>
<tr>
<td>Release1 (Release)</td>
<td></td>
</tr>
<tr>
<td>ResourceReleases:</td>
<td></td>
</tr>
<tr>
<td>ObjectName: LineRestriction</td>
<td></td>
</tr>
<tr>
<td>ChooseShuntLine</td>
<td></td>
</tr>
<tr>
<td>LoadsWaitToEnter-1 (Assign)</td>
<td></td>
</tr>
<tr>
<td>----------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>StateVariableName:</strong> LoadsWaitToEnter</td>
<td></td>
</tr>
<tr>
<td><strong>NewValue:</strong> LoadsWaitToEnter-1</td>
<td></td>
</tr>
<tr>
<td>SetPlantDestination</td>
<td></td>
</tr>
</tbody>
</table>

**Steps**

<table>
<thead>
<tr>
<th>SetNode2 (SetNode)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NodeName:</strong> GeneralInfo[Entity1.Destination].RTCDestinations</td>
</tr>
<tr>
<td><strong>Is.Entity1</strong> (Decide)</td>
</tr>
<tr>
<td><strong>DecideType:</strong> ConditionBased</td>
</tr>
<tr>
<td><strong>Expression:</strong> Is.Entity1</td>
</tr>
<tr>
<td>CheckShuntline1_Availability</td>
</tr>
</tbody>
</table>

**Steps**

<table>
<thead>
<tr>
<th>Decide5 (Decide)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DecideType:</strong> ConditionBased</td>
</tr>
<tr>
<td><strong>Expression:</strong> Is.Vehicle</td>
</tr>
<tr>
<td>Is Vehicle Loaded? (Decide)</td>
</tr>
<tr>
<td><strong>DecideType:</strong> ConditionBased</td>
</tr>
<tr>
<td><strong>Expression:</strong> MyVehicle.RideStation.Contents&gt;0</td>
</tr>
<tr>
<td>Decide8 (Decide)</td>
</tr>
<tr>
<td><strong>DecideType:</strong> ConditionBased</td>
</tr>
<tr>
<td><strong>Expression:</strong> (ShuntLine1_InboundShuntingYard3.ParentOutputBuffer.Contents == 0) &amp;&amp; (ShuntLine1_InboundShuntingYard3.MemberOutputBuffer.Contents == 0)</td>
</tr>
<tr>
<td>ResetShuntLine1 (Assign)</td>
</tr>
<tr>
<td><strong>StateVariableName:</strong> ShuntLines</td>
</tr>
<tr>
<td><strong>NewValue:</strong> 0</td>
</tr>
<tr>
<td>CheckShuntLineAvailability</td>
</tr>
</tbody>
</table>

**Steps**

<table>
<thead>
<tr>
<th>ShuntLine1 open? (Decide)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DecideType:</strong> ConditionBased</td>
</tr>
<tr>
<td><strong>Expression:</strong> ShuntLines[1]==0</td>
</tr>
<tr>
<td>AllocateShuntLine1 (Assign)</td>
</tr>
<tr>
<td><strong>StateVariableName:</strong> ShuntLines</td>
</tr>
<tr>
<td><strong>NewValue:</strong> 1</td>
</tr>
<tr>
<td>Assignments:</td>
</tr>
<tr>
<td><strong>AssignmentsStateVariableName:</strong> Entity1.WhichShuntLine</td>
</tr>
<tr>
<td><strong>AssignmentsNewValue:</strong> 1</td>
</tr>
<tr>
<td>ShuntLine1_Inbound (SetNode)</td>
</tr>
<tr>
<td><strong>NodeName:</strong> Input@ShuntLine1_InboundShuntingYard3</td>
</tr>
<tr>
<td>AllocateShuntLine1 (Assign)</td>
</tr>
<tr>
<td><strong>NewValue:</strong></td>
</tr>
<tr>
<td>Assignments:</td>
</tr>
<tr>
<td><strong>AssignmentsStateVariableName:</strong> Entity1.WhichShuntLine</td>
</tr>
<tr>
<td><strong>AssignmentsNewValue:</strong> 1</td>
</tr>
</tbody>
</table>
Is.Entity1? (Decide)
DecideType: ConditionBased
Expression: Is.Entity1

ProductA? (Decide)
DecideType: ConditionBased
Expression: Entity1.Destination==1

ShuntLine2 open? (Decide)
DecideType: ConditionBased
Expression: ShuntLines[2]==0

AllocateShuntLine2 (Assign)
StateVariableName: ShuntLines
NewValue: 1
Assignments:
AssignmentsStateVariableName: Entity1.WhichShuntLine
AssignmentsNewValue: 2

ShuntLine2_Inbound (SetNode)
NodeName: Input@ShuntLine2_Inbound

AllocateShuntLine1 (Assign)
NewValue:
Assignments:
AssignmentsStateVariableName: Entity1.WhichShuntLine
AssignmentsNewValue: 2

Shuntline1StatusChange (Wait)
EventName: Shuntline1Status.Event
Delay1 (Delay)
DelayTime: Math.EPSILON

Shuntline2StatusChange (Wait)
EventName: Shuntline2Status.Event
Delay1 (Delay)
DelayTime: Math.EPSILON

LoadsWaitToEnter+1 (Assign)
StateVariableName: LoadsWaitToEnter
NewValue: LoadsWaitToEnter+1
Assignments:
AssignmentsStateVariableName: Entity1.ArrivalTime
AssignmentsNewValue: Timenow

ResetShuntlineStatus

Steps
Decide5 (Decide)
DecideType: ConditionBased
Expression: Is.Vehicle

Shuntline1Empty? (Decide)
DecideType: ConditionBased
Expression: (ShuntLine1_InboundShuntingYard3.ParentOutputBuffer.Contents == 0) &&
(ShuntLine1_InboundShuntingYard3.MemberOutputBuffer.Contents == 0)

FromShuntline1? (Decide)
**DecideType:** ConditionBased
**Expression:** Vehicle.PreviousNode.Name=="ParentOutput@ShuntLine1_InboundShuntingYard3"

Shuntline2Empty? (Decide)
**DecideType:** ConditionBased
**Expression:** (ShuntLine2_Inbound.ParentOutputBuffer.Contents==0)&&(ShuntLine2_Inbound.MemberOutputBuffer.Contents==0)

AllocateShuntLine1 (Assign)
**StateVariableName:** ShuntLines
**NewValue:** 0

AllocateShuntLine2 (Assign)
**StateVariableName:** ShuntLines
**NewValue:** 0

CountExitLoads

Steps

Assign3 (Assign)
**StateVariableName:** LoadsExitCount
**NewValue:** LoadsExitCount[MyVehicle.RideStation.Contents.FirstItem.Entity1.ProductType]+1

Assignments:
**AssignmentsStateVariableName:** LoadsExitCount
**AssignmentsNewValue:** LoadsExitCount[4]+1

ArrivaAtProductAPlant

Steps

Is.Entity1? (Decide)
**DecideType:** ConditionBased
**Expression:** Is.Entity1

TimeFromArrivalToProductAPlant (Tally)
**TallyStatisticName:** TimeFromArrivalToProductAPlant
**Value:** TimeNow-Entity1.ArrivalTime

ArrivaAtProductB(ProductCPlant)

Steps

Is.Entity1? (Decide)
**DecideType:** ConditionBased
**Expression:** Is.Entity1

TimeFromArrivalToProductB/ProductCPlant (Tally)
**TallyStatisticName:** TimeFromArrivalToProductB/ProductCPlant
**Value:** TimeNow-Entity1.ArrivalTime

Objects  Show / Hide 295 items
ArrivingProductATrains (Source)

**EntityType:** BlockTrainProductA  
**InterarrivalTime:** Random.Exponential(7)  
**ArrivalNoShowProbability:** 0  
**MaximumArrivals:** 0  

**AssignmentsBeforeExiting:**

- **AssignmentsBeforeExitingStateVariableName:** Entity1.Destination  
- **AssignmentsBeforeExitingNewValue:** 1  
- **AssignmentsBeforeExitingStateVariableName:** Entity1.NumberWagons  
- **AssignmentsBeforeExitingNewValue:** 40  
- **AssignmentsBeforeExitingStateVariableName:** LoadNumberOverall  
- **AssignmentsBeforeExitingNewValue:** LoadNumberOverall+1  
- **AssignmentsBeforeExitingStateVariableName:** Entity1.LoadNumber  
- **AssignmentsBeforeExitingNewValue:** LoadNumberOverall

**Associated Nodes**

- **Output@ArrivingProductATrains (TransferNode)**
  - **FlowSplitAllocationRule:** Proportional Based On Link Weights  
  - **DestinationType:** Specific  
  - **DestinationNodeName:** Shunting1_Inbound  
  - **DestinationSelectionExpression:** Candidate.Node.InputLocation.Overload  
  - **TransporterSelectionExpression:** Candidate.Transporter.RideCapacityRemaining

  **Outbound Links:**
  - Path1 going to BasicNode1

ExitShuntingYard1 (Sink)

**Associated Nodes**

- **Input@ExitShuntingYard1 (BasicNode)**
  - **FlowSplitAllocationRule:** Proportional Based On Link Weights
  - **Outbound Links:**
    - Path170 going to Shunting1_Inbound

Inbound Links:

- Path2 coming from Shunting1_Inbound

Source2 (Source)

**EntityType:** ProductE  
**ArrivalNoShowProbability:** 0  
**MaximumArrivals:** 1

**Associated Nodes**
Output@Source2 (TransferNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
**DestinationType:** Specific
**DestinationNodeName:** Input@Server1
**DestinationSelectionExpression:**
Candidate.Node.InputLocation.Overload
**RideOnTransporter:** True
**TransporterName:** MineTrain
**TransporterSelectionExpression:**
Candidate.Transporter.RideCapacityRemaining
Outbound Links:
- Path30 going to BasicNode21

Inbound Links:
- Path155 coming from Input@Sink2

Sink2 (Sink)
**Associated Nodes**
Input@Sink2 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
- Path155 going to Output@Source2

Inbound Links:
- Path31 coming from BasicNode21

Server2 (Server)
**Associated Nodes**
Input@Server2 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Inbound Links:
- Path43 coming from BasicNode29

Output@Server2 (TransferNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
**DestinationNodeName:**
**DestinationSelectionExpression:**
Candidate.Node.InputLocation.Overload
**TransporterSelectionExpression:**
Candidate.Transporter.RideCapacityRemaining
Outbound Links:
**Server3 (Server)**

### Associated Nodes

**Input@Server3 (BasicNode)**

*FlowSplitAllocationRule*: Proportional Based On Link Weights

<table>
<thead>
<tr>
<th>Inbound Links:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Path44 going to BasicNode29</td>
</tr>
</tbody>
</table>

**Output@Server3 (TransferNode)**

*FlowSplitAllocationRule*: Proportional Based On Link Weights

<table>
<thead>
<tr>
<th>DestinationNodeName:</th>
<th>DestinationSelectionExpression:</th>
<th>TransporterSelectionExpression:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Candidate.Node.InputLocation.Overload</td>
<td>Candidate.Transporter.RideCapacityRemaining</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outbound Links:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Path46 coming from BasicNode28</td>
</tr>
</tbody>
</table>

**Server4 (Server)**

### Associated Nodes

**Input@Server4 (BasicNode)**

*FlowSplitAllocationRule*: Proportional Based On Link Weights

<table>
<thead>
<tr>
<th>Inbound Links:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Path55 coming from BasicNode34</td>
</tr>
</tbody>
</table>

**Output@Server4 (TransferNode)**

*FlowSplitAllocationRule*: Proportional Based On Link Weights

<table>
<thead>
<tr>
<th>DestinationNodeName:</th>
<th>DestinationSelectionExpression:</th>
<th>TransporterSelectionExpression:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Candidate.Node.InputLocation.Overload</td>
<td>Candidate.Transporter.RideCapacityRemaining</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outbound Links:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Path56 going to BasicNode34</td>
</tr>
</tbody>
</table>

**Server5 (Server)**

### Associated Nodes

**Input@Server5 (BasicNode)**

*FlowSplitAllocationRule*: Proportional Based On Link Weights

| Inbound Links: |
- **Path60** coming from **BasicNode37**

**Output@Server5 (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights  
**DestinationNodeName:**  
**DestinationSelectionExpression:** Candidate.Node.InputLocation.Overload  
**TransporterSelectionExpression:** Candidate.Transporter.RideCapacityRemaining  
Outbound Links:

- **Path61** going to **BasicNode37**

**Server6 (Server)**

**Associated Nodes**

**Input@Server6 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights  
Inbound Links:

- **Path107** coming from **BasicNode67**

**Output@Server6 (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights  
**DestinationNodeName:**  
**DestinationSelectionExpression:** Candidate.Node.InputLocation.Overload  
**TransporterSelectionExpression:** Candidate.Transporter.RideCapacityRemaining  
Inbound Links:

- **Path108** coming from **BasicNode67**

**Server7 (Server)**

**Associated Nodes**

**Input@Server7 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights  
Inbound Links:

- **Path112** coming from **BasicNode70**

**Output@Server7 (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights  
**DestinationNodeName:**  
**DestinationSelectionExpression:** Candidate.Node.InputLocation.Overload  
**TransporterSelectionExpression:** Candidate.Transporter.RideCapacityRemaining
Outbound Links:

- **Path113** going to **BasicNode70**

Server8 (Server)

Associated Nodes

Input@Server8 (BasicNode)

FlowSplitAllocationRule: Proportional Based On Link Weights

Inbound Links:

- **Path117** coming from **BasicNode73**

Output@Server8 (TransferNode)

FlowSplitAllocationRule: Proportional Based On Link Weights

DestinationNodeName:

DestinationSelectionExpression:
Candidate.Node.InputLocation.Overload

TransporterSelectionExpression:
Candidate.Transporter.RideCapacityRemaining

Outbound Links:

- **Path118** going to **BasicNode73**

Server9 (Server)

Associated Nodes

Input@Server9 (BasicNode)

FlowSplitAllocationRule: Proportional Based On Link Weights

Inbound Links:

- **Path120** coming from **BasicNode74**

Output@Server9 (TransferNode)

FlowSplitAllocationRule: Proportional Based On Link Weights

DestinationNodeName:

DestinationSelectionExpression:
Candidate.Node.InputLocation.Overload

TransporterSelectionExpression:
Candidate.Transporter.RideCapacityRemaining

Outbound Links:

- **Path121** going to **BasicNode74**

ProductB_ProductCLoading (Server)

ProcessingTime: Entity1.LoadTime

EnteredAddOnProcess: ArrivaAtProductB_ProductCPlant

Associated Nodes
**Input@ProductB_ProductCLoading (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path126** going to **Output@ProductB_ProductCLoading**

**Inbound Links:**

- **Path125** coming from **BasicNode78**

**Output@ProductB_ProductCLoading (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**DestinationType:** Specific

**DestinationNodeName:**
CouplingDestinations[Entity1.ParentOrMember].Shuntline2Outbound

**DestinationSelectionExpression:**
Candidate.Node.InputLocation.Overload

**RideOnTransporter:** True

**TransporterName:** BMLoco[Entity1.WhichLoco]

**TransporterSelectionExpression:**
Candidate.Transporter.RideCapacityRemaining

**Outbound Links:**

- **Path142** going to **BasicNode78**

**Inbound Links:**

- **Path126** coming from **Input@ProductB_ProductCLoading**

**Server11 (Server)**

**Associated Nodes**

**Input@Server11 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Inbound Links:**

- **Path129** coming from **BasicNode79**

**Output@Server11 (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**DestinationNodeName:**

**DestinationSelectionExpression:**
Candidate.Node.InputLocation.Overload

**TransporterSelectionExpression:**
Candidate.Transporter.RideCapacityRemaining

**Outbound Links:**

- **Path130** going to **BasicNode79**
Server12 (Server)

**Associated Nodes**

**Input@Server12 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Inbound Links:**

- **Path134** coming from **BasicNode81**

**Output@Server12 (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**DestinationNodeName:**

**DestinationSelectionExpression:**

Candidate.Node.InputLocation.Overload

**TransporterSelectionExpression:**

Candidate.Transporter.RideCapacityRemaining

**Outbound Links:**

- **Path135** going to **BasicNode81**

ProductALoading (Server)

**ProcessingTime:** Entity1.LoadTime

**EnteredAddOnProcess:** ArrivaAtProductAPlant

**Associated Nodes**

**Input@ProductALoading (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path141** going to **Output@ProductALoading**

**Inbound Links:**

- **Path137** coming from **BasicNode82**

**Output@ProductALoading (TransferNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**DestinationType:** Specific

**DestinationNodeName:**

CouplingDestinations[Entity1.ParentOrMember].Shuntline1Outbound

**DestinationSelectionExpression:**

Candidate.Node.InputLocation.Overload

**RideOnTransporter:** True

**TransporterName:** BMLoco[Entity1.WhichLoco]

**TransporterSelectionExpression:**

Candidate.Transporter.RideCapacityRemaining

**Outbound Links:**

- **Path138** going to **BasicNode82**
Inbound Links:

- **Path141** coming from **Input@ProductALoading**

BlockTrainProductA (Entity1)
BlockTrainProductB (Entity1)
BlockTrainProductC (Entity1)
ArrivingProductBTrains (Source)

**EntityType**: BlockTrainProductB  
**TimeOffset**: 1  
**InterarrivalTime**: Random.Exponential(7)  
**ArrivalNoShowProbability**: 0  
**MaximumArrivals**: 0  
**AssignmentsBeforeExiting**:  
**AssignmentsBeforeExitingStateVariableName**: Entity1.Destination  
**AssignmentsBeforeExitingNewValue**: 2  
**AssignmentsBeforeExitingStateVariableName**: Entity1.NumberWagons  
**AssignmentsBeforeExitingNewValue**: 40  
**AssignmentsBeforeExitingStateVariableName**: LoadNumberOverall  
**AssignmentsBeforeExitingNewValue**: LoadNumberOverall+1  
**AssignmentsBeforeExitingStateVariableName**: Entity1.LoadNumber  
**AssignmentsBeforeExitingNewValue**: LoadNumberOverall

Associated Nodes

**Output@ArrivingProductBTrains (TransferNode)**  
**FlowSplitAllocationRule**: Proportional Based On Link Weights  
**DestinationType**: Specific  
**DestinationNodeName**: Shunting1_Inbound  
**DestinationSelectionExpression**: Candidate.Node.InputLocation.Overload  
**TransporterSelectionExpression**: Candidate.Transporter.RideCapacityRemaining

Outbound Links:

- **Path139** going to **BasicNode1**

ArrivingProductCTrains (Source)

**EntityType**: BlockTrainProductC  
**TimeOffset**: 2  
**InterarrivalTime**: Random.Exponential(7)  
**ArrivalNoShowProbability**: 0  
**MaximumArrivals**: 0  
**AssignmentsBeforeExiting**:  
**AssignmentsBeforeExitingStateVariableName**: Entity1.Destination  
**AssignmentsBeforeExitingNewValue**: 2
AssignmentsBeforeExitingStateVariableName: Entity1.NumberWagons
AssignmentsBeforeExitingNewValue: 40
AssignmentsBeforeExitingStateVariableName: LoadNumberOverall
AssignmentsBeforeExitingNewValue: LoadNumberOverall+1
AssignmentsBeforeExitingStateVariableName: Entity1.LoadNumber
AssignmentsBeforeExitingNewValue: LoadNumberOverall

Associated Nodes
Output@ArrivingProductCTrains (TransferNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
DestinationType: Specific
DestinationNodeName: Shunting1_Inbound
DestinationSelectionExpression: Candidate.Node.InputLocation.Overload
TransporterSelectionExpression: Candidate.Transporter.RideCapacityRemaining

Outbound Links:
- Path140 going to BasicNode1

LineRestriction (Resource)
Server1 (Server)

Associated Nodes
Input@Server1 (BasicNode)
FlowSplitAllocationRule: Proportional Based On Link Weights

Outbound Links:
- Path37 going to Output@Server1

Inbound Links:
- Path36 coming from BasicNode24

Output@Server1 (TransferNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
DestinationType: Specific
DestinationNodeName: Input@Sink2
DestinationSelectionExpression: Candidate.Node.InputLocation.Overload
RideOnTransporter: True
TransporterName: MineTrain
TransporterSelectionExpression: Candidate.Transporter.RideCapacityRemaining
ExitedAddOnProcess: EnterSingleLine

Outbound Links:
- **Path154** going to **BasicNode24**

Inbound Links:

- **Path37** coming from **Input@Server1**

MineTrain (Vehicle)

- **InitialNode:** Output@Source2
- **IdleAction:** Park At Home

ProductE (Entity1)

- **InitialDesiredSpeed:** 15
- **InitialNumberOfInSystem:** 2

ShuntLine1 (Resource)

- **InitialNode:** LocoBase
- **IdleAction:** Park At Home

ShuntLine2ShuntingYard3 (Resource)

- **BatchQuantity:** 3
- **MatchingRule:** Match Members And Parent
- **MemberMatchExpression:** Entity1.LoadNumber
- **ParentMatchExpression:** Entity1.LoadNumber

BMLoco (MyVehicle)

- **InitialDesiredSpeed:** 15
- **InitialNumberOfInSystem:** 2

Associated Nodes

- **ParentInput@ShuntLine1_Outbound (BasicNode)**

Inbound Links:

- **Path156** coming from **BasicNode54**

MemberInput@ShuntLine1_Outbound (BasicNode)

Inbound Links:

- **Path158** coming from **BasicNode54**

Output@ShuntLine1_Outbound (TransferNode)

- **DestinationType:** Specific
- **DestinationNodeName:** Input@ExitShuntingYard1
- **RideOnTransporter:** True
- **TransporterName:** BMLoco[Entity1.WhichLoco]

Outbound Links:

- **Path159** going to **BasicNode46**

ShuntLine1_InboundShuntingYard3 (MySeparator)

- **SeparationMode:** Make Copies
- **CopyQuantity:** 3
- **ProcessingTime:** DecoupleTime

AssignmentsBeforeParentExiting:
AssignmentsBeforeParentExitingStateVariableName: Entity1.ParentOrMember
AssignmentsBeforeParentExitingNewValue: 1
AssignmentsBeforeParentExitingStateVariableName: Entity1.LoadTime
AssignmentsBeforeParentExitingNewValue: Random.Triangular(17,20,23)
AssignmentsBeforeMemberExiting:
AssignmentsBeforeMemberExitingStateVariableName: Entity1.ParentOrMember
AssignmentsBeforeMemberExitingNewValue: 2
AssignmentsBeforeMemberExitingStateVariableName: Entity1.LoadTime
AssignmentsBeforeMemberExitingNewValue: Random.Triangular(4,5,6)

Associated Nodes

Input@ShuntLine1_InboundShuntingYard3 (BasicNode)
Inbound Links:
- Path147 coming from BasicNode48

ParentOutput@ShuntLine1_InboundShuntingYard3 (TransferNode)
RideOnTransporter: True
TransporterName: BMLoco[Entity1.WhichLoco]
EnteredAddOnProcess: SetPlantDestination
ExitedAddOnProcess: ResetShuntlineStatus
Outbound Links:
- Path82 going to BasicNode54

Inbound Links:
- Connector1 coming from MemberOutput@ShuntLine1_InboundShuntingYard3

MemberOutput@ShuntLine1_InboundShuntingYard3 (TransferNode)
Outbound Links:
- Connector1 going to ParentOutput@ShuntLine1_InboundShuntingYard3

ShuntLine2_Inbound (MySeparator)
SeparationMode: Make Copies
CopyQuantity: 3
ProcessingTime: DecoupleTime
AssignmentsBeforeParentExiting:
**AssignmentsBeforeParentExitingStateVariableName:** Entity1.ParentOrMember  
**AssignmentsBeforeParentExitingNewValue:** 1  
**AssignmentsBeforeParentExitingStateVariableName:** Entity1.LoadTime  
**AssignmentsBeforeParentExitingNewValue:** Random.Triangular(13,15,17)  
**AssignmentsBeforeMemberExiting:**  
**AssignmentsBeforeMemberExitingStateVariableName:** Entity1.ParentOrMember  
**AssignmentsBeforeMemberExitingNewValue:** 2  
**AssignmentsBeforeMemberExitingStateVariableName:** Entity1.LoadTime  
**AssignmentsBeforeMemberExitingNewValue:** Random.Triangular(4,5,6)  

**Associated Nodes**  
**Input@ShuntLine2_Inbound (BasicNode)**  
Inbound Links:  
- **Path148** coming from **BasicNode60**  

**ParentOutput@ShuntLine2_Inbound (TransferNode)**  
**RideOnTransporter:** True  
**TransporterName:** BMLoco[Entity1.WhichLoco]  
**EnteredAddOnProcess:** SetPlantDestination  
**ExitedAddOnProcess:** ResetShuntlineStatus  
Outbound Links:  
- **Path162** going to **BasicNode54**  

**Inbound Links:**  
- **Connector2** coming from **MemberOutput@ShuntLine2_Inbound**  

**MemberOutput@ShuntLine2_Inbound (TransferNode)**  
Outbound Links:  
- **Connector2** going to **ParentOutput@ShuntLine2_Inbound**  

**ShuntLine2_OutboundShuntingYard3 (Combiner)**  
**BatchQuantity:** 3  
**MatchingRule:** Match Members And Parent  
**MemberMatchExpression:** Entity1.LoadNumber  
**ParentMatchExpression:** Entity1.LoadNumber  

**Associated Nodes**  
**ParentInput@ShuntLine2_OutboundShuntingYard3 (BasicNode)**  
Inbound Links:
<table>
<thead>
<tr>
<th><strong>Path150</strong> coming from <strong>BasicNode54</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MemberInput@ShuntLine2_OutboundShuntingYard3</strong> (BasicNode)</td>
</tr>
<tr>
<td><strong>Inbound Links:</strong></td>
</tr>
<tr>
<td>• <strong>Path146</strong> coming from <strong>BasicNode54</strong></td>
</tr>
<tr>
<td><strong>Output@ShuntLine2_OutboundShuntingYard3</strong> (TransferNode)</td>
</tr>
<tr>
<td><strong>DestinationType:</strong> Specific</td>
</tr>
<tr>
<td><strong>DestinationNodeName:</strong> Input@ExitShuntingYard1</td>
</tr>
<tr>
<td><strong>RideOnTransporter:</strong> True</td>
</tr>
<tr>
<td><strong>TransporterName:</strong> BMLoco[Entity1.WhichLoco]</td>
</tr>
<tr>
<td><strong>Outbound Links:</strong></td>
</tr>
<tr>
<td>• <strong>Path71</strong> going to <strong>BasicNode44</strong></td>
</tr>
<tr>
<td><strong>ArrivalsShuntingYard1</strong> (Source)</td>
</tr>
<tr>
<td><strong>EntityType:</strong> Products.ProductType</td>
</tr>
<tr>
<td><strong>ArrivalMode:</strong> Arrival Table</td>
</tr>
<tr>
<td><strong>ArrivalTimeProperty:</strong> ArrivalTable.ArrivalTimes</td>
</tr>
<tr>
<td><strong>EntitiesPerArrival:</strong> ArrivalTable.ArrivalQty</td>
</tr>
<tr>
<td><strong>RepeatArrivalPattern:</strong> True</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExiting:</strong></td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> Entity1.Destination</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> Products.LoadDestination</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> LoadNumberOverall</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> LoadNumberOverall + 1</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> Entity1.LoadNumber</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> LoadNumberOverall</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> Entity1.ProdType</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> Products.ProdType</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> LoadsEnterCount</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> LoadsEnterCount[Entity1.ProdType] + 1</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingStateVariableName:</strong> LoadsEnterCount</td>
</tr>
<tr>
<td><strong>AssignmentsBeforeExitingNewValue:</strong> LoadsEnterCount[4] + 1</td>
</tr>
<tr>
<td><strong>AssignmentBeforeCreatingEntitiesTableName:</strong> Products</td>
</tr>
<tr>
<td><strong>AssignmentBeforeCreatingEntitiesTableRow:</strong> Products.Probability.RandomRow</td>
</tr>
<tr>
<td><strong>Associated Nodes</strong></td>
</tr>
<tr>
<td><strong>Output@ArrivalsShuntingYard1</strong> (TransferNode)</td>
</tr>
<tr>
<td><strong>DestinationType:</strong> Specific</td>
</tr>
<tr>
<td><strong>DestinationNodeName:</strong> Shunting1_Inbound</td>
</tr>
<tr>
<td><strong>Outbound Links:</strong></td>
</tr>
<tr>
<td>• <strong>Path172</strong> going to <strong>Shunting1_Inbound</strong></td>
</tr>
</tbody>
</table>
BasicNode1 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path3** going to BasicNode3
- **Path143** going to Shunting1_Inbound

Inbound Links:
- **Path1** coming from Output@ArrivingProductATrains
- **Path139** coming from Output@ArrivingProductBTrains
- **Path140** coming from Output@ArrivingProductCTrains

BasicNode2 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

BasicNode3 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path4** going to BasicNode4
- **Path5** going to BasicNode5

Inbound Links:
- **Path3** coming from BasicNode1

BasicNode4 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path11** going to BasicNode9
- **Path27** going to BasicNode19

Inbound Links:
- **Path4** coming from BasicNode3

BasicNode5 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path6** going to BasicNode6
- **Path10** going to BasicNode8

Inbound Links:
- **Path5** coming from BasicNode3
**BasicNode6 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- Path7 going to BasicNode7
- Path8 going to BasicNode7

**Inbound Links:**

- Path6 coming from BasicNode5

**BasicNode7 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- Path9 going to BasicNode8

**Inbound Links:**

- Path7 coming from BasicNode6
- Path8 coming from BasicNode6

**BasicNode8 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- Path14 going to BasicNode12

**Inbound Links:**

- Path9 coming from BasicNode7
- Path10 coming from BasicNode5

**BasicNode9 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- Path12 going to BasicNode10
- Path17 going to BasicNode13

**Inbound Links:**

- Path11 coming from BasicNode4

**BasicNode10 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path13** going to **BasicNode11**
- **Path15** going to **BasicNode11**

### Inbound Links:
- **Path12** coming from **BasicNode9**

### BasicNode11 (BasicNode)

**FlowSplitAllocationRule**: Proportional Based On Link Weights

- **Path16** going to **BasicNode12**

### Inbound Links:
- **Path13** coming from **BasicNode10**
- **Path15** coming from **BasicNode10**

### BasicNode12 (BasicNode)

**FlowSplitAllocationRule**: Proportional Based On Link Weights

- **Path21** going to **BasicNode15**

### Inbound Links:
- **Path14** coming from **BasicNode8**
- **Path16** coming from **BasicNode11**

### BasicNode13 (BasicNode)

**FlowSplitAllocationRule**: Proportional Based On Link Weights

- **Path18** going to **BasicNode14**
- **Path19** going to **BasicNode14**

### Inbound Links:
- **Path17** coming from **BasicNode9**

### BasicNode14 (BasicNode)

**FlowSplitAllocationRule**: Proportional Based On Link Weights

- **Path20** going to **BasicNode15**

### Inbound Links:
- **Path18** coming from **BasicNode13**
- **Path19** coming from **BasicNode13**

**BasicNode15 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path26** going to **Shunting1_Outbound**

Inbound Links:

- **Path20** coming from **BasicNode14**
- **Path21** coming from **BasicNode12**

**BasicNode16 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path25** going to **Shunting1_Outbound**

Inbound Links:

- **Path24** coming from **BasicNode19**
- **Path28** coming from **BasicNode18**

**Shunting1_Outbound (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path32** going to **BasicNode22**

Inbound Links:

- **Path25** coming from **BasicNode16**
- **Path26** coming from **BasicNode15**
- **Path144** coming from **Shunting1_Inbound**
- **Path168** coming from **BasicNode22**

**BasicNode18 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path28** going to **BasicNode16**

Inbound Links:

- **Path23** coming from **BasicNode20**
- **Path29** coming from **BasicNode20**
<table>
<thead>
<tr>
<th>BasicNode19 (BasicNode)</th>
<th>FlowSplitAllocationRule: Proportional Based On Link Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path24 going to BasicNode16</td>
<td></td>
</tr>
<tr>
<td>- Path22 going to BasicNode20</td>
<td></td>
</tr>
<tr>
<td>Inbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path27 coming from BasicNode4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BasicNode20 (BasicNode)</th>
<th>FlowSplitAllocationRule: Proportional Based On Link Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path23 going to BasicNode18</td>
<td></td>
</tr>
<tr>
<td>- Path29 going to BasicNode18</td>
<td></td>
</tr>
<tr>
<td>Inbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path22 coming from BasicNode19</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BasicNode21 (BasicNode)</th>
<th>FlowSplitAllocationRule: Proportional Based On Link Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path33 going to BasicNode22</td>
<td></td>
</tr>
<tr>
<td>- Path31 going to Input@Sink2</td>
<td></td>
</tr>
<tr>
<td>Inbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path30 coming from Output@Source2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BasicNode22 (BasicNode)</th>
<th>FlowSplitAllocationRule: Proportional Based On Link Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path34 going to BasicNode23</td>
<td></td>
</tr>
<tr>
<td>- Path168 going to Shunting1_Outbound</td>
<td></td>
</tr>
<tr>
<td>Inbound Links:</td>
<td></td>
</tr>
<tr>
<td>- Path32 coming from Shunting1_Outbound</td>
<td></td>
</tr>
<tr>
<td>- Path33 coming from BasicNode21</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BasicNode23 (BasicNode)</th>
<th>FlowSplitAllocationRule: Proportional Based On Link Weights</th>
</tr>
</thead>
</table>
Inbound Links:

- Path35 going to BasicNode24
- Path38 going to BasicNode25

Inbound Links:

- Path34 coming from BasicNode22
- Path169 coming from BasicNode25

BasicNode24 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- Path36 going to Input@Server1

Inbound Links:

- Path35 coming from BasicNode23
- Path154 coming from Output@Server1

BasicNode25 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- Path39 going to BasicNode27
- Path169 going to BasicNode23

Inbound Links:

- Path38 coming from BasicNode23
- Path40 coming from ShuntingYard2

ShuntingYard2 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- Path40 going to BasicNode25
- Path48 going to BasicNode30

Inbound Links:

- Path41 coming from BasicNode27
- Path51 coming from BasicNode31

BasicNode27 (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- Path41 going to ShuntingYard2
- **Path171** going to **BasicNode30**

**Inbound Links:**
- **Path39** coming from **BasicNode25**
- **Path42** coming from **BasicNode29**
- **Path45** coming from **BasicNode28**

**BasicNode28 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path45** going to **BasicNode27**
- **Path46** going to **Input@Server3**

**Inbound Links:**
- **Path47** coming from **Output@Server3**

**BasicNode29 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path42** going to **BasicNode27**
- **Path43** going to **Input@Server2**

**Inbound Links:**
- **Path44** coming from **Output@Server2**

**BasicNode30 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path49** going to **BasicNode32**
- **Path50** going to **BasicNode31**

**Inbound Links:**
- **Path48** coming from **ShuntingYard2**
- **Path171** coming from **BasicNode27**

**BasicNode31 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path52** going to **BasicNode32**
- **Path63** going to **BasicNode39**
- **Path51** going to *ShuntingYard2*

**Inbound Links:**

- **Path50** coming from *BasicNode30*
- **Path152** coming from *BasicNode39*

---

**BasicNode32 (BasicNode)**

- **FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path53** going to *BasicNode33*

**Inbound Links:**

- **Path49** coming from *BasicNode30*
- **Path52** coming from *BasicNode31*

---

**BasicNode33 (BasicNode)**

- **FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path54** going to *BasicNode34*
- **Path57** going to *BasicNode35*

**Inbound Links:**

- **Path53** coming from *BasicNode32*

---

**BasicNode34 (BasicNode)**

- **FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path55** going to *Input@Server4*

**Inbound Links:**

- **Path54** coming from *BasicNode33*
- **Path56** coming from *Output@Server4*

---

**BasicNode35 (BasicNode)**

- **FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**

- **Path58** going to *BasicNode36*

**Inbound Links:**
**Path57** coming from **BasicNode33**

**BasicNode36 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path59** going to **BasicNode37**
- **Path62** going to **BasicNode38**

**Inbound Links:**
- **Path58** coming from **BasicNode35**

**BasicNode37 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path60** going to **Input@Server5**

**Inbound Links:**
- **Path59** coming from **BasicNode36**
- **Path61** coming from **Output@Server5**

**BasicNode38 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path92** going to **LocoBase**

**Inbound Links:**
- **Path62** coming from **BasicNode36**
- **Path81** coming from **BasicNode52**

**BasicNode39 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path64** going to **BasicNode42**
- **Path66** going to **BasicNode40**
- **Path152** going to **BasicNode31**

**Inbound Links:**
- **Path63** coming from **BasicNode31**
- **Path149** coming from **BasicNode59**
BasicNode40 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- Path65 going to BasicNode41
- Path67 going to BasicNode41

Inbound Links:
- Path66 coming from BasicNode39

BasicNode41 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- Path69 going to BasicNode43

Inbound Links:
- Path65 coming from BasicNode40
- Path67 coming from BasicNode40

BasicNode42 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- Path68 going to BasicNode43
- Path70 going to BasicNode44

Inbound Links:
- Path64 coming from BasicNode39

BasicNode43 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- Path73 going to BasicNode45

Inbound Links:
- Path68 coming from BasicNode42
- Path69 coming from BasicNode41

BasicNode44 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path145** going to **BasicNode59**
- **Path151** going to **BasicNode60**

**Inbound Links:**
- **Path70** coming from **BasicNode42**
- **Path71** coming from **Output@ShuntLine2_OutboundShuntingYard3**

**BasicNode45** (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path74** going to **BasicNode47**

**Inbound Links:**
- **Path73** coming from **BasicNode43**

**BasicNode46** (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path76** going to **BasicNode48**

**Inbound Links:**
- **Path159** coming from **Output@ShuntLine1_Outbound**
- **Path153** coming from **BasicNode17**
- **Path157** coming from **BasicNode60**

**BasicNode47** (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path83** going to **BasicNode53**

**Inbound Links:**
- **Path74** coming from **BasicNode45**

**BasicNode48** (BasicNode)

**FlowSplitAllocationRule:** Proportional Based On Link Weights

**Outbound Links:**
- **Path77** going to **BasicNode49**
- **Path147** going to **Input@ShuntLine1_InboundShuntingYard3**
Inbound Links:

- **Path76** coming from **BasicNode46**

**BasicNode49 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path78** going to **BasicNode50**
- **Path87** going to **BasicNode55**
- **Path165** going to **BasicNode84**

Inbound Links:

- **Path77** coming from **BasicNode48**
- **Path164** coming from **BasicNode54**

**BasicNode50 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path79** going to **BasicNode51**
- **Path89** going to **BasicNode57**

Inbound Links:

- **Path78** coming from **BasicNode49**

**BasicNode51 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path80** going to **BasicNode52**
- **Path90** going to **BasicNode58**

Inbound Links:

- **Path79** coming from **BasicNode50**

**BasicNode52 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path81** going to **BasicNode38**
- **Path91** going to **Loco2Base**

Inbound Links:
BasicNode53 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
- Path80 coming from BasicNode51

Inbound Links:
- Path84 going to BasicNode54

BasicNode54 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
- Path99 going to BasicNode62
- Path156 going to ParentInput@ShuntLine1_Outbound
- Path158 going to MemberInput@ShuntLine1_Outbound
- Path146 going to MemberInput@ShuntLine2_OutboundShuntingYard3
- Path150 going to ParentInput@ShuntLine2_OutboundShuntingYard3
- Path164 going to BasicNode49

Inbound Links:
- Path84 coming from BasicNode53
- Path86 coming from BasicNode55
- Path82 coming from ParentOutput@ShuntLine1_InboundShuntingYard3
- Path162 coming from ParentOutput@ShuntLine2_Inbound
- Path167 coming from BasicNode64

BasicNode55 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
- Path86 going to BasicNode54

Inbound Links:
- Path85 coming from BasicNode56
- Path87 coming from BasicNode49

BasicNode56 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
Path85 going to BasicNode55
Path97 going to BasicNode61

Inbound Links:
Path88 coming from BasicNode57

BasicNode57 (BasicNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
Outbound Links:
Path88 going to BasicNode56

Inbound Links:
Path89 coming from BasicNode50
Path95 coming from BasicNode58

BasicNode58 (BasicNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
Outbound Links:
Path95 going to BasicNode57

Inbound Links:
Path90 coming from BasicNode51
Path94 coming from Loco2Base

Loco2Base (BasicNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
Outbound Links:
Path94 going to BasicNode58
Path96 going to BasicNode61

Inbound Links:
Path91 coming from BasicNode52
Path93 coming from LocoBase

LocoBase (BasicNode)
FlowSplitAllocationRule: Proportional Based On Link Weights
Outbound Links:
Path93 going to Loco2Base
Path100 going to BasicNode63
Path72 going to BasicNode84
<table>
<thead>
<tr>
<th>Inbound Links:</th>
<th>Outbound Links:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- <strong>Path92</strong> coming from <strong>BasicNode38</strong></td>
<td>- <strong>Path98</strong> going to <strong>BasicNode62</strong></td>
</tr>
<tr>
<td><strong>BasicNode61</strong> (BasicNode)</td>
<td><strong>FlowSplitAllocationRule</strong>: Proportional Based On Link Weights</td>
</tr>
<tr>
<td>- <strong>Path96</strong> coming from <strong>Loco2Base</strong></td>
<td>- <strong>Path102</strong> going to <strong>BasicNode64</strong></td>
</tr>
<tr>
<td>- <strong>Path97</strong> coming from <strong>BasicNode56</strong></td>
<td>- <strong>Path109</strong> going to <strong>BasicNode68</strong></td>
</tr>
<tr>
<td><strong>BasicNode62</strong> (BasicNode)</td>
<td><strong>FlowSplitAllocationRule</strong>: Proportional Based On Link Weights</td>
</tr>
<tr>
<td>- <strong>Path100</strong> coming from <strong>LocoBase</strong></td>
<td>- <strong>Path101</strong> going to <strong>BasicNode64</strong></td>
</tr>
<tr>
<td>- <strong>Path98</strong> coming from <strong>BasicNode61</strong></td>
<td>- <strong>Path103</strong> going to <strong>BasicNode65</strong></td>
</tr>
<tr>
<td>- <strong>Path99</strong> coming from <strong>BasicNode54</strong></td>
<td><strong>BasicNode64</strong> (BasicNode)</td>
</tr>
<tr>
<td><strong>FlowSplitAllocationRule</strong>: Proportional Based On Link Weights</td>
<td><strong>Outbound Links</strong>:</td>
</tr>
<tr>
<td>- <strong>Path104</strong> going to <strong>BasicNode66</strong></td>
<td>- <strong>Path104</strong> going to <strong>BasicNode66</strong></td>
</tr>
<tr>
<td>- <strong>Path167</strong> going to <strong>BasicNode54</strong></td>
<td><strong>Inbound Links:</strong></td>
</tr>
<tr>
<td>- <strong>Path101</strong> coming from <strong>BasicNode63</strong></td>
<td><strong>BasicNode64</strong> (BasicNode)</td>
</tr>
</tbody>
</table>
BasicNode65 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path111** going to BasicNode69

Inbound Links:
- **Path103** coming from BasicNode63
- **Path105** coming from BasicNode66

BasicNode66 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path105** going to BasicNode65
- **Path106** going to BasicNode67

Inbound Links:
- **Path104** coming from BasicNode64

BasicNode67 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path107** going to Input@Server6
- **Path108** going to Output@Server6

Inbound Links:
- **Path106** coming from BasicNode66

BasicNode68 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:
- **Path110** going to BasicNode70

Inbound Links:
- **Path109** coming from BasicNode62

BasicNode70 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:

- **Path112** going to **Input@Server7**
- **Path114** going to **BasicNode71**

Inbound Links:

- **Path113** coming from **Output@Server7**
- **Path110** coming from **BasicNode68**

**BasicNode71 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path115** going to **BasicNode72**
- **Path116** going to **BasicNode73**

Inbound Links:

- **Path114** coming from **BasicNode70**

**BasicNode72 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path119** going to **BasicNode74**

Inbound Links:

- **Path115** coming from **BasicNode71**

**BasicNode73 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path117** going to **Input@Server8**

Inbound Links:

- **Path116** coming from **BasicNode71**
- **Path118** coming from **Output@Server8**

**BasicNode74 (BasicNode)**

**FlowSplitAllocationRule:** Proportional Based On Link Weights

Outbound Links:

- **Path120** going to **Input@Server9**
Inbound Links:

- **Path119** coming from **BasicNode72**
- **Path121** coming from **Output@Server9**

BasicNode69 (BasicNode)

*FlowSplitAllocationRule*: Proportional Based On Link Weights

Outbound Links:

- **Path122** going to **BasicNode75**
- **Path132** going to **BasicNode80**
- **Path166** going to **BasicNode64**

Inbound Links:

- **Path111** coming from **BasicNode65**

BasicNode75 (BasicNode)

*FlowSplitAllocationRule*: Proportional Based On Link Weights

Outbound Links:

- **Path123** going to **BasicNode76**

Inbound Links:

- **Path122** coming from **BasicNode69**

BasicNode76 (BasicNode)

*FlowSplitAllocationRule*: Proportional Based On Link Weights

Outbound Links:

- **Path124** going to **BasicNode78**
- **Path127** going to **BasicNode77**

Inbound Links:

- **Path123** coming from **BasicNode75**

BasicNode77 (BasicNode)

*FlowSplitAllocationRule*: Proportional Based On Link Weights

Outbound Links:

- **Path128** going to **BasicNode79**

Inbound Links:

- **Path127** coming from **BasicNode76**
- **Path131** coming from **BasicNode79**
BasicNode78 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:

- Path125 going to Input@ProductB_ProductCLoading

Inbound Links:

- Path124 coming from BasicNode76
- Path142 coming from Output@ProductB_ProductCLoading

BasicNode79 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:

- Path129 going to Input@Server11
- Path131 going to BasicNode77

Inbound Links:

- Path128 coming from BasicNode77
- Path130 coming from Output@Server11

BasicNode80 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:

- Path133 going to BasicNode81
- Path136 going to BasicNode82

Inbound Links:

- Path132 coming from BasicNode69

BasicNode81 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:

- Path134 going to Input@Server12

Inbound Links:

- Path133 coming from BasicNode80
- Path135 coming from Output@Server12

BasicNode82 (BasicNode)
**FlowSplitAllocationRule:** Proportional Based On Link Weights
Outbound Links:
• **Path137** going to **Input@ProductALoading**

**Inbound Links:**

• **Path136** coming from **BasicNode80**
• **Path138** coming from **Output@ProductALoading**

**Shunting1_Inbound (TransferNode)**

- **FlowSplitAllocationRule:** Proportional Based On Link Weights
- **DestinationNodeName:** GeneralInfo[Entity1.Destination].RTCDestinations
- **DestinationSelectionExpression:** Candidate.Node.InputLocation.Overload
- **RideOnTransporter:** True
- **TransporterName:** BMLoco[Entity1.WhichShuntLine]
- **TransporterSelectionExpression:** Candidate.Transporter.RideCapacityRemaining
- **EnteredAddOnProcess:** CheckShuntLineAvailability

**Outbound Links:**

• **Path144** going to **Shunting1_Outbound**
• **Path2** going to **Input@ExitShuntingYard1**

**Inbound Links:**

• **Path143** coming from **BasicNode1**
• **Path170** coming from **Input@ExitShuntingYard1**
• **Path172** coming from **Output@ArrivalsShuntingYard1**

**BasicNode17 (BasicNode)**

**Outbound Links:**

• **Path153** going to **BasicNode46**
• **Path160** going to **BasicNode83**

**Inbound Links:**

• **Path163** coming from **BasicNode84**

**BasicNode59 (BasicNode)**

**Outbound Links:**

• **Path149** going to **BasicNode39**

**Inbound Links:**

• **Path145** coming from **BasicNode44**
• **Path161** coming from **BasicNode83**
BasicNode60 (BasicNode)
Outbound Links:
- Path148 going to Input@ShuntLine2_Inbound
- Path157 going to BasicNode46

Inbound Links:
- Path75 coming from BasicNode83
- Path151 coming from BasicNode44

BasicNode83 (BasicNode)
Outbound Links:
- Path75 going to BasicNode60
- Path161 going to BasicNode59

Inbound Links:
- Path160 coming from BasicNode17

BasicNode84 (BasicNode)
Outbound Links:
- Path163 going to BasicNode17

Inbound Links:
- Path72 coming from LocoBase
- Path165 coming from BasicNode49

Path1 (Path)
From Output@ArrivingProductATrains to BasicNode1
AllowPassing: False
Path3 (Path)
From BasicNode1 to BasicNode3
Type: Bidirectional
Path4 (Path)
From BasicNode3 to BasicNode4
Path5 (Path)
From BasicNode3 to BasicNode5
Path6 (Path)
From BasicNode5 to BasicNode6
Path7 (Path)
From BasicNode6 to BasicNode7
Path8 (Path)
From BasicNode6 to BasicNode7
Path9 (Path)
From BasicNode7 to BasicNode8
<table>
<thead>
<tr>
<th>Path</th>
<th>Description</th>
<th>Type</th>
<th>DrawnToScale</th>
<th>LogicalLength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path35</td>
<td>From BasicNode22 to BasicNode23</td>
<td>Bidirectional</td>
<td>False</td>
<td>4.5</td>
</tr>
<tr>
<td>Path38</td>
<td>From BasicNode23 to BasicNode24</td>
<td>Bidirectional</td>
<td>False</td>
<td>500</td>
</tr>
<tr>
<td>Path39</td>
<td>From BasicNode23 to BasicNode27</td>
<td>Bidirectional</td>
<td>False</td>
<td>250</td>
</tr>
<tr>
<td>Path41</td>
<td>From BasicNode27 to ShuntingYard2</td>
<td>False</td>
<td>False</td>
<td>250</td>
</tr>
<tr>
<td>Path42</td>
<td>From BasicNode29 to BasicNode27</td>
<td>False</td>
<td>False</td>
<td>250</td>
</tr>
<tr>
<td>Path43</td>
<td>From BasicNode29 to Input@Server2</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path44</td>
<td>From Input@Server2 to BasicNode29</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path45</td>
<td>From BasicNode28 to BasicNode27</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path46</td>
<td>From BasicNode28 to Input@Server1</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path47</td>
<td>From Input@Server3 to BasicNode28</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path49</td>
<td>From BasicNode30 to BasicNode32</td>
<td>False</td>
<td>False</td>
<td>2</td>
</tr>
<tr>
<td>Path50</td>
<td>From BasicNode30 to BasicNode31</td>
<td>Bidirectional</td>
<td>False</td>
<td>2</td>
</tr>
<tr>
<td>Path52</td>
<td>From BasicNode31 to BasicNode32</td>
<td>Bidirectional</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path53</td>
<td>From BasicNode31 to BasicNode33</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path54</td>
<td>From BasicNode33 to BasicNode34</td>
<td>False</td>
<td>False</td>
<td>20</td>
</tr>
<tr>
<td>Path55</td>
<td>From BasicNode33 to BasicNode34</td>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Path</td>
<td>From</td>
<td>To</td>
<td>DrawnToScale</td>
<td>LogicalLength</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>----</td>
<td>--------------</td>
<td>---------------</td>
</tr>
<tr>
<td>126</td>
<td>Input@ProductALoading</td>
<td>Output@ProductALoading</td>
<td>False</td>
<td>50</td>
</tr>
<tr>
<td>142</td>
<td>Input@ProductB_ProductCLoading</td>
<td>Output@ProductB_ProductCLoading</td>
<td>False</td>
<td>50</td>
</tr>
<tr>
<td>143</td>
<td>Output@ProductB_ProductCLoading</td>
<td>BasicNode78</td>
<td>False</td>
<td>240</td>
</tr>
<tr>
<td>144</td>
<td>BasicNode1</td>
<td>Shunting1_Inbound</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
<tr>
<td>63</td>
<td>BasicNode31</td>
<td>BasicNode39</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
<tr>
<td>36</td>
<td>BasicNode24</td>
<td>Input@Server1</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
<tr>
<td>37</td>
<td>Input@Server1</td>
<td>Output@Server1</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
<tr>
<td>82</td>
<td>Input@Sink2</td>
<td>Output@Source2</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
<tr>
<td>Connector1</td>
<td>MemberOutput@ShuntLine1_InboundShuntingYard3</td>
<td>ParentOutput@ShuntLine1_InboundShuntingYard3</td>
<td>Bidirectional</td>
<td>False</td>
</tr>
</tbody>
</table>
LogicalLength: 340
Path156 (Path)
From BasicNode54 to ParentInput@ShuntLine1_Outbound
Type: Bidirectional
InitialTravelerCapacity: 1
DrawnToScale: False
LogicalLength: 400
Path158 (Path)
From BasicNode54 to MemberInput@ShuntLine1_Outbound
Type: Bidirectional
InitialTravelerCapacity: 1
DrawnToScale: False
LogicalLength: 400
Path159 (Path)
From Output@ShuntLine1_Outbound to BasicNode46
Type: Bidirectional
InitialTravelerCapacity: 1
DrawnToScale: False
LogicalLength: 340
Path2 (Path)
From Shunting1_Inbound to Input@ExitShuntingYard1
DrawnToScale: False
EnteredAddOnProcess: CountExit Loads
Path149 (Path)
From BasicNode59 to BasicNode39
Path153 (Path)
From BasicNode17 to BasicNode46
Type: Bidirectional
InitialTravelerCapacity: 1
Connector2 (Connector)
From MemberOutput@ShuntLine2_Inbound to ParentOutput@ShuntLine2_Inbound
Path71 (Path)
From Output@ShuntLine2_OutboundShuntingYard3 to BasicNode44
Type: Bidirectional
InitialTravelerCapacity: 1
DrawnToScale: False
LogicalLength: 340
Path145 (Path)
From BasicNode44 to BasicNode59
Type: Bidirectional
InitialTravelerCapacity: 1
DrawnToScale: False
LogicalLength: 500
Path150 (Path)
From BasicNode54 to ParentInput@ShuntLine2_OutboundShuntingYard3
Type: Bidirectional
**Type:** Bidirectional  
**InitialTravelerCapacity:** 1

### Path 166 (Path)
**From:** BasicNode63 to BasicNode64  
**DrawnToScale:** False  
**LogicalLength:** 60

### Path 167 (Path)
**From:** BasicNode64 to BasicNode54  
**DrawnToScale:** False  
**LogicalLength:** 40

### Path 168 (Path)
**From:** BasicNode22 to Shunting1_Outbound  
**DrawnToScale:** False  
**LogicalLength:** 300

### Path 169 (Path)
**From:** Input@ExitShuntingYard1 to Shunting1_Inbound  
**DrawnToScale:** False  
**LogicalLength:** 500

### Path 170 (Path)
**From:** Input@ProductALoading to BasicNode25  
**DrawnToScale:** False  
**LogicalLength:** 500

### Path 171 (Path)
**From:** BasicNode27 to BasicNode30  
**DrawnToScale:** False

### Path 172 (Path)
**From:** Output@ArrivalsShuntingYard1 to Shunting1_Inbound

#### Tables **Show / Hide 6 items**

<table>
<thead>
<tr>
<th>GeneralInfo</th>
<th>RTCDestinations</th>
<th>Input@ProductALoading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Input@ProductB_ProductCLoading</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CoupleDecoupleTimes</th>
<th>Description</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decouple</td>
<td>Random.Triangular(1,2,5)</td>
<td></td>
</tr>
<tr>
<td>Couple</td>
<td>Random.Triangular(1,3,6)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ShuntLineInfo</th>
<th>ShuntLineWaitingArea</th>
<th>DecoupleEvents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Decoupled1</td>
<td></td>
</tr>
</tbody>
</table>
Decoupled2

### Coupling Destinations

<table>
<thead>
<tr>
<th>Shuntline1 Outbound</th>
<th>Shuntline2 Outbound</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParentInput@ShuntLine1_Outbound</td>
<td>ParentInput@ShuntLine2_Outbound</td>
</tr>
<tr>
<td>MemberInput@ShuntLine1_Outbound</td>
<td>MemberInput@ShuntLine2_Outbound</td>
</tr>
</tbody>
</table>

### Products

<table>
<thead>
<tr>
<th>ProductType</th>
<th>Probability</th>
<th>LoadDestination</th>
<th>ProdType</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlockTrainProductA</td>
<td>0.333</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>BlockTrainProductB</td>
<td>0.333</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>BlockTrainProductC</td>
<td>0.333</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

### Arrival Table

<table>
<thead>
<tr>
<th>ArrivalTimes</th>
<th>ArrivalQty</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>84</td>
<td>1</td>
</tr>
<tr>
<td>168</td>
<td>0</td>
</tr>
</tbody>
</table>

### Schedules

**Show / Hide 1 items**

**Standard Week**

- **Days:** 7
- **Start Date:** 2011-01-03T00:00:00

### Entity 1

**Author:** Thabiso

### Property Overrides **Show / Hide 14 items**

- **Override (ResourceIdleCostRate)**
  - **Visible:** False
- **Override (ResourceCostPerUse)**
  - **Visible:** False
- **Override (ResourceUsageCostRate)**
  - **Visible:** False
- **Override (LogResourceUsage)**
  - **Visible:** False
- **Override (DisplayCategory)**
  - **Visible:** False
- **Override (DisplayColor)**
  - **Visible:** False
- **Override (CapacityType)**
  - **Visible:** False
- **Override (WorkSchedule)**
  - **Visible:** False
- **Override (WorkDayExceptions)**
  - **Visible:** False
- **Override (WorkPeriodExceptions)**
  - **Visible:** False
- **Override (InitialCapacity)**
  - **Visible:** False
- **Override (RankingRule)**
  - **Visible:** False
**Override (RankingExpression)**

**Visible:** False

**Override (DynamicSelectionRule)**

**Visible:** False

**States**  
*Show / Hide 9 items*

- IntegerState (Destination)
- IntegerState (NumberWagons)
- IntegerState (WhichShuntLine)
- IntegerState (LoadNumber)
- IntegerState (WhichLoco)
- IntegerState (ParentOrMember)

**Description:** 1=Parent 2=Member

- RealState (LoadTime)
- IntegerState (ProdType)
- RealState (ArrivalTime)

**InitialValueUnits:** Hours

**UnitType:** Time

---

**Processor**

**Author:** Thabiso

**Property Overrides**  
*Show / Hide 14 items*

- Override (ResourceIdleCostRate)
  **Visible:** True

- Override (ResourceCostPerUse)
  **Visible:** True

- Override (ResourceUsageCostRate)
  **Visible:** True

- Override (LogResourceUsage)
  **Visible:** True

- Override (DisplayCategory)
  **Visible:** True

- Override (DisplayColor)
  **Visible:** True

- Override (CapacityType)
  **Visible:** True

- Override (WorkSchedule)
  **Visible:** True

- Override (WorkDayExceptions)
  **Visible:** True

- Override (WorkPeriodExceptions)
  **Visible:** True

- Override (InitialCapacity)
  **Visible:** True

- Override (RankingRule)
  **Visible:** True

- Override (RankingExpression)
  **Visible:** True
Visible: True
Override (DynamicSelectionRule)
Visible: True

Elements  Show / Hide 1 items
Processing (Station)

Processes  Show / Hide 1 items
MainProcess
TriggeringEventName: Processing.Entered
TokenActionOnAssociatedObjectDestroyed: EndProcess

Steps
IntoProcessing (EndTransfer)
ToOutput (Transfer)
FromType: CurrentStation
ToType: ParentExternalNode
ExternalNodeName: Output

Transfer Points  Show / Hide 2 items
Input
NodeClassName: BasicNode
InputLocationType: Station
StationName: Processing
Output
NodeClassName: TransferNode

MyVehicle
Base: Vehicle
Author: Thabiso

States  Show / Hide 2 items
IntegerState (WhichShuntLine)
IntegerState (Direction)
Description: 1=Inbound 2=Outbound

MySeparator
Base: Separator
Author: Thabiso