

ScenarioBuilder

Version 0.23 working (This document needs updating)

October 2015

Brian Farrimond

Contents

INTRODUCTION	4
INSTALLING SCENARIOBUILDER	5
RUNNING SCENARIOBUILDER	6
SCENARIOBUILDER MENUS	7
FILE	7
EDIT	7
BUILD	7
EXPORT	8
TOOLS	8
CREATING A NEW SCENARIO	9
ENTERING A NARRATIVE	11
BUILDING THE SCENARIO	11
<i>Showing / hiding ship bow waves, wakes and funnel smoke</i>	12
NARRATIVE ERROR REPORTING	13
CONTROLLING THE SIMULATION	14
<i>Adding the sea tide</i>	14
<i>Selecting viewpoints</i>	15
<i>Running the simulation</i>	16
<i>Controlling the time steps and speed</i>	16
<i>Controlling atmospheric effects</i>	16
ADDING A SHIP AND A LOCATION TO THE SCENARIO	17
VIEWING THE SCENARIO IN 2D	18
<i>Manipulating the 2D view</i>	18
<i>Wind direction and location indicators</i>	19
<i>Universal Transverse Mercator System</i>	19
<i>Running the simulation in 2D</i>	19
SHIP WAKES AND SMOKE	20
<i>Bow wave</i>	20
<i>Boundary layer</i>	21
<i>Turbulence from ship paddles</i>	21
<i>Bow wave, wake and turbulence scene graph elements</i>	22
IMPLEMENTING A SHIP'S FUNNEL SMOKE WITH DYNAMIC NODES	22
EXPORTING A SCENARIO	23
EXPORTING THE RAW XML	23
EXPORTING TO SCENARIOVIEWERJ3D	ERROR! BOOKMARK NOT DEFINED.
USING A NARRATIVE INSIDE ANOTHER SCENARIO	24
THE NARRATIVE GRAMMAR	25
INTRODUCTION	25

AN EXAMPLE NARRATIVE	26
SCENARIOBUILDER NARRATIVES.....	27
SHORT EXAMPLE	27
GEOGRAPHY AND TIME NARRATIVE CONVENTIONS	27
THE NARRATIVE LANGUAGE.....	28
SCENARIO INFORMATION SECTION.....	29
<i>Scenario scenes</i>	29
<i>Starting and ending scenarios</i>	29
<i>Starting and ending simulations</i>	30
<i>Specifying the tides</i>	30
<i>Using scenario narratives</i>	30
CAST LIST	30
<i>Specifying Locations</i>	31
<i>Specifying Ships</i>	31
<i>Manually controlling a ship</i>	31
<i>Specifying trains</i>	32
ACTIONS	32
<i>Sea state</i>	32
<i>Weather</i>	33
<i>Initial ship placement</i>	33
<i>Docking of ships</i>	33
<i>Sailing of ships</i>	34
<i>Ship positions</i>	34
<i>Ship heading and speed</i>	35
<i>Initial train placement</i>	36
<i>Moving of trains</i>	36
<i>Manually controlled trains</i>	36
LONGER EXAMPLE	36
GEOGRAPHICAL TYPES	38
TIME TYPES	39
Time phrases used in sentences	39

Introduction

ScenarioBuilder creates interactive 3D visualisations of historical scenarios. ScenarioBuilder uses textual descriptions of scenarios and a library of dynamic 3D models to generate the 3D simulation. The textual description is in the form of a narrative using natural English with certain restrictions to the grammar. The library of 3D dynamic models is created with the **Thing Builder** tool.

The narrative consists of sentences such as "*SS Orion docks in Liverpool Docks at berth 124 at 11:00 am on 23 Jan 1888.*" **Thing Builder** is used to build a dynamic model of *Liverpool Docks* including the routes to be followed when docking at berth 124. It is also used to build the ship *SS Orion*. The user can specify multiple viewpoints on the harbour and ship.

Once the models have been built and the narrative written, the simulation can run in real time and the user hop from viewpoint to viewpoint as the simulation runs. Figure 1 shows the **ScenarioBuilder** interface with the textual narrative on the left and the 3D visualisation on the right.

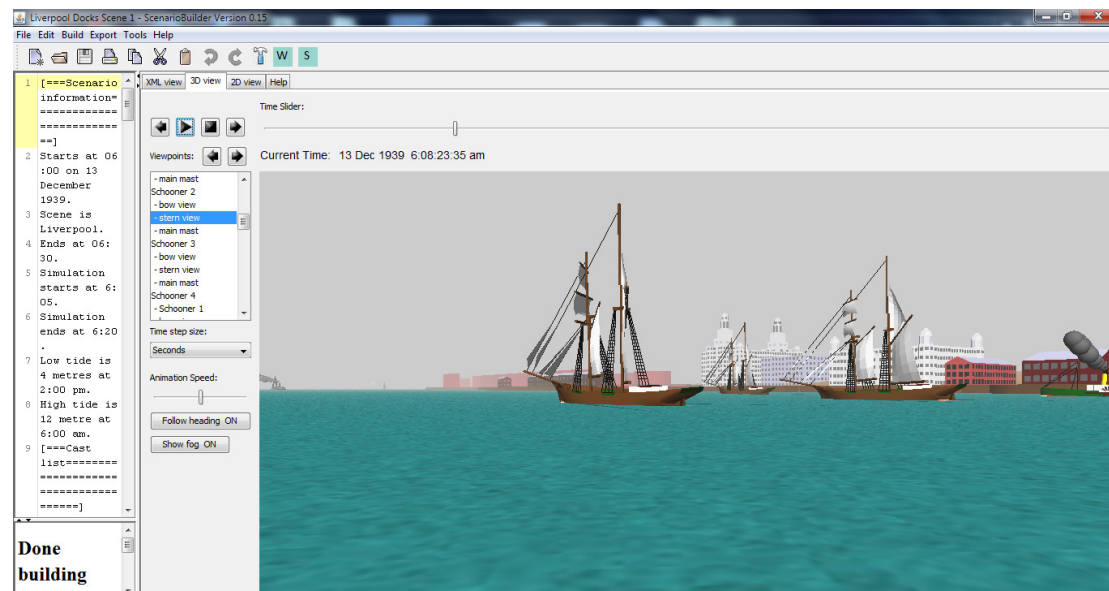


Figure 1: 3D visualisation running in ScenarioBuilder

A **ScenarioBuilder** scenario starts as a textual narrative and is **built** into a 3D visualisation.

An existing scenario can be visualised within Scenario Builder by loading its narrative and rebuilding it. Alternatively a built narrative can be **exported** as a set of files that can be visualised in a separate **ScenarioViewer** program. The set of files will include copies of the VRML files that make up the 3D dynamic models.

Installing ScenarioBuilder

ScenarioBuilder is a Windows program and runs on Windows 2000, Windows XP, Windows 7. It is not yet tested on Windows 10. It is installed by using a setup file. The current setup file is *scenariobuildersetup023.exe*.

Obtain a copy of the setup file and launch it by double clicking on its icon in Windows Explorer.

The default installation places **ScenarioBuilder** in the folder *c:\ScenarioBuilder*.

ScenarioBuilder is a Java 3D application and so the Java Runtime environment must be installed for it to run. The Java 3D libraries are included in the installation.

There are issues with Java 3D and Java 7 so we currently use a Java 6 JRE.

ScenarioBuilder works successfully with the JRE installed by *jre-6u24-windows-i586.exe* .

Running ScenarioBuilder

Once installed, run **ScenarioBuilder** like any other installed program, through the Start menu or the Desktop shortcut that may be optionally installed.

When **ScenarioBuilder** starts, the screen shown below in Figure 2 is opened.

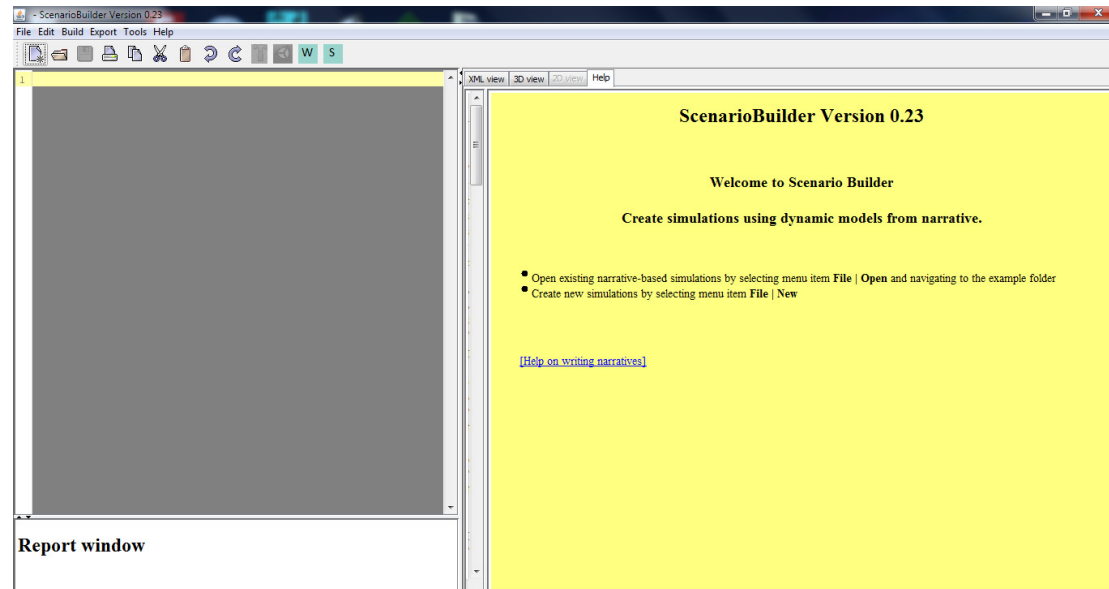


Figure 2: The ScenarioBuilder opening screen

The screen is divided into three main panels:

- textual narratives will appear in the upper left panel
- error messages and reports appear in the lower left panel.
- the 3D visualisation will appear in the right hand panel.

ScenarioBuilder Menus

File

New Scenario...	Create a new scenario
Open Scenario...	Open an existing scenario
Save Scenario	Save the currently opened scenario
Save Scenario As..	Save the currently opened scenario under a new name and/or new directory.
Save Narrative For Use	Save narrative text for use in other scenarios
Close Scenario	Close the scenario
Scenario Properties...	Edit the scenario properties

Edit

These edit functions all refer to editing the narrative in the left hand panel.

Undo	Undo last text edit.
Redo	Redo last text edit.
Cut	Cut selected text.
Copy	Copy selected text.
Paste	Paste selected text.
Find...	Find text within the narrative text.
Replace...	Replace text within the narrative text.
Select All	Select all the narrative text.

Build

Build Scenario	Generates the 3D simulation from the narrative and displays it in the right hand panel under the 3D View tab. Once built it can also be viewed in 2D under the 2D view tab in the right hand panel.
-----------------------	--

Export

Export Scenario	Export as a set of files that can be processed to enable the scenario to be viewed in other viewers
Export Scenario to Unity 3D	Export as a set of files that can be immediately viewed in the ScenarioViewerU3D Unity 3D application.
Export Scenario to ScenarioViewerJ3D	Export as a set of files that can be immediately viewed in the ScenarioViewerJ3D

Tools

Options	Set options for the ScenarioBuilder - see below.
Clean example and test folders	This clears out those scenarios in the examples and test folders. It deletes the copies of models resulting from building a scenario and any products of the Export Export Scenario menu item.
Debug ON Debug OFF	Toggle debugging reports in output - little used at present.

The menu item **Tools | Options** opens the dialog shown below in Figure 3:

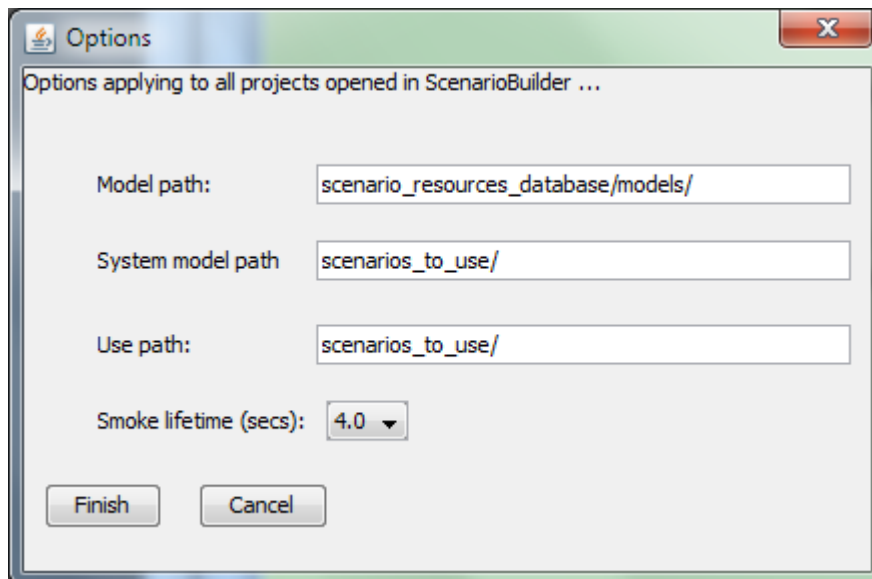


Figure 3: The Options dialog

Model path: the path within the ScenarioBuilder installation to the models imported from ThingBuilder.

System model path: additional models (not currently used)

Use path: the path to the scenarios that can be used in other scenarios

Smoke lifetime (secs): the number of seconds each main smoke puff lasts. The longer this is then the greater demands on graphics processing

Creating a new scenario

To create a new scenario, click on the menu item **File | New Scenario...** This opens the dialog shown below in Figure 4 .

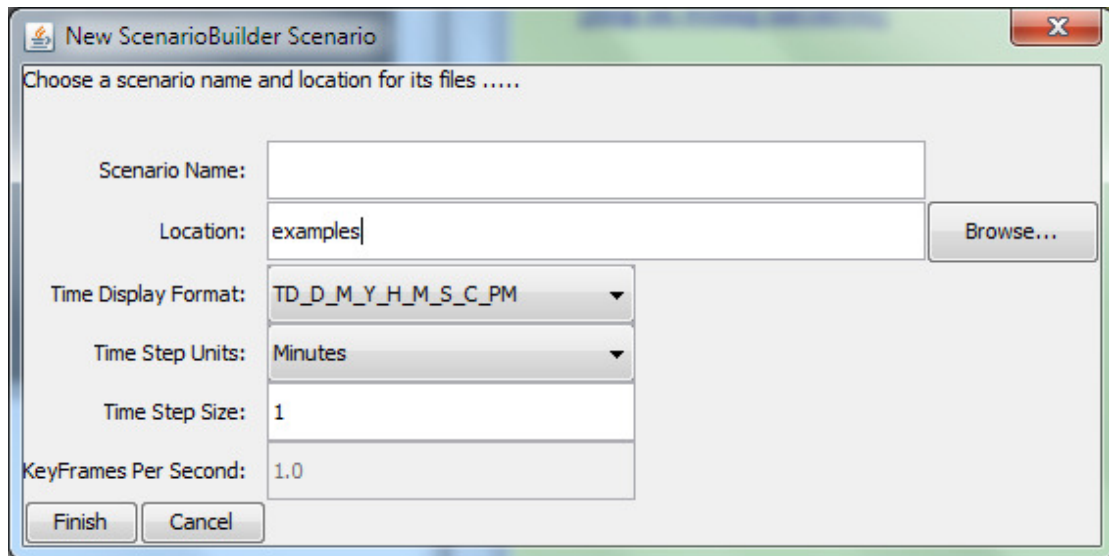


Figure 4: New scenario dialog

Explanation of the fields in the dialog:

- The **Scenario Name** and **Location** fields are used to identify the scenario and locate it on disc.
- **Time Display Format** determines how the current time within a scenario will be displayed during 3D visualisation. The default one shown appears as shown in Figure 5 below.

Current Time: 13 Dec 1939 6:57:00:00 am

Figure 5: Time display TD_D_M_Y_H_M_S_C_PM

- **Time Step Units** refer to the simulation controls that enable the user to step backwards and forwards through time. This value gives the time units for each step
- **Time Step Size** determines how many units of time to jump on each click of a step forward or step back button. This is currently ignored and treated as 1.0 within **ScenarioBuilder**.
- **KeyFrames Per Second**. When the scenario simulation is running, the values for the locations, heading, pitch and roll etc of the ships and other components are determined from keyframes which contain their values at a regular intervals of time. ScenarioBuilder interpolates between keyframes bracketing the current time. The KeyFrames Per Second value determines the number of keyframes to be generated per second of simulation time when the simulation is built. The rate is currently fixed at 1 keyframe per second.

Clicking on the **Finish** button opens the screen shown in Figure 6 below:

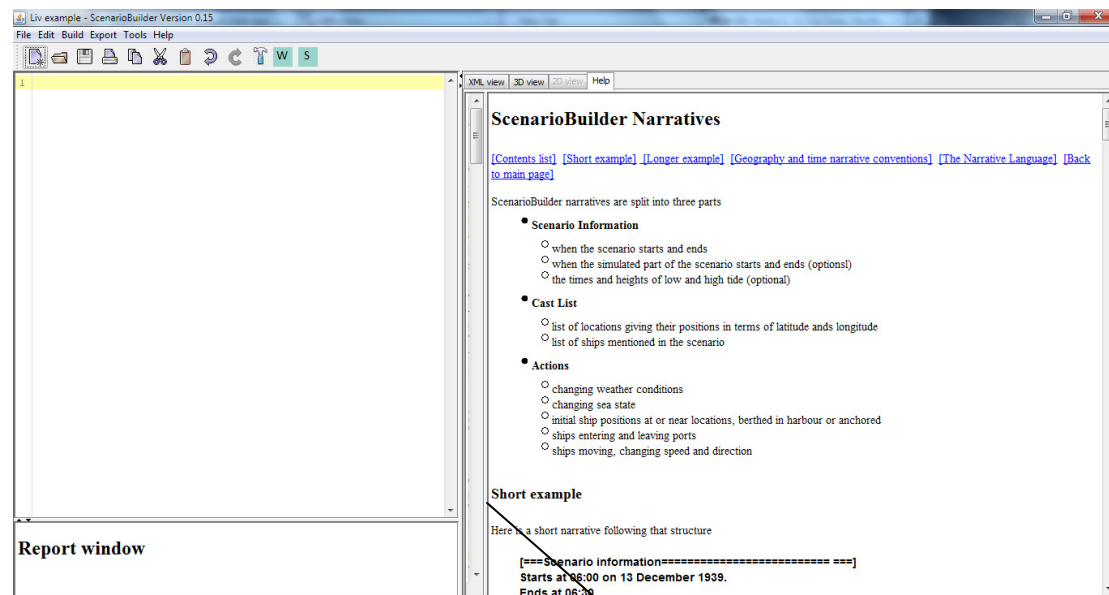


Figure 6: A new scenario

The left hand panel is a text editor where the narrative is typed (or pasted from another document). The right hand panel displays help on the narrative grammar. Dragging to the right as indicated reveals the available models as shown in Figure 7 below.

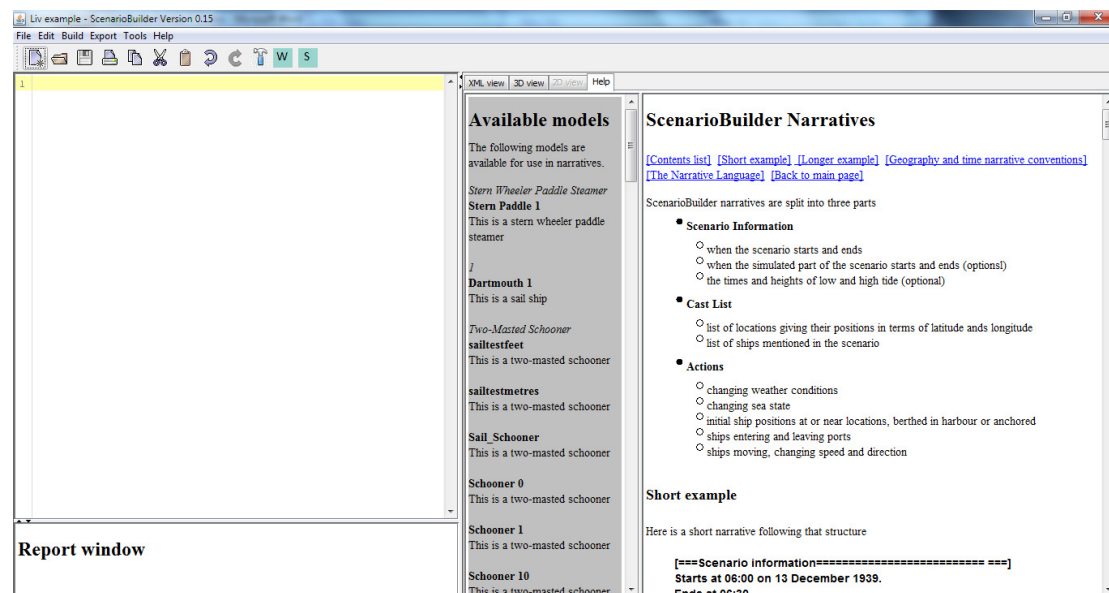


Figure 7: Seeing the available models

ThingBuilder models are exported to *ScenarioBuilder's* model database. The central panel lists details of the models currently in the model database. These are the objects that can be mentioned in the scenario narrative.

Entering a narrative

We shall start with a very simple narrative:

Starts at 06:00 on 13 December 1939.

Scene is Liverpool.


Ends at 06:30.

These three lines can be copied and pasted into the left hand panel or typed afresh there.

The first line gives the scenario start time, the last line gives the scenario end time. The middle line is a sentence identifying the background scene to be used in the scenario..

A later section explains all the available sentences for narratives.

Building the scenario

To build the scenario either click on the menu item **Build | Build Scenario** or click on the tool bar button . This results in the screen appearing as shown below in Figure 8:

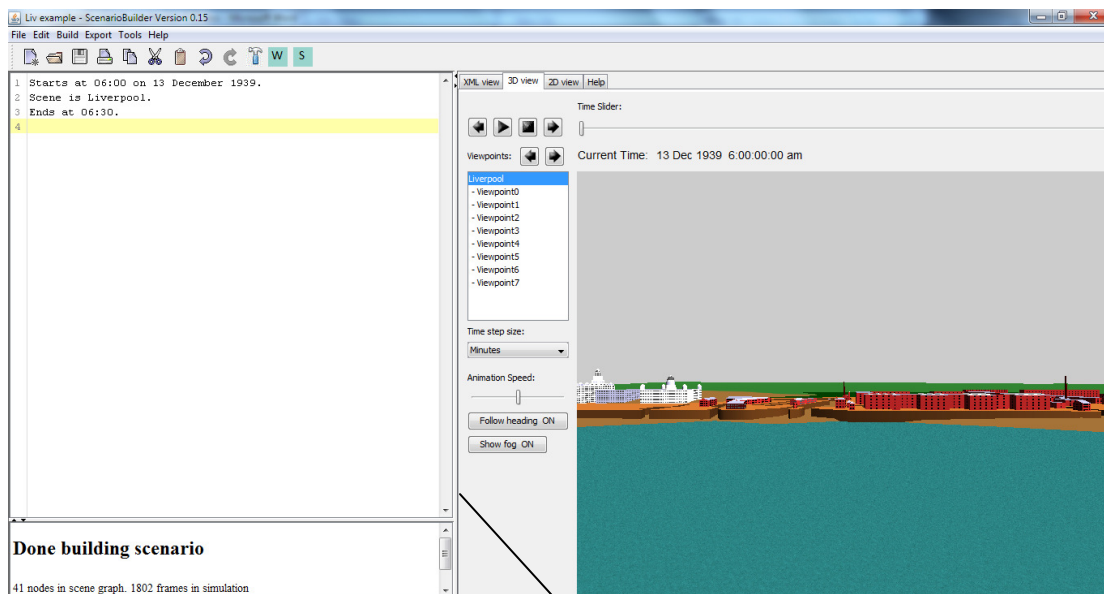
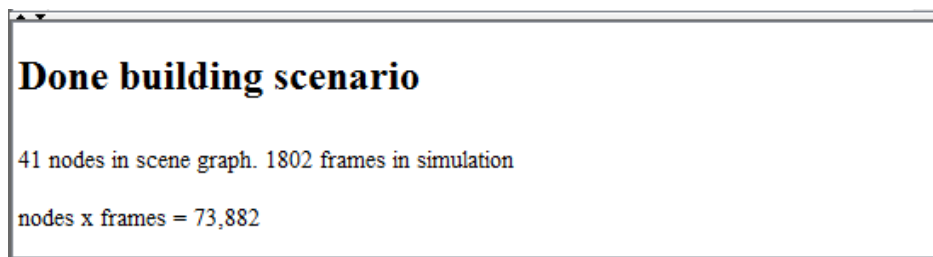


Figure 8: The built scenario

The splitter boundary can be used to maximise the right hand panel to improve the 3D view.

The bottom left panel reports on the size of the simulation.



If there are too many nodes in the scene graph for the machine to cope with then movement will be jerky. A decent machine can cope with up to around 2000 nodes,





If there are too many frames then the machine may run out of memory and the program crashes.

You can reduce the number of nodes by:

- using fewer ships
 - note that a ship that does not move during the simulation requires fewer nodes than one which does move i.e. anchored and berthed ships are comparatively cheap and add to the ambience.
- not showing ship bow waves and wake
- not showing smoke from ship funnels
 - alternatively, by reducing the lifetime of smoke puffs in the menu **Tools | Options**

Showing / hiding ship bow waves, wakes and funnel smoke

The toolbar buttons are used to toggle showing or hiding these features as listed in the following table:

 	Show / hide ship bow waves and wakes
 	Show / hide ship funnel smoke

Narrative error reporting

Errors in the narrative are reported in the bottom left pane. Errors include:

- using names not present in the available models
- breaking the *ScenarioBuilder* grammar rules

Figure 9 shows an error of the first type. Note the highlighting of the problem sentence. Use the list of available models to help correct the mistake.

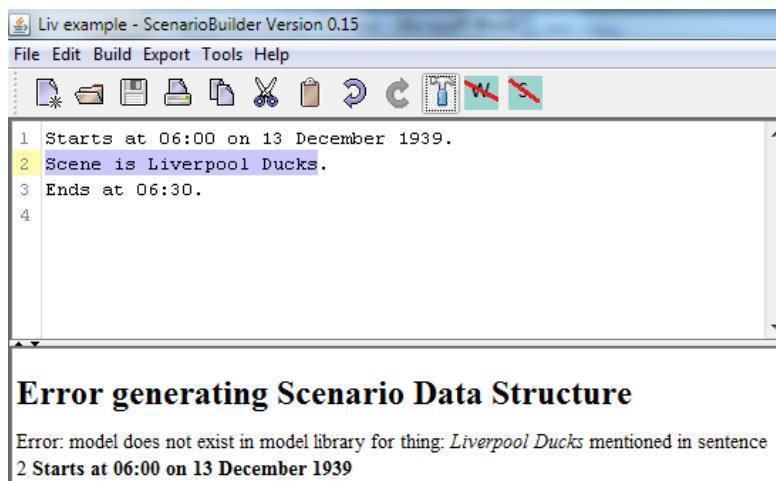


Figure 9: Narrative error using non-existent model

Figure 10 shows an error of the second type. Sentences need to be in the present tense - **Scene is Liverpool**. The grammar is restricted so that ScenarioBuilder does not have to have a very complex system for understanding it.

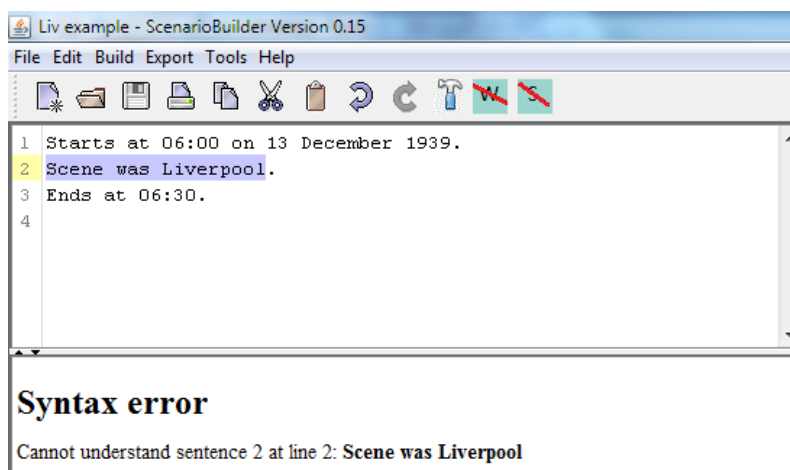


Figure 10: Narrative error using disallowed grammar

Controlling the simulation

Adding the sea tide

Figure 8 shows the tide is out. This is because the narrative has not yet identified the state of the tide. Here is a modified narrative:

Starts at 06:00 on 13 December 1939.

Scene is Liverpool.

Ends at 06:30.

High tide is 12 metre at 6:00 am.

Low tide is 4 metres at 2:00 pm.

The last two sentences specify the behaviour of the tide during the narrative. ScenarioBuilder will set the tide at each time during the simulation using this information. Thus our scenario starts when the tide is at its highest. This is shown below in Figure 11

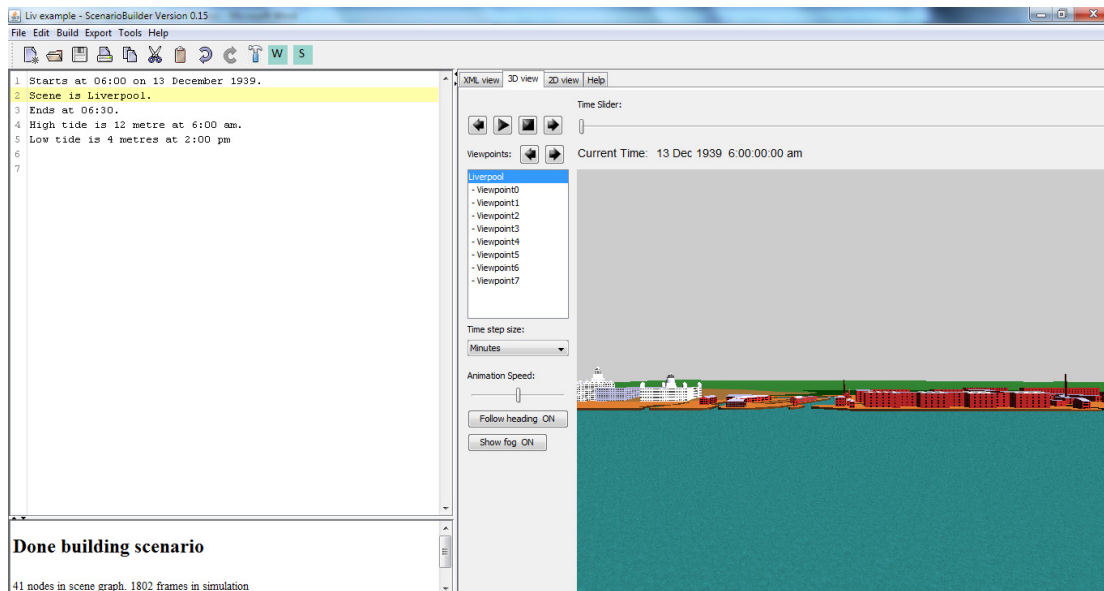


Figure 11: Liverpool at high tide

Selecting viewpoints

The 3D view panel includes a list box containing the available viewpoints. The viewpoints are specified in ThingBuilder for dynamic 3D models. For Liverpool, the first viewpoint is a scene setting view you can see in .Figure 11 Switch viewpoint by clicking in the list of viewpoints or using the Next viewpoint and Prev viewpoint buttons. Figure 12 and Figure 13 show two of the other viewpoints:

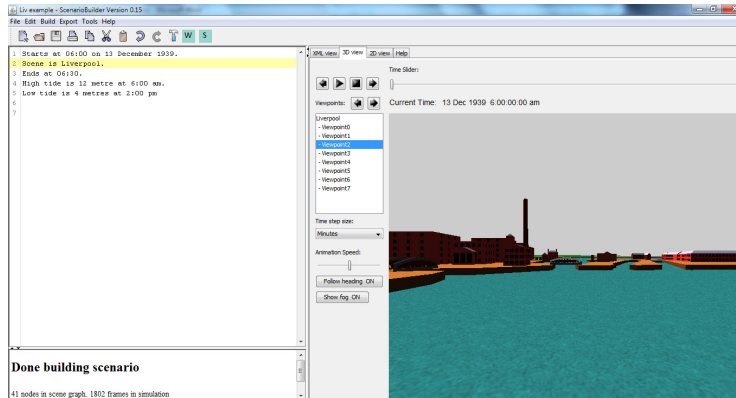


Figure 12: Liverpool Viewpoint2

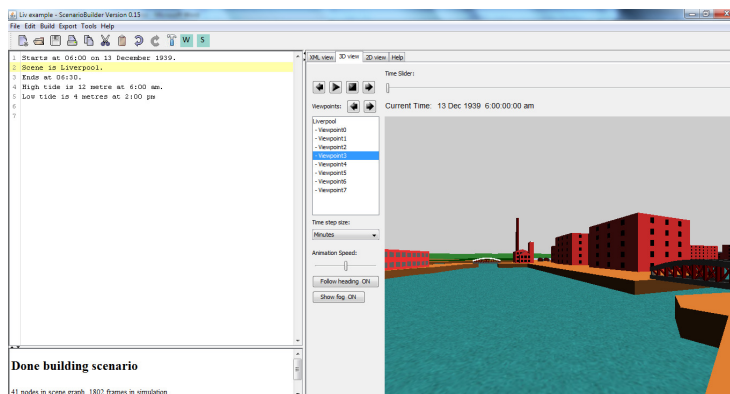


Figure 13: Liverpool Viewpoint3

Control the view by using mouse or keyboard as follows:

Turn left and right	Mouse drag with left button down Left and right arrow keys
Look up or down	Mouse drag with right button down Up and down arrow keys
Zoom in and out	Mouse drag with centre button down 'A' and 'Z' keys
Move back from the turning point to view it	'S' and 'Z' keys. 'D' and 'C' keys for greater range

Running the simulation

The simulation run is controlled by the controls shown below in Figure 14.

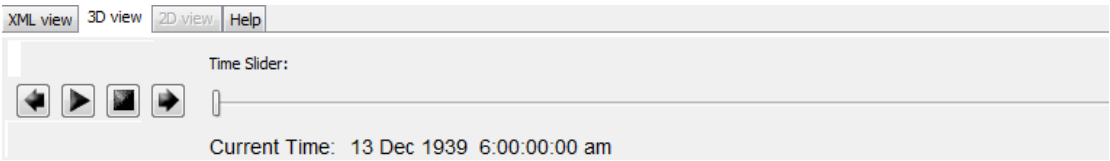


Figure 14: Simulation run controls

The time slider can be used to move quickly to a desired time.

	Step back in time
	Toggle between <i>Run freely</i> and <i>Pause</i>
	Stop the simulation returning the timee to the start time.
	Step forward in time

Controlling the time steps and speed

Figure 15 shows the controls used to step the period of time steps and the animation speed.

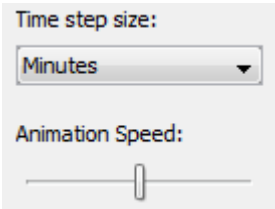


Figure 15: Time step and run speed controls

Controlling atmospheric effects

The narrative can control meteorological visibility by narrative sentences such as **Visibility is. 1000 metres.**

This effect can be switched off to show the simulation in full clarity using the button **Show fog ON/OFF** below the viewpoint buttons.

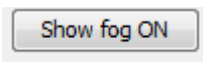


Figure 16: Show fog ON/OFF button

Adding a ship and a location to the scenario

We can add a ship to the scenario, for example as follows:

Starts at 06:00 on 13 December 1939.

Wind is heading 270 at speed 9 knots.

Scene is Liverpool.

A is located at 53.444N, 3.0234W.

Side Paddle 1 is a Liverpool ship.

Side Paddle 1 is initially berthed in Liverpool facing south at berth Berth3.

At 06:01 Side Paddle 1 sails unassisted.

At 06:30 Side Paddle 1 is at A.

Ends at 06:30.

High tide is 12 metre at 6:00 am.

Low tide is 4 metres at 2:00 pm.

The ship, *Side Paddle 1* is initially berthed in the docks then sails at 06:01 am, arriving at location A (actually at the mouth of the River Mersey) at 06:30 am. The ship sails unassisted. In a future version of Scenario Building ship sailings and berthings will optionally be assisted by tug boats. Figure shows the ship passing by viewpoint1 on its way to location A.

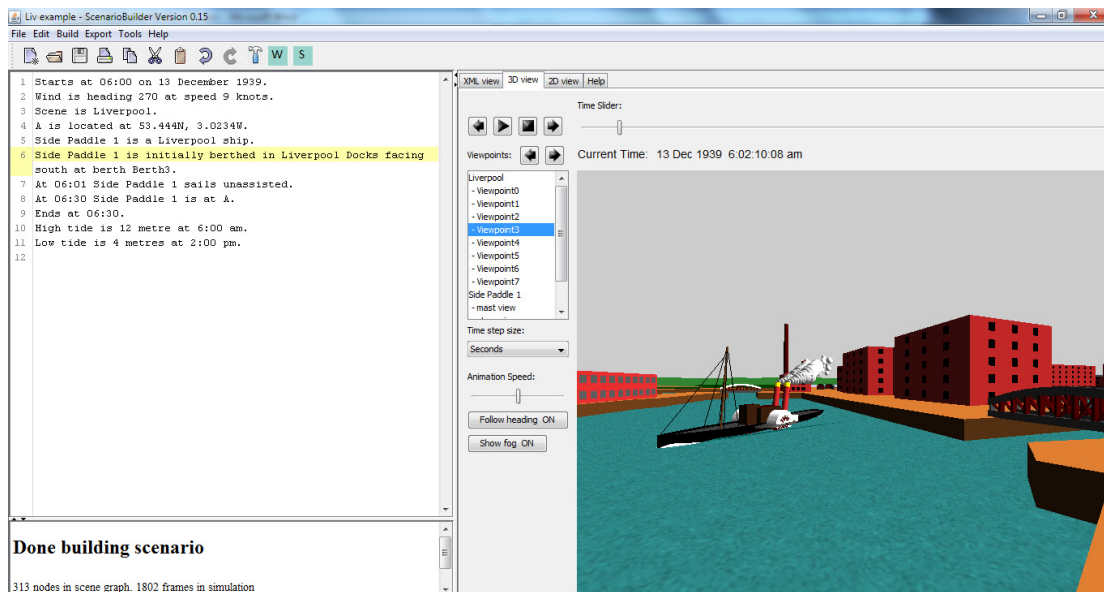


Figure 17: Side Paddle 1 passing viewpoint3 on its way to A.

Viewing the scenario in 2D

The scenario can be run in a 2D map view by clicking on the **2D view** tab in the right hand panel.

The scenario given above initially looks like Figure 18 in 2D.

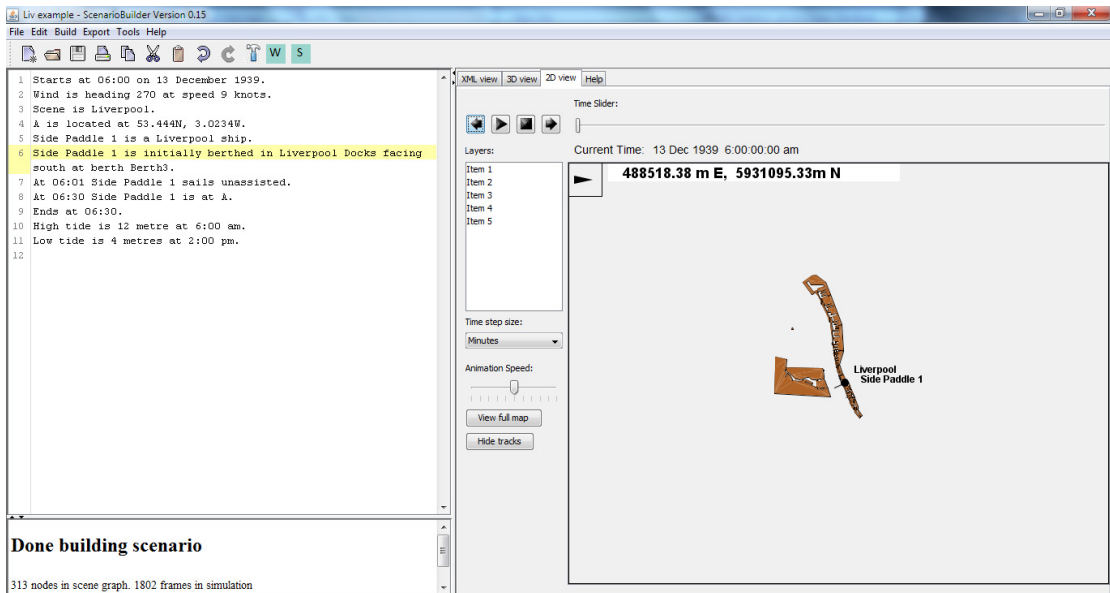


Figure 18: Initial 2D view of the scenario

Manipulating the 2D view

Manipulation of the 2D view is achieved using the mouse as follows:

Zoom in and out	Drag mouse with centre button pressed.
Pan across the map	Drag mouse with right button pressed.

Figure 19 shows the screen after panning and zooming.

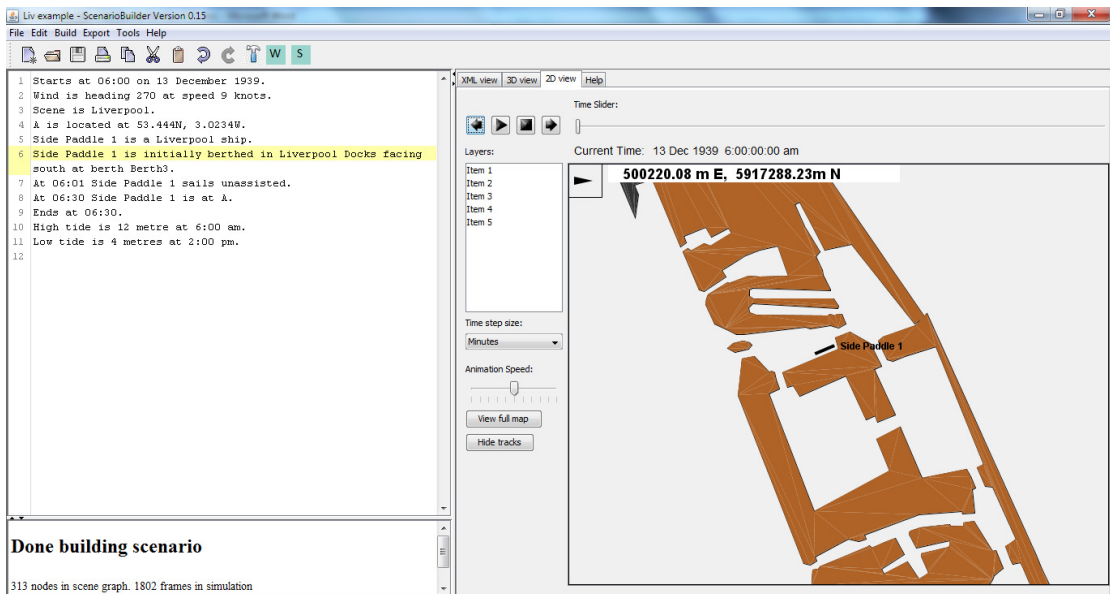


Figure 19: 2D view panned and zoomed

Wind direction and location indicators

The wind direction and cursor location are indicated at the top left of the 2D view as shown in detail in .

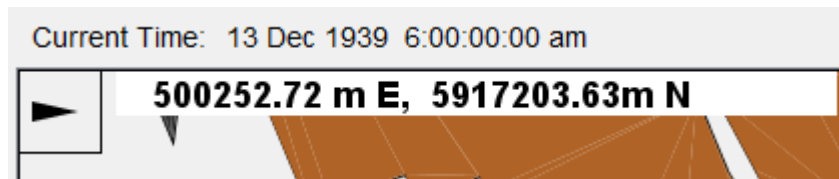


Figure 20: Wind direction and cursor location

The black triangular arrowhead is pointing to the direction the wind is blowing TO. In the figure it indicates that the wind heading is 270 since that is the direction the wind is coming FROM.

The numbers *500252.72 m E, 5917203.63 m N* give the cursor position on the map using the Universal Transverse Mercator System (UTM).

Universal Transverse Mercator System

This system is an alternative way to latitude and longitude of representing location on the earth's surface. Since the earth is nearly spherical in shape, the lines of longitude get closer together as we move towards the poles. This means that two points 1 degree of longitude apart on the equator are further apart than two points 1 degree of longitude apart away from the equator. The further away from the equator, the closer the two points get until they coincide at the poles. In order to avoid this distortion when measuring distances, the UTM was invented which splits the earth's surface between 80° S and 86° N into 60 rectangular zones each 6° of longitude wide. Inside each zone a location is identified by a *northing* number of metres from the zone origin and an *easting* number of metres from the zone origin. This is a rectangular coordinate system so the distortion of the longitude lines is removed.

ThingBuilder and ScenarioBuilder use UTM to create and use geographical features such as harbours. The movements of ships is thus made easier to calculate. Narratives however stick to latitude and longitude to identify locations.

The layers list is not yet implemented.

Running the simulation in 2D

The controls for running, stepping forwards and backwards and the slider to quickly go to a desired time operate in the same way as in the 3D view.

There are two other buttons:

View full map	This button is used to return to the initial view showing the full scenario extent
Show tracks	This button toggles the displaying of the routes followed by moving objects such as ships.

Ship wakes and smoke

The visualisation of the interaction between a ship and its environment is enhanced by adding the ship's bow wave, wake and funnel smoke.

We are carrying out experiments to model the interaction of ships with the sea. The real world interaction is very complex and modelling it accurately is a major contribution to marine engineering. For our purposes we provide a much simplified model that gives the viewer some impression of the effect of the movement of the ship through water.

Like many other aspects of the builder tools, there is ample opportunity to continue refinement.

We aim to model the following:

- bow wave
- boundary layer
- sea turbulence from paddles on paddle steamers

Bow wave

The **bow wave** models the divergent and transverse waves generated as the ship disturbs the water it moves through. Figure 21 shows these waves (from <http://en.wikipedia.org/wiki/File:Fjord.surface.wave.boat.jpeg>)



Figure 21: Bow wave

The angle of the apex of the triangular envelope containing the waves to the centre line of the ship is fixed regardless of the size or speed of ship and is referred to as the Kelvin angle. Its value is approximately 20 degrees.

Our model (shown in Figure 22: Ship bow wave from above and side) represents this with a white bow wave front and a dark blue sinusoidal wave angled at 20 degrees on either side of the ship's centre line. The height and length of each are dependent on the speed of the ship. The height of the bow wave oscillates with the sea state period.

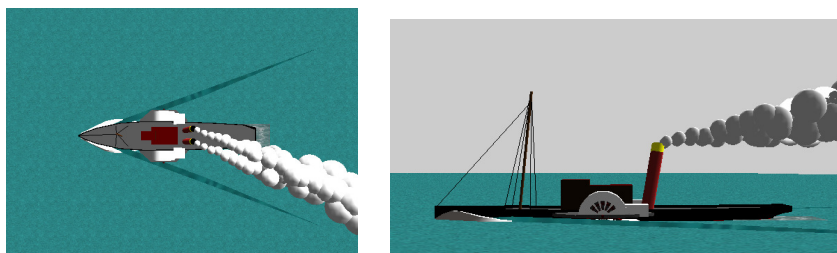
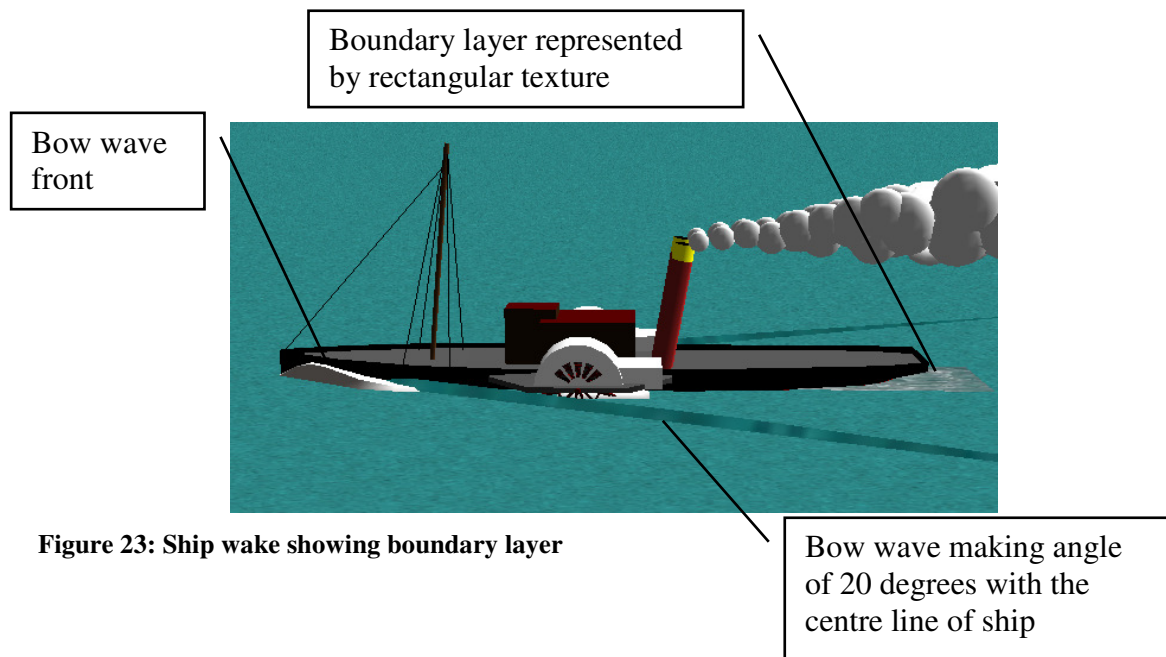


Figure 22: Ship bow wave from above and side

Boundary layer

The boundary layer is the disturbance in the water resulting from the ship hull "dragging" along the water as it passes through it.

This is modelled by a rectangular texture whose width and length depend upon the ship speed. This is shown below in Figure 23.



Turbulence from ship paddles

The turbulence from ship paddles is modelled as a rectangular white block whose height and length depends on the speed of the rotation of the paddle wheel. An example is shown in Figure 24.

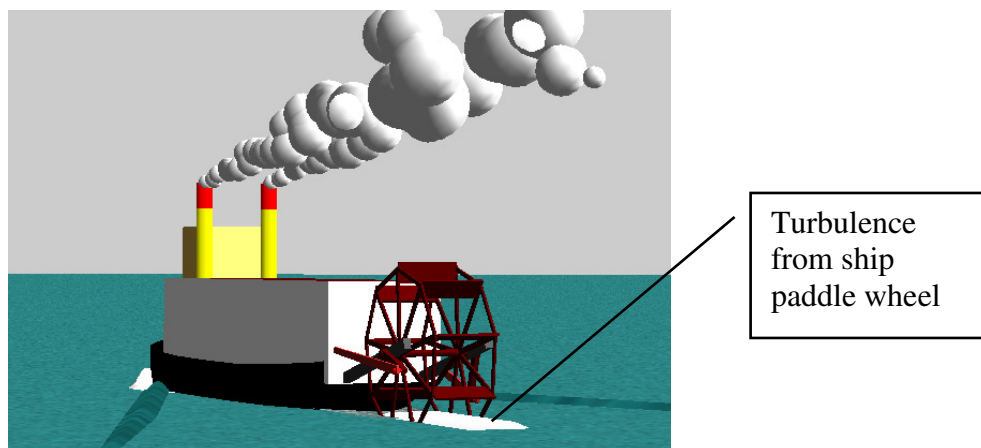


Figure 24: Turbulence from a stern paddle wheel

Bow wave, wake and turbulence scene graph elements

The building of a project causes additional VRML files to be created to model each ship's funnel smoke, bow wave, boundary layer and paddle wheel turbulence.

Implementing a ship's funnel smoke with dynamic nodes

Smoke is implemented by using copies of a VRML model that represents a puff of smoke. The first attempt uses a sphere for the model.

A puff follows a cycle of duration `m_lifetime` seconds. The cycle starts with the puff just above the owning funnel then the puff moves according to the wind for `m_lifetime` seconds. After `m_lifetime` seconds, the cycle begins again by the puff being located above the new position of the funnel. The set of puffs for a funnel are given staggered start times so that there appears to be a regular stream of puffs of smoke.

Superimposed on this is another set of puffs whose lifetimes are twice `m_lifetime`. These produce the separated wisps of smoke as shown below in .

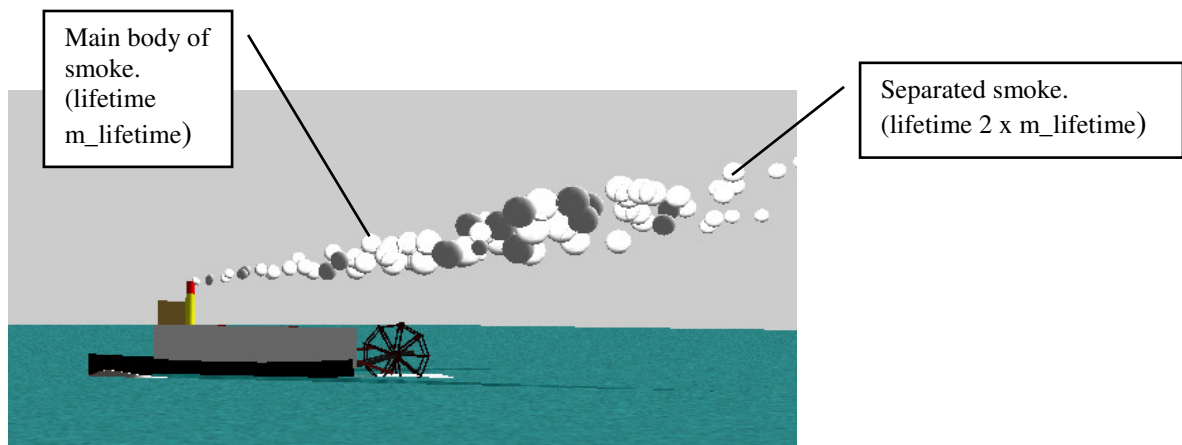


Figure 25: Smoke puffs from funnel

Exporting a scenario

Exporting the raw XML

This is achieved using the menu item **Export | Export Scenario**.

The set of files are stored inside the subfolder *export_pack*.

Scenarios can be exported for display in dedicated viewers.

Exporting to a viewer

Java 3D Viewer

ScenarioViewerJ3D is a viewer which has been developed using Java3D.

ScenarioBuilder generates the necessary files to be read by **ScenarioViewerJ3D**.

The generated files are placed in the folder identified by the *Java 3D Viewer Export path* which can be edited through the menu item: **Tools | Options**.

For **ScenarioBuilderJ3D**, compression and decompression is achieved using GZip.

Unity 3D Viewer

ScenarioViewerU3D is a viewer implemented using the **Unity 3D** gaming platform.

ScenarioBuilder generates the necessary files to be read by **ScenarioViewerU3D**.

The generated files are placed in the viewer's examples folder identified by the *Unity 3D Export path* which can be edited through the menu item: **Tools | Options**. When the viewer is launched, it detects the scenarios that have been exported to the examples folder and puts their names in a drop down list. The user can then select scenarios to run from this list.

Using a narrative inside another scenario

In order to use a scenario as a backdrop to new events, we can save the scenario ready to be used to this end.

This is achieved using the menu item **File | Save Narrative For Use** .

When it is saved, subsequent scenario narratives can use the sentence

Use scenario <scenario name>.

For example, we can

- open the scenario *Liverpool Scene 1*
- select menu item **File | Save Narrative For Use**
- close the scenario
- create a new scenario.
- Make the first line of its narrative

Use scenario Liverpool Scene 1.

The narrative grammar

Introduction

A **Scenario Builder** narrative consists of a sequence of sentences each of which is terminated by a full stop. The sentences are written in English but the **ScenarioBuilder** parser can only understand them if they follow certain rules. This section explains those rules and suggests approaches to checking that a narrative obeys them.

The parser processes each sentence in turn as follows.

- it identifies any date and time information present in the sentence and records it as a time stamp. The following rules also apply
 - If no date and time information was present then the previous time stamp (derived from the previous sentence) is used.
 - If the date and time information is partial (e.g. time of day only) then the missing information is obtained from the previous time stamp. Suppose we have the following narrative fragment: *Edward's army stops moving at 10:00 am on 25th April 1471. Edward's army starts moving at 2:00pm.* We assume that Edward's army starts to move at 2 pm on 25th April 1471.
- it removes the date and time information from the sentence.
- the remainder of the sentence is then searched for particular verb and verb phrases with the following sentence structure being assumed:

<subject> <verb> <object>

The processor generates records according to the particular verb or verb phrase found in the sentence.

If the <subject> is "It", "He", "She", "They" then the subject of the previous sentence is assumed.

An example narrative

Here is a typical narrative:

Starts at 06:00 on 13 December 1939. Wind is heading 270 at speed 9 knots. Sea state is 1.5 metres high and period is 8 seconds from heading 270.

Scene is Cape Town. A is located at 33.90S, 18.445E.

Side Paddle 1 is a British ship. Side Paddle 1 is initially berthed in Cape Town Harbour facing north at berth Berth B.

Schooner 0 is a German ship. Schooner 0 is initially berthed in Cape Town Harbour facing north at berth Berth0.

At 06:01 Side Paddle 1 sails unassisted. At 06:30 Side Paddle 1 is at A. Schooner 0 sails unassisted at 06:02. At 06:30 Schooner 0 arrives 1 mile north of location A.

Ends at 06:30.

The first paragraph is giving the start time and the state of the wind and the sea.

The second paragraph declares a scene (which includes Cape Town Harbour) and names a location at sea near Cape Town.

The third paragraph declares that a side paddle steamer is berthed in the harbour.

The fourth paragraph declares that a schooner is also berthed in the harbour.

The fifth paragraph describes the sailing of the two ships and their positions at 6:30 am.

The final sentence gives the end time of the scenario.

Note that all ships and places are case sensitive. 'Ben' is not the same as 'ben' and they will be treated as separate ships.

A unit such as a ship or place must be "declared" before it can be used. For a unit a "declaration" is a sentence saying what side the unit is belongs to. For example, "II Corps is a German unit". When the scenario is visualised in 2D, the symbols representing the units will be coloured according to which side they are on. For a place a "declaration" is a sentence saying where the place is geographically located. For example, "Weymouth is located at 50.6 N 2.45 W."

Note the next section is a copy of the HTML file *ScenarioBuilder_Narratives.htm* used in the **ScenarioBuilder** installation. To maintain this document when the HTML file is edited, do the following:

- delete the following pages
- insert the HTML file by selecting menu item **Insert | File**
- select file type *All Web Pages* and navigate to the folder containing the XML file.
- select the XML file named *ScenarioBuilder_Narratives.htm*
- update the Contents page

ScenarioBuilder Narratives

[\[Contents list\]](#) [\[Short example\]](#) [\[Longer example\]](#) [\[Geography and time narrative conventions\]](#) [\[The Narrative Language\]](#) [\[Back to main page\]](#)

ScenarioBuilder narratives are split into three parts

- **Scenario Information**
 - when the scenario starts and ends
 - when the simulated part of the scenario starts and ends (optional)
 - the times and heights of low and high tide (optional)
- **Cast List**
 - list of locations giving their positions in terms of latitude and longitude
 - list of ships mentioned in the scenario
 - list of trains mentioned in the scenario
- **Actions**
 - changing weather conditions
 - changing sea state
 - initial ship positions at or near locations, berthed in harbour or anchored
 - ships entering and leaving ports
 - ships moving, changing speed and direction

Short example

Here is a short narrative following that structure

```
[===Scenario information=====
===]
Starts at 06:00 on 13 December 1939.
Ends at 06:30.
[===Cast list=====
===]
A is located at 53.444N, 3.0284W.
Dartmouth 1 is a Commonwealth ship.
[===Actions=====
=]
Wind is heading 270 at speed 9 knots.
At 06:00 Dartmouth 1 is 1.0 miles north west of
location A.
At 06:25 Dartmouth 1 is 1.0 miles north east of
location A.
```

We can see that a ship is moving in this scenario.

Geography and time narrative conventions

ScenarioBuilder narratives have sets of conventions for expressing direction (north, east, south, etc) , location (latitude and longitude) and time (date and time of day). These conventions will be found below by clicking on these links:

[geographical types](#)

[time types](#)

The Narrative Language

The next sections describe in detail the narrative language:

Scenario Information Section

[Scenario scenes](#)

[Starting and ending scenarios](#)

[Starting and ending simulations](#)

[Specifying the tides](#)

[Using scenario narratives](#)

Cast List

[Specifying locations](#)

[Specifying ships](#)

[Specifying trains](#)

Actions

[Sea state](#)

[Weather](#)

[Initial ship placement](#)

[Docking of ships](#)

[Sailing of ships](#)

[Ship positions](#)

[Ship heading and speed](#)

[Initial train placement](#)

[Moving of trains](#)

Scenario Information Section

Scenario scenes

Scenarios take place in a landscape or seascape. Landscapes are created in ThingBuilder where they are referred to as ScenarioScenes. They are inserted into a scenario with a sentence of the form:

Scene is <scene name>.

e.g. **Scene is** Liverpool.

If a scenario narrative does not contain a Scene sentence then ScenarioBuilder assumes that a default seascape is to be used. It determines the extent of the seascape from the locations traversed by the objects in the scenario.

When making videos it may be that items in a scene that can move (such as railway signal arms and swing bridges) do not move for the duration of the video clip. In these circumstances we can include the scene as a single, static model rather than multiple models for the moving parts. This makes the visualisation run smoother on less powerful graphics machines. This is achieved by sentences of the form:

Scene static is <scene name>.

e.g. **Scene static is** Liverpool.

[\[back to contents\]](#)

Starting and ending scenarios

These sentences must be present otherwise the Builder does not know the duration of the scenario..

Starts <time>.

e.g. **Starts** at 6:00am on 1st June 1780.

Ends <time>.

e.g. **Ends** at 7:00am on 1st June 1780.

[\[back to contents\]](#)

Starting and ending simulations

A simulation can be made of a part of a scenario as well as the whole scenario. The scenario sets up the whole scene and we capture the action in the simulation when it gets interesting. If these simulation starting and ending sentences are left out then the scenario start and end times are used.

Simulation starts <time>.

e.g. **Simulation starts** at 6:00am on 1st June 1780.

Simulation ends <time>.

e.g. **Simulation ends** at 7:00am on 1st June 1780.

These sentences are optional. If omitted then the simulation is for the full duration of the scenario.

[\[back to contents\]](#)

Specifying the tides

In scenarios involving ports and harbours, realism is enhanced by simulating the rise and fall of the tide. We can specify the state of the tide at every moment during the simulation by specifying the time and height of high and low tides.

High tide is <number> **metres** <time>.

e.g. **High tide is** 12 **metres** at 6:00 am.

Low tide is <number> **metres** <time>.

e.g. **Low tide is** 4 **metres** at 2:00 pm.

These sentences are optional. If one or both is omitted then the tide is assumed to be fixed at 0 metres.

[\[back to contents\]](#)

Using scenario narratives

We can reuse scenarios in other scenarios. All the scenes and actions in the reused scenario are present in this scenario. This sentence should go at the beginning of the new narrative.

Use scenario <narrative name>.

e.g. **Use scenario** Liverpool Scene 1.

[\[back to contents\]](#)

Cast list

This section lists the locations and ships mentioned in the narrative.

Specifying Locations

Locations are specified by giving their name and latitude and longitude:

`<name> is located at <lat-long>.`

e.g. Portland Bill **is located at** 50.514N 2.456 W.

e.g. A **is located at** 49.5N 2.4W.

[\[back to contents\]](#)

Specifying Ships

Ships are specified by giving their name and affiliation (e.g. British, White Star Line etc).

`<name> is (a|an) <affiliation> ship.`

e.g. HMS Hood **is a** British **ship.**

e.g. SS City of Benares **is an** Ellerman Lines **ship.**

[\[back to contents\]](#)

Manually controlling a ship

Ships can be manually controlled by using the sentence

`<ship name> is manually controlled.`

The ship should be given an initial position. For example ininitally berthed or located at a point.

e.g.

SS Flower **is a** British **ship.**

It is manually controlled.

At 06:00 it is atX.

When the scenario runs, the ship can be controlled by using the throttle and rudder controls.

Specifying trains

Trains are specified by giving a list of the locomotives, coaches and wagons forming the train.

Railway: Train <name> **consists of** <list of names>.

e.g. **Railway: Train** train101 **consists of** Aspinall 0-6-0, 15 x wagon444.

This example has one locomotive and 15 wagons.

e.g. **Railway: Train** train101 **consists of** Aspinall 0-6-0, Rapid Shunter, 15 x wagon444, Banking engine.

This example has two locomotives leading (double header) and one locomotive at the back pushing (banking engine).

<list of names> is a comma separated list. Multiples of a given name can be specified by multiplying the name by the number required e.g. 15 x wagon444.

[\[back to contents\]](#)

Actions

This section describes the actions performed by the members of the cast. It also describes changes to the weather and sea conditions.

Sea state

We can specify the sea state giving the wave trough to peak height, the time period for a complete wave to pass a particular point and the direction from which the wave is coming. This is partially implemented for ships pitching. However, rolling and actual sea surface movement has not yet been implemented.

Sea state is <number> **metres high and period is** <number> **seconds from heading** <number>

e.g. **Sea state is 1.5 metres high and period is 8 seconds from heading 270 at 07:00 am.**

[\[back to contents\]](#)

Weather

We can specify the wind strength and direction it is coming from.

Wind is heading <number> at speed <number> knots

e.g. **Wind is heading 090 at speed 10 knots at 7:15 am.**

We can specify the level of visibility.

Visibility is <number> (miles | nautical miles | metres | yards)

e.g. **Visibility is 500 yards at 7:15 am.**

We can specify that it has started raining

It is raining.

Rain starts to fall at the specified time. Currently this affects the adhesion of locomotive wheels on the rails.

[\[back to contents\]](#)

Initial ship placement

We can specify where a ship is berthed or anchored at the beginning of a scenario.

<ship name> is initially berthed in <harbour name> at berth <berth name>

e.g. **Buttercup is initially berthed in Liverpool Docks at berth139.**

<ship name> is initially anchored at <location name>.

e.g. **A is located at 34N 45W. Buttercup is initially anchored at A.**

[\[back to contents\]](#)

Docking of ships

In the real world, docking can be assisted by, for example, tug boats. Currently, only unassisted docking has been implemented.

<ship name> docks (assisted | unassisted) in <harbour name> at berth <berth name>

e.g. **Buttercup docks unassisted in Liverpool Docks at berth 139 at 11:45 am.**

Each berth has a default direction in which a ship is berthed. This can be overridden with the following sentence type./ Note that the directions are only approximate. Builder will line up the ship precisely using the sentence information as a clue as to which way round to face.

<ship name> **docks (assisted | unassisted) in** <harbour name> **facing** <direction> **at berth** <berth name>.
e.g. *Buttercup docks unassisted in Liverpool Docks facing south east at berth 139 at 11:45 am.*

[\[back to contents\]](#)

Sailing of ships

We can specify a ship leaving a berth. The narrative must have already identified where the ship is berthed. If it hasn't then ScenarioBuilder will report an error. It will need somewhere to sail to so it should be followed up with a sentence indicating where the ship is at a later time.

<ship name> **sails (assisted | unassisted)**

e.g. *Buttercup sails unassisted at 11:00 am.*

[\[back to contents\]](#)

Ship positions

We can specify where a ship is relative to locations and other ships in the cast list.

<ship name> **(is at | is in | are at | are in | stops at | arrives at | leaves)** <location name>.
e.g. *HMS Hood is at A at 05:55 am..*

<ship name> **(starts to move | starts to retreat | moves | retreats)** .
e.g. *HMS Hood arrives at A at 8:00am. It starts to move at 11:00am. It is at B at 11:30am.*

This example includes time information to make the point. The middle sentence ensures the ship does not start moving until 11:00am. Without the middle sentence it would begin moving at 8:00am.

<ship name> **(is | stops | arrives)** <number> **miles** <direction> **of location** <location name>.
e.g. *HMS Hood stops 5 miles south of location Portland Bill.*

<ship name> **(is | stops | arrives)** <number> **miles**
<direction> **of** <ship name>.
e.g. *HMS Hood* **is 20 miles SE of Bismarck.**

<ship name> **(is | stops | arrives)** <number> **miles from**
location <location name> **on bearing** <number>
e.g. *Bismarck* **stops 100 miles from location Brest on bearing 250**

<ship name> **(is | stops | arrives)** <number> **miles from**
<ship name> **on bearing** <number>
e.g. *Bismarck* **stops 30 miles from Rodney on bearing 080**

[\[back to contents\]](#)

Ship heading and speed

<ship name> **(is on|turns to) heading** <number> **at speed**
<number> **knots**

e.g. *HMS Hood* **is on heading 050 at speed 25 knots** at 05:55 am.

<ship name> **turns (to port|port|to starboard|starboard)**
<number>

e.g. *HMS Hood* **turns to port 030** at 05:58 am.

<ship name> **(increases|decreases) speed (to|by)** <number>
knots

e.g. *HMS Hood* **increases speed by 5 knots** at 07:00 am.

e.g. *HMS Hood* **increases speed to 28 knots** at 07:05 am..

<ship name> **turns to heading** <number>.,

e.g. *HMS Hood* **turns to heading 270** at 06:05 am.

<ship name> **(sets speed to|has speed|is at speed)**
<number> **knots**

e.g. *HMS Hood* **sets speed to 31 knots** at 06:07 am.

[\[back to contents\]](#)

Initial train placement

Railway: <train name> **is initially parked at** <mark name> **back towards** <mark name>

The head of the train is placed at the first mark and the train body is placed back towards the second mark.

Marks are embedded in the ScenarioScene that forms the backdrop to the scenario.

e.g. **Railway:** train101 **is initially parked at** Mark0 **back towards** Mark2.

[\[back to contents\]](#)

Moving of trains

Railway: <train name> **moves to** <mark name> **facing** <direction> **at speed** <number> **mph.**

e.g. **Railway:** train101 **moves to** Mark6 **facing** south **at speed** 5 **mph** at 6:14 am.

[\[back to contents\]](#)

Manually controlled trains

In a scenario a train can be specified to be manually controlled.

Railway: <train name> **is manually controlled.**

e.g. **Railway:** train101 **is manually controlled.**

The train can only move along a specified train route. A signal will hold the train until it is cleared at the specified time. (In version 0.22 the signal has not yet been implemented).

Railway: <train name> **is cleared for route** <train route name>.

e.g. **Railway:** train101 **is cleared for route** R001 at 06:01.

[\[back to contents\]](#)

Longer example

```
[===Scenario information=====]  
Starts at 06:00 on 13 December 1939.  
Scene is Liverpool.  
Ends at 06:30.  
Simulation starts at 6:01.
```

Simulation ends at 6:30.
Low tide is 4 metres at 2:00 pm.
High tide is 12 metre at 6:00 am.
[===Cast list=====]
A is located at 53.444N, 3.0234W.
B is located at 53.444N, 3.0264W.
C is located at 53.444N, 3.0294W.
D is located at 53.444N, 3.0314W.
E is located at 53.444N, 3.0334W.
F is located at 53.444N, 3.0354W.
Schooner 0 is a Commonwealth unit.
Schooner 1 is a Commonwealth unit.
Schooner 2 is a Commonwealth unit.
Schooner 3 is a Commonwealth unit.
Schooner 4 is a Commonwealth unit.
Side Paddle 1 is a Commonwealth ship.
Side Paddle 2 is a Commonwealth ship.
Side Paddle 3 is a Commonwealth ship.
Side Paddle 4 is a Commonwealth ship.
[===Actions=====]
Wind is heading 090 at speed 9 knots.
Sea state is 0.5 metres high and period is 8 seconds from heading 270.
Schooner 0 is initially berthed in Liverpool Docks facing north at berth Berth0.
Schooner 1 is initially berthed in Liverpool Docks facing north at berth Berth1.
Schooner 2 is initially berthed in Liverpool Docks facing north at berth Berth2.
Schooner 3 is initially berthed in Liverpool Docks facing north at berth Berth7.
Schooner 4 is initially berthed in Liverpool Docks facing north at berth Berth8.
At 06:00 Side Paddle 1 is 0.1 miles north west of location A.
Side Paddle 2 is initially berthed in Liverpool Docks facing west at berth Berth3.
Side Paddle 3 is initially berthed in Liverpool Docks facing west at berth Berth9.
Side Paddle 4 is initially berthed in Liverpool Docks facing west at berth Berth10.
Schooner 0 sails unassisted at 06:02.
At 06:25 Schooner 0 arrives at A.
Schooner 1 sails unassisted at 06:01.
At 06:24 Schooner 1 arrives at B.
Schooner 2 sails unassisted at 06:00.
At 06:23 Schooner 2 arrives at C.
Side Paddle 2 sails unassisted at 06:01.
At 06:27 Side Paddle 2 arrives at D.
Side Paddle 3 sails unassisted at 06:01.
At 06:27 Side Paddle 3 arrives at E.
Side Paddle 4 sails unassisted at 06:01.
At 06:27 Side Paddle 4 arrives at F.
At 06:25 Side Paddle 1 docks unassisted in Liverpool Docks at berth Berth6.
At 06:00 visibility is 3000 metres.
At 06:01 visibility is 2000 metres.
At 06:10 visibility is 230 metres.

[\[back to contents\]](#)

Geographical types

We specify direction and position using the following text conventions:

<direction> = (north | west | east | south | north west | north east | south west | south east | N | S | E | W | NE | NW | SE | SW)

<lat-long> = <real>(N|S) <real>(E|W)

e.g. 7.5N 6.4W

[\[back\]](#)

Time types

We specify the time of an event using the following conventions:

<year> = four digit Gregorian year

e.g. 2006.

<month> = (Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |

January | February | March | April | May | June | July | August | September | October |
November | December)

<time of day> = [1..12]:[00 .. 59](am|pm)

e.g. 9:45am.

<time of day> = [0..24]:[00 .. 59]

e.g. 14:45

<day of month> = [1..31]

<day of month> = [1..31](st|nd|rd|th)

<date> = <day of month> <month> <year>

<date> = <day of month> <month>

Time phrases used in sentences

The parser can recognise the following phrases identifying time.

<time> = at <time of day> on <date>

e.g. at 9:45am on 1st June 1780

<time> = at <time of day>

e.g. at 9:45am

<time> = on <date>

e.g. on 1st June 1780

These might be used as in the following examples:

The regiment arrived at Rouen **at 9:45am on 1st June 1780**.

At 9:45am on 1st June 1780 the regiment arrived at Rouen.

At 6:00am on 1st June 1780 HMS Bounty sailed.

[\[back\]](#)