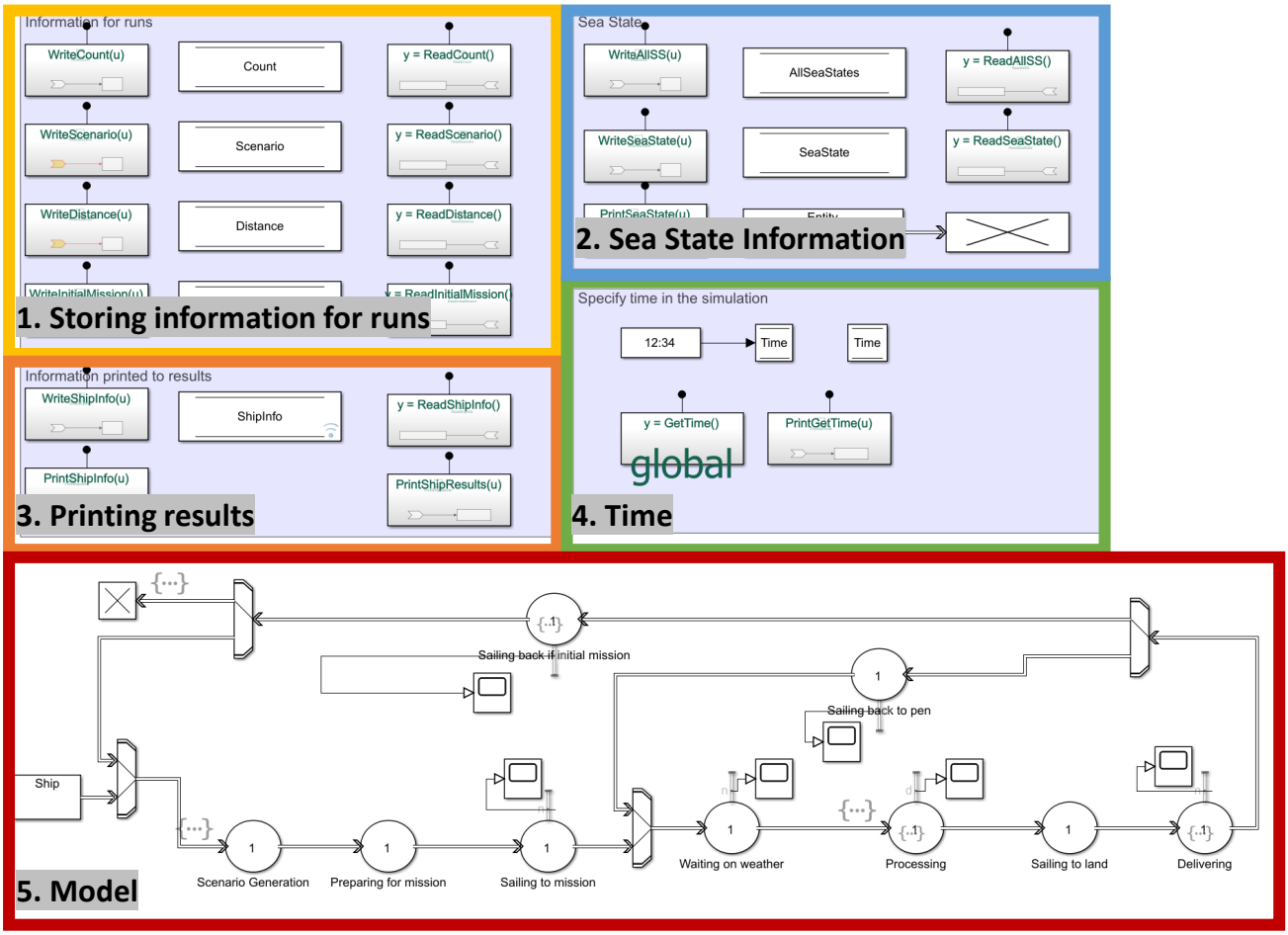
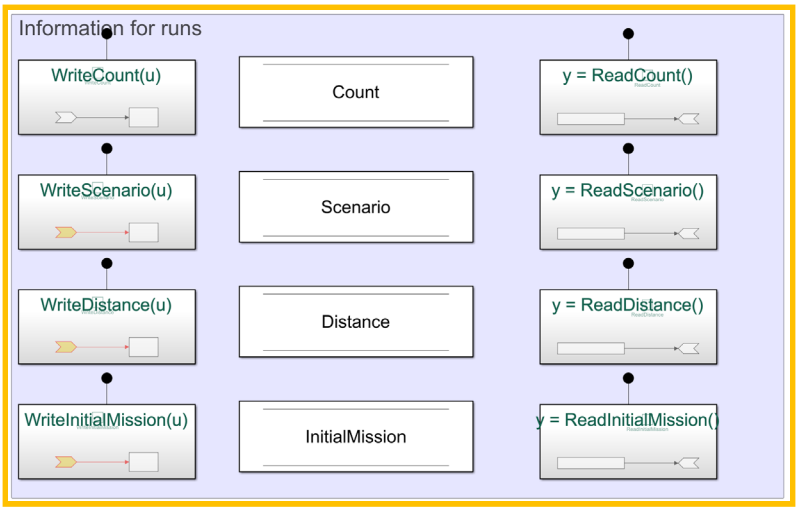
Simulation Model overview



A brief explenation of the simulation will be provided in this document.

Above, the whole simulink model is depicted. Each seaction will be commented on.

1. Storing information for runs

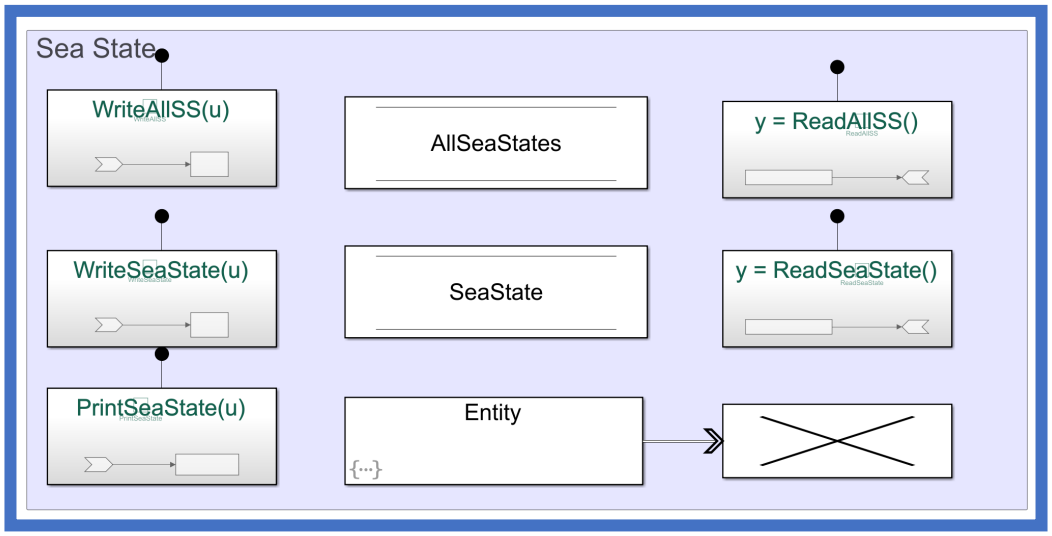


This part of the simulation consists of data memory stores with corresponding functions to write and read information to them.

There is one store for each of the parameters of the mission and in addition a store that stores the count of what mission the ship is currently on.

I

2. Sea state information

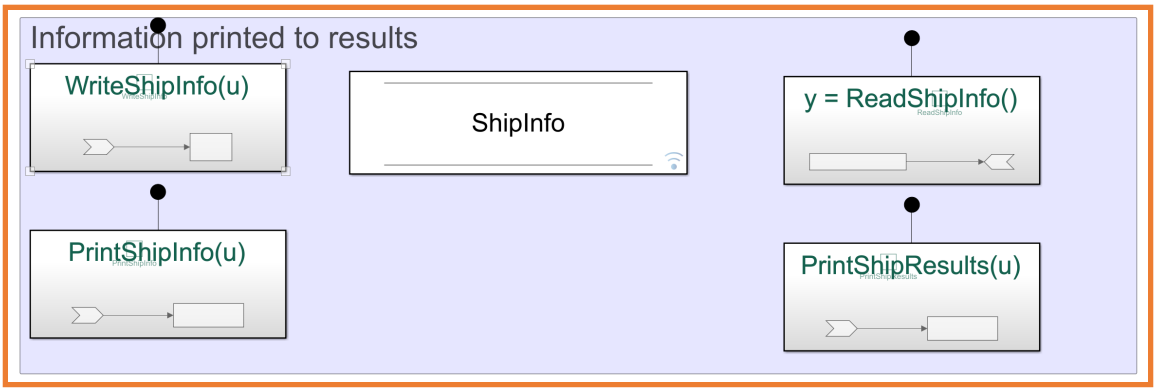


This section of the model determines the sea states of the simulation. There is an entity generator that for every whole time instant runs a code. The first time it runs, all the sea states are loaded from an excel-file (The sea states where processed in MatLAB by first “cleaning” the imported nc file and then using a Marcov chain).

All the seastates are saved to «AllSeaStates». The code in the entity generator reads the sea states and writes the sea state at the current time instance to the store called «SeaState»

«AllSeaStates» is needed to access more than just the current sea state at the same time. This is needed for the servers where a for loop is used to calculate the sailing time with varying sailing speed due to seastates, as well as calculating the WoW.

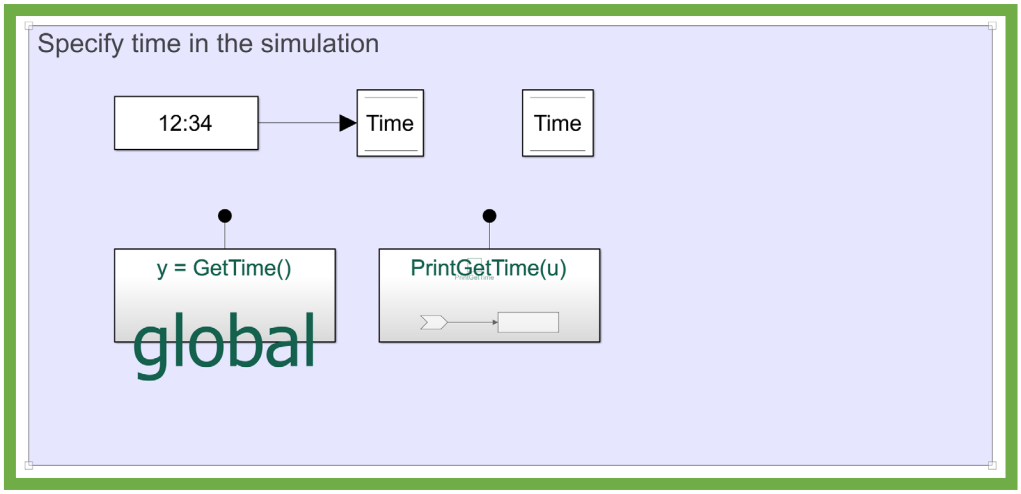
3. Printing results



This section of the code stores information that follows the ship, such as time spent sailing, time waited for weather, etc. This info is then printed to the workspace of MatLAB both as a vector that is printed each time it is changed, and as a vector that is printed at the end of each run.

This data is the resuts of the simulation.

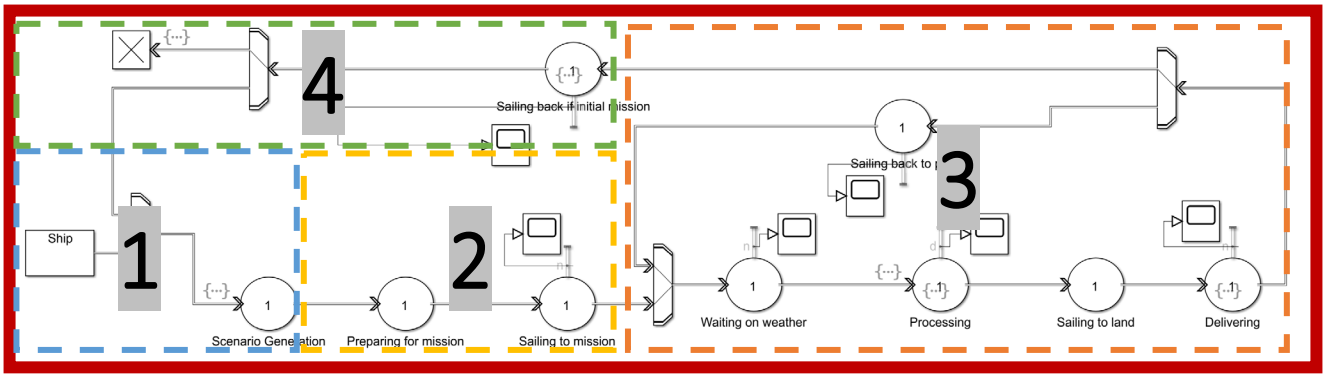
4. Time



This section of the model writes the time of the simulation (from a digital clock) to a data store memory. So that the current time can be used in other parts in the simulation (for example as when the sea state is determined, as mentioned earlier)

The section includes a read function to read the current time, and also a print function to be able to print the current time to the workspace if needed.

4. Time



Part 1:
The vessel is generated with input from MatLAB
The vessel reaches the first server where the scenario is read from excel. Updates for each loop.

Part 2:
The vessel aborts its current mission if it has one. The time used to determine the time spent aborting is based on the scenario and the designs ability to abort.

The next server models the vessel sailing to the mission. The distance is determined by the scenario (and is for simplicity the same for all sailing legs). The sailing speed is affected by the sea state.

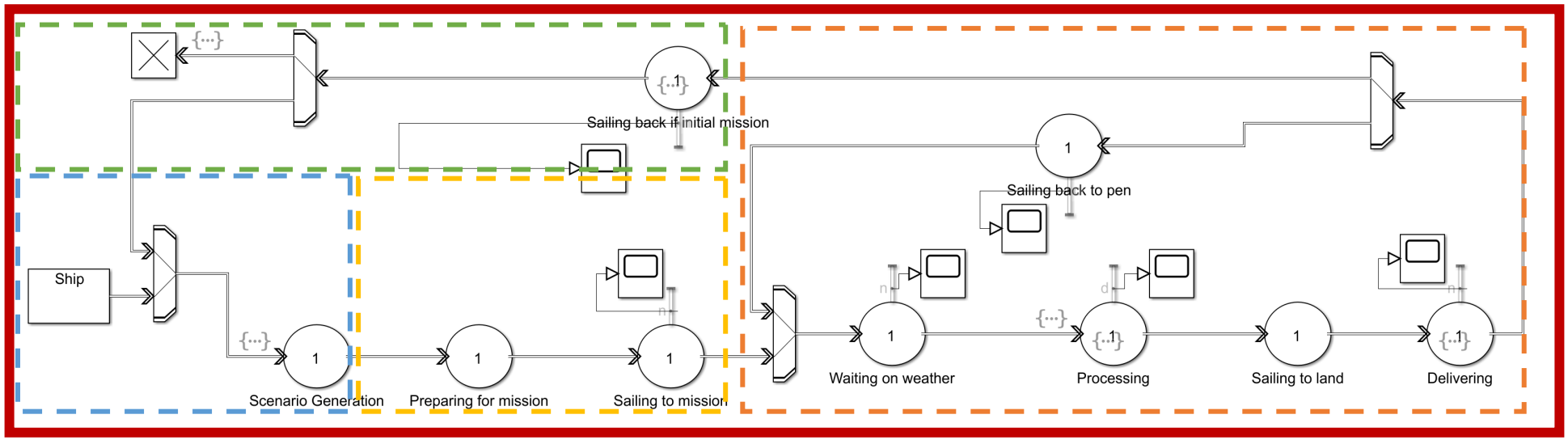
Part 3:
The vessel reaches the “mission loop” where it enters a loop of processing and sailing. The vessel first reaches the waiting for weather server. Here it waits for a weather window long enough to carry out the whole mission.

The ship processes as much salmon as possible, either the vessel capacity or what is left in the fish pen (vessel 1 has an infinite capacity due to ship to ship transfer). The processing speed is reduced when the sea states increases.

The vessel (except design 1) sails to land. Then delivers the salmon. Sails back to the fish pen if there is more fish left, exits the loop if not.

Part 4:
The vessel sails back to the initial mission if it had one.

The vessel exits the whole simulation through the entity terminator when all the predetermined missions are carried out.


Sailing back if initial mission
Sailing back to pen
Ship
Scenario Generation
Preparing for mission
Sailing to mission
Waiting on weather
Processing
Sailing to land
Delivering

How to run the simulation

To run the simulation:

1) Run “RunSim.m”

This script also prints the results to the folder.

The plots in the excel file “Plots” uses data from the resultfiles. The origin file might need updating due to it being in a zipped folder. This is done in the data tab of excel.

Optional:

The weather file that the simulation uses is already saved in the folder. However, if you want to reproduce them or create a new set with a different seed you must:

1) Run “WeatherImport.m” (to import the nc data and clean them)
2) Run “MarcovChain.m” (to create a longer set of data that starts at different sea states)