Maestro first initialises all permanent parameters (which do not change during runtime)

A for loop then begins in order to generate multiple runs (if specified) from the start to end tic.

Maestro then initialises all variables which need to be reset between runs and starts Stella.

If it finds that CRAFTY has an undeleted updated.txt file it hangs until that file is deleted to prevent false data and crashes.

Maestro initialises all variables which must be reinitialised each tic.

When Stella informs Maestro through the status.txt file that data is ready Maestro loads export demand in Gigagrams for the current year (maestro tic -1).

Maestro loads internal demand in Gigagrams for the current year

The total demand data for each considered service is converted into CRAFTY units

The CRAFTY demand file is modified with the demand as well as demand for unconsidered services **which must be entered here and not in the Original CRAFTY File**.

CRAFTY is then told demand is ready and Maestro waits for the update file from CRAFTY which comes in at the start of every next year in CRAFTY.

This file tells Maestro that it is safe to collect output data for the year while CRAFTY once again waits for the demand file to be updated.

This also means CRAFTY does not generate this file in the last tick of any run thus to prevent Maestro hanging if the tick is the last tick Maestro instead checks for the CRAFTY output file to be created manually.

Once Maestro finds that CRAFTY data is ready it loads the output file in.

The CRAFTY output file and region file are read simultaneously line by line to give loaded cells municipality id

This relies on cell order in output file in CRAFTY to be the same as in region file (which by default it is)

Produce data totals per state are also found

The produce is converted from CRAFTY units to Gigagrams and modified by state conversion values where necessary (for meat and dairy).

This gives service totals in Gigagrams

Maestro maintains values for soy and maize storage – produce left over from last year.

**For Soy and Maize:**

Internal demand is first met from storage.

If any internal demand remains it is met from freshly produced goods.

If some internal demand is unmet there is no penalty but nothing will be passed back to Stella which will impact demand totals next year.

If any storage is left this is used to meet demand from Stella as much as possible.

If any storage is left after this it is discarded.

If Stella demand cannot be met by storage it is met from freshly produced goods.

If Stella demand cannot be fully met at this point Stella receives less goods than it expects and this impacts next year’s total demand.

If any freshly produced goods remain they are added to storage.

**For Meat and Dairy:**

Freshly produced goods are used to meet internal demand

What remains is used to meet Stella demand

If Stella demand not fully met this impacts next year’s demand.

Any remaining produce is discarded.

Maestro modifies the ToStella file in the Stella data folder with produce which Brazil exports this year in Gigagrams.

Maestro informs Stella it is ready for next year’s data and the next tick starts.

If it is the final tick stella terminates itself and Maestro begins a new run (if Specified) and reinitialises Stella.

21,09,2018

Added functionality to deduct, for soy and maize the ammount in storage (in gigagrams) from internal demand and then from Stella demand prior to submitting to conversion to CRAFTY units

25-26., 09, 2018

Added informative error message when waiting for updated.txt to be removed at start of run.

Demand for unsimulated services now loaded from demand file at start of batch.

Changed output names and Stella zip names so that they are not overwritten between batch runs.

27,09,2018

Added a number of bufferedreader.close()statements to remove memory leak