



**To:** NYISO and Stakeholders  
**From:** Analysis Group  
**Date:** August 10, 2020  
**Re:** Net EAS Fossil Model Readme

---

This memo describes the Net EAS Fossil Model provided to NYISO and stakeholders by Analysis Group. It outlines the model logic and folder structure and provides directions on how to use the model. It also includes selected troubleshooting suggestions.

## *Net EAS Fossil Model Overview*

The Net EAS Fossil Model is an R model. The model is designed to take a series of **cases** -- a group of combustion turbine or combined-cycle units, with various fixed parameters determining characteristics such as whether the units have scrubbers or whether they run on gas and oil or only gas -- and evaluate each case in succession. Case evaluation is simply the application of the model logic to each unit in the case.

A set of cases is known as a **run**. Each case can contain up to four units, and there is no formal limit on the number of cases in a run. Each run produces a pair of output workbooks in .xlsx format summarizing revenues and other markers of unit behavior for each case; runs can also produce an intermediate dataset, depending on user specifications.

Users will primarily rely upon the following files:

- Running the Model
  - **EAS Model Control.R**, which executes the model. It is stored in the *Programs* folder.
  - **Run Specifications.xlsx**, which specifies the cases and most of the toggles that users can set for each run. It is stored in the *1. Run Inputs* subfolder within the *Runs* folder.
- Viewing Model Results
  - The **DCM Inputs** and **Diagnostic Tables** output workbooks, which are generated by each run and stored in the *Outputs* folder.
  - **Results Table Viewer.xlsx**, which is stored in the *Results Table Viewer* folder and can be used to view results from the **Diagnostic Tables** workbook in a more easy-to-read format.

## *Model Elements and Folder Structure*

The Net EAS Fossil Model is stored in four folders to cleanly separate model elements.

- *Programs*: Contains the Net EAS Fossil Model code.
  - **EAS Model Control.R** is the code's control script. This script runs the model; it does so by calling a series of functions stored in other scripts.
  - The scripts containing those functions are all stored in the folder "Functions."
  - The model has been executed on a Windows operating system on R version 3.6.1 and a Linux operating system in R version 3.6.2. The model relies on several widely-used R packages. For users who do not already have these, the model installs versions of them that

have worked in executions on the operating systems (and, where appropriate, with the R versions) mentioned above.

- The code is described in greater detail below in section “Model Code.”
- *Shared Inputs*: contains four input files that are the model’s default inputs.
  - Price and Cost Input Data.xlsx contains all price and cost input data used from the 2016 DCR Reset through the 2019 Net EAS Fossil Model Annual Update.
  - General Model Parameters.xlsx contains miscellaneous parameters used for all fossil units around fuel costs.
  - Unit Technical Parameters.xlsx contains the technology-specific parameters for each unit.
  - Output\_Files\_Readme.csv is a readme that is included with each diagnostic output workbook describing the information contained in the workbook.
- *Runs*: contains folders to each distinct run you wish to evaluate on the model. You can name your “run folder” as you please. Each run folder will contain all run-specific inputs and outputs from the model, divided into three subfolders:
  - *1. Run Inputs*: this folder contains all run-specific inputs. At a minimum, it must include a copy of the file **Run Specifications.xlsx**, which identifies the units and toggle settings for each run in a given model execution. It can also contain versions of the two default input items that include user-specified data (Price and Cost Input Data.xlsx and Unit Technical Parameters.xlsx).
    - See sheet “Legend” of **Run Specifications.xlsx** for more details.
  - *2. Intermediate Datasets*: this folder contains the intermediate dataset that users may select to export as part of their run. This dataset reflects the full hourly results of the model (unit dispatch behavior and profits) and is produced at the end of Module 2.
    - Users can choose in the **Run Specifications.xlsx** workbook whether to output intermediate datasets, and if so, whether to output them in .rds or .csv formats (or both formats). See sheet “Legend” of **Run Specifications.xlsx** for more details.
    - .rds intermediate datasets are produced as single files for each run, covering all units for all cases and tagged with a user-set run ID (see the section “Running and Updating the Net EAS Fossil Model” below). .csv files are produced separately for each unit within each case and are tagged with user-set run IDs and with case IDs set for each case by the user in the **Run Specifications.xlsx** workbook.
    - Export filenames are formatted as *[run\_ID] Model\_Outputs.rds* or *[run\_ID] Model\_Outputs [Case\_ID] [Unit Code].csv*.
      - **Example**: *example\_run Model\_Outputs ct\_1 j25.csv* means the following:
        - Run\_ID: *example\_run*
        - Dataset: *Model\_Outputs*
        - Case\_ID: *ct\_1*
        - Unit Code: *j25 (1x GE HA.02 25ppm [frame combustion turbine])*
    - *3. Outputs*: this folder contains output workbooks generated for each run. Each output workbook is tagged with the date and time of the run, as well as the run’s user-specified run\_ID. Two workbooks are generated for each run:
      - **DCM Inputs**: this workbook reports the inputs from the Net EAS Fossil Model used in the DCM model; that is, the average \$/kW-yr net EAS revenues figures for each unit in each zone modeled.
      - **Diagnostic Tables**: this workbook includes various diagnostic tables from the runs breaking down revenues, operating hours, and start counts across different time

- periods, dispatch behaviors, and types of fuel. A readme at the front of the workbook explains the contents.
- *Results in these workbooks do not reflect annual flat revenue adders calculated by NYISO for voltage-support services (combustion turbines, \$2.04/kW-year) or for voltage-support services and ancillary services (combined-cycle units, \$5.53/kW-year).*
- *Results Tables Viewer*: this folder contains a **Results Tables Viewer.xlsx** workbook, which allows users to summarize key outputs from the Net EAS Fossil Model in easy-to-read form.

## ***Model Code***

The Net EAS Fossil Model is split into 3 code modules, each containing several subsections of code. Each module corresponds to a single script containing a single or a handful of functions that executes the needed analyses. There is a single Control script (**EAS Model Control.R**) that executes the model by calling these functions in appropriate order. As noted above, the **EAS Model Control.R** script is stored in the *Programs* folder, while all other scripts are stored in the *Functions* subfolder of the *Programs* folder.

The code of the Model is as follows [with the corresponding R scripts for each module in brackets]:

1. Model Inputs: Import and Merging
  1. Economic Inputs Import [1.1 Economic Inputs Import.R]
  2. Engineering Inputs Import [1.2 Engineering Inputs Import.R]
  3. Data Merges [1.3 Data Merges.R]
2. Model Evaluation
  1. Calculation of Day-Ahead and Real-Time Hourly Potential Profits [2.1 Calculate Base DAM and RTD Profits.R]
  2. Determination of Day-Ahead Behavior and of Real-Time Buyout Decisions [2.2 Set DAM Behavior & RT Buyout Behavior.R]
  3. Determination of Real-Time Behavior [2.3 Set RTD Energy and Non-Energy Block Behavior.R]
  4. Application of EFORD Adjustment and Runtime Limits, Calculation of Final Real-Time Profits [2.4 Apply EFORD and Runtime Limits.R]
3. Model Summary and Data Export
  1. Model Result Summary [3.1 Summarize Model Results]
  2. Model Result Summary Export [executed within the control script]

There are also a series of helper functions called across these modules in the script file *Helper Functions.R*.

## ***Running or Updating the Net EAS Fossil Model***

The Net EAS Fossil Model can be updated as follows:

1. Create a new folder within *Runs* with any name. Insert in your new folder three separate subfolders: 1. *Run Inputs*, 2. *Intermediate Datasets*, and 3. *Outputs*. Place into the 1. *Run Inputs* subfolder a copy of **Run Specifications.xlsx** from a prior run folder or from the model copy you downloaded; you will use this copy for your run.
  - a. *[If you would like to use an existing run folder, ignore this step.]*

2. In your run copy of **Run Specifications.xlsx** workbook, set run-level and case-level specifications you would like for this run on sheets “All\_Case\_Specs” and “Individual\_Case\_Specs.”
3. Run the R control code, **EAS Model Control.R**.
  - a. To run, first:
    - i. Update `runfolder_name` on line 22 of the code to correspond to the name of your run folder. **Place the name in double quotation marks.**

*Example: say you are a Windows user and you have saved your copy of the model at the location “C:\Users\rsmith\Desktop\DCR\_Model” and your run folder is at the location “C:\Users\rsmith\Desktop\DCR\_Model\Runs\Sample\_Run.” You should update line 22 as follows:*

```
runfolder_name <- "Sample_Run"
```

- ii. Set `run_ID` on line 25 of the code to an ID you would like to include in every output file’s filename. This is to help you identify files for different runs. It can be anything, but if you are exporting these files to a directory in a Windows operating system, **we recommend keeping it to around 10-15 characters, to avoid breaking Windows’ filepath character limits. Place the name in double quotation marks.**
- iii. Update `path` on line 89 of the code to reflect the location where the model directories have been saved on your system (i.e. the *Programs*, *Runs*, and *Shared Inputs* folders). **Place the path in double quotation marks and use double backslashes for each single backslash (see bullet iv below).**

*Example: say you are a Windows user and you have saved the model at “C:\Users\rsmith\Desktop\DCR\_Model”. You should update line 89 as follows:*

```
path <- "C:\\Users\\rsmith\\Desktop\\DCR_Model"
```

- iv. Make sure that forward slashes and backslashes are used properly depending on whether you are executing the code on a Windows-based or Linux-based operating system. (See lines 86-88 for instructions.)
  1. **By default, these are set to work on Windows-based operating systems.**
- v. Update the variable `case_vec` on line 120 of the code to contain the run numbers of all runs you would like to evaluate.
  1. See lines 121-132 for instructions on how to set `case_vec`. Case numbers reflect the case numbers in the “Individual\_Case\_Specs” sheet of **Run Specifications.xlsx**.
  2. **By default, it is set to evaluate all cases.**
- b. Then, place the cursor inside the script and hit Ctrl + Alt + R to run the whole script.

## *Viewing Results from the Net EAS Fossil Model*

The **DCM Inputs** and **Diagnostic Tables** output workbooks generated by the Net EAS Fossil Model contain summary tables describing the results of the model. You can paste them into your **Results Tables Viewer.xlsx** workbook to view certain key results in user-friendly form.

- Information viewable in the **Results Tables Viewer.xlsx** workbook includes:
  - Average annual revenues, operating hours, and start counts over the three-year period, including VSS and ancillary service adders;
  - Total energy revenues and operating hours by fuel type for each modeled year; and
  - Total operating hours and revenues for each year by day-ahead commitment and real-time dispatch decision, including VSS and ancillary service adders.
- Total revenues in the Results Table Viewer.xlsx workbook will reflect VSS and ancillary service adders.
- See the sheet “Instructions” in **Results Tables Viewer.xlsx** workbook for more information.

This workbook is included to help users highlight key results from the model, but note that the **Diagnostic Tables** workbook contains additional diagnostics beyond what is displayed in **Results Tables Viewer.xlsx**.

## Selected Troubleshooting Recommendations

Error Message	What's Happening?	What Do I Do?
In <code>open.connection(path, "wb") :</code> cannot open file [...] no such file or directory	The model is trying to export a file (e.g. an output or an intermediate dataset file), but the filepath is too long. This error should only happen on Windows-based operating systems.	-- Make sure your <code>run_ID</code> and <code>case_IDs</code> are 10-15 characters maximum.  -- Check the length of the filepath for your run folder. If it is more than 120 characters, move the entire model folder to a location in your drive with a shorter filepath. An example would be your Desktop or the Documents folder on your C drive.
Unexpected input in " <code>path &lt;- [...]</code> "	The model is not able to assign the path variable. These are user-inputted models that point it to the directories where your model is saved.	-- Make sure that you have placed double quotation marks around the filepaths.
<code>`path` does not exist</code>	The model is trying to call a file but is not able to find it.	-- Confirm that you have inputted <code>runfolder_name</code> and <code>path</code> accurately. <code>runfolder_name</code> should be a folder name of your specific run folder with the <i>Runs</i> folder. <code>path</code> should be the filepath of the folder for the entire model (i.e. the folder containing <i>Programs</i> , <i>Runs</i> , and <i>Shared Inputs</i> ).  -- Confirm that your <code>Price_Cost_Inputs_Filename</code> and <code>Technical_Inputs_Filename</code> in your <b>Run Specifications.xlsx</b> refer to actual files located in your <i>1. Run Inputs</i> folder, and that they include the .xlsx filetype suffix (MyInputs.xlsx, not just MyInputs).

Error: std::bad_alloc	Your current R session has run out of memory	Close out of your session, restart R, and try executing the run again.
Error: Sheet 'tech_params_NA' not found	A case in your <b>Run Specifications.xlsx</b> file lists more units under Number_of_Units than you are actually evaluating.	Make sure the number of units you specify in each case equals the number of units in that case.
Error in 5:eval(5 + case_to_setup\$Number_of_Units - 1) : argument of length 0	At least one of the case numbers specified in your case_vec does not correspond in any case number for your cases in your <b>Run Specifications.xlsx</b> file.	Make sure that all case numbers selected in case_vec correspond to case numbers in your <b>Run Specifications.xlsx</b> file.