**Adjusting Phase Timing on a Formatted Demand File**

The phase shifter tool is used to adjust the phase timing information of a formatted demand file. If updates to the timing of the phases for different scenarios are made after the demand file has already been created, this tool can be used to adjust the data without having to update the original inputs.

**Inputs**

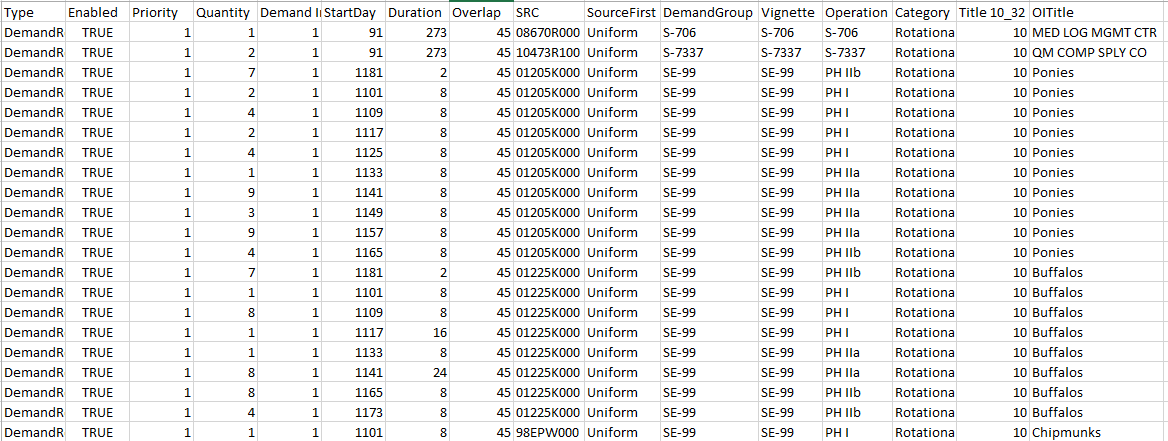
This tool requires two inputs: A formatted demand file, and a file describing the changes to be made to each phase that is to be updated.

**Demand File**

The Formatted Demand file is a MARATHON demand file. This will usually be the output from the BuildDemand script. The Formatted Demand file should contain the following columns headers (in any order):

Type, Enabled, Priority, Quantity, Demand, Index, StartDay, Duration, Overlap, SRC, SourceFirst, DemandGroup, Vignette, Operation, Category, Title 10\_32, and OITitle.

Example of formatted Demand File:



The Demand file should already be formatted, so the Scenario phases should already be synchronized to match the map and the final phase should have been extended if needed to span the full duration.

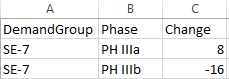
The Demand File should be formatted as a tab delimited (.txt) file if it is not already.

**Phase Change File**

The Phase-Change file describes how the current phases should be adjusted. This file is user created and should include the following columns (in any order):

|  |  |  |
| --- | --- | --- |
| DemandGroup | Phase | Change |

Example of a change file:



The Phase column should match the data from the Vignette column in the Demand File. The DemandGroup is the Scenario that is being changed and should also match what is listed in the Demand File.

The Change column takes any integer value. This column indicated the number of days the phase should be adjusted by.

Shifts are required to be a multiple of 8 days. If the change is not divisible by 8, an error screen will pop up telling the user to adjust the change to a multiple of 8.

The shifts cannot result in phases being completely overlapped or underlapped. If this occurs, an error screen will pop up telling the user the change will result in an over/underlap of the specified phase(s), and to adjust the change quantity to a decreased value.

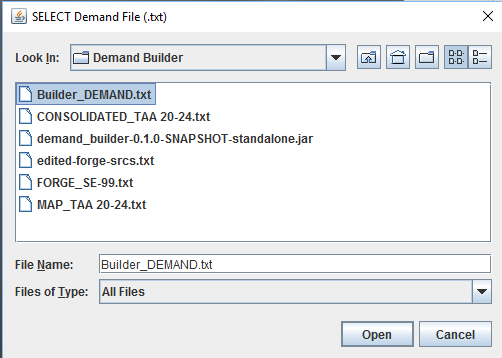
The phase shifter always operates on the indicated phases and the phase to the immediate right. Any phase shift will also effect all other phases when a gap or overlap is created. The sum of all duration will remain constant unless the final phase is extended (An increase in duration of one phase guarantees a decrease in the duration of the next phase else unless the final phase is being shifted).

The change is a change in the DURATION of a phase, however the start days can also be adjusted by shifting the previous phase.

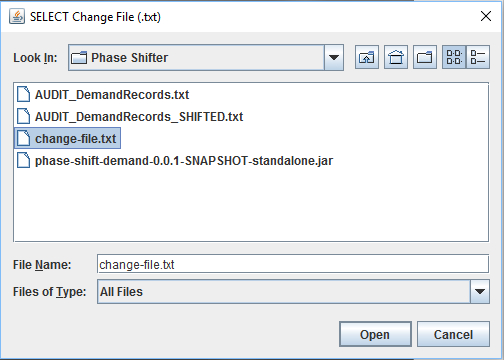
**Running Phase Shifter**

Once the change file has been created and the demand file has been formatted to tab delimited (if not already), the script can be ran by opening the jar file.

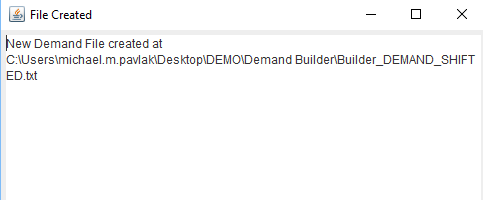


The first window will ask the user to select the Demand File. Use the file select menu to locate the demand file to be shifted and click open. 

The next window will ask the user to select the Change File (that the user created/edited). Use the file select menu to locate the change file and click open.

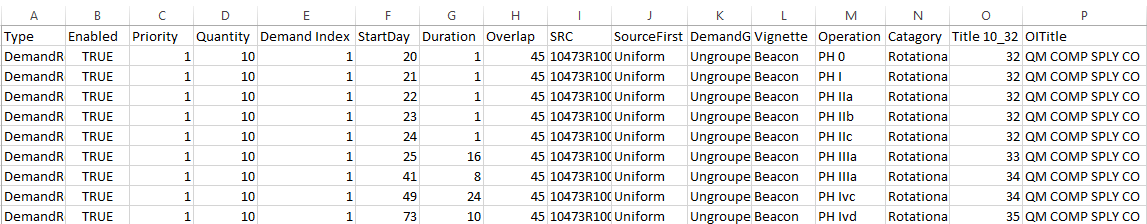


The output will be in the same directory as the input demand file with the same filename with “\_SHIFTED” appended to the end. The output will be a tab delimited text file of the input demand file with the changes from the change file. A window will appear confirming the file has been created.



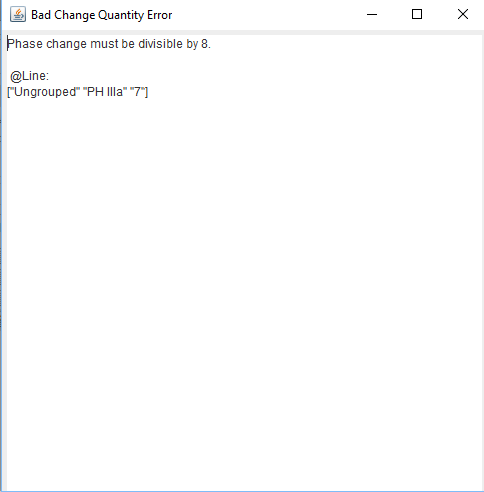
The shifted output can add or remove rows. In each row, the only data that will ever change will be the StartDay, Duration, and/or Operation (Phase). The output fill not simplify any rows that can now be combined into a single row, however, all the same data will still be available, just in a more broken out format.

Example of output file (should look the same as input with only a few rows changed and/or added/removed:

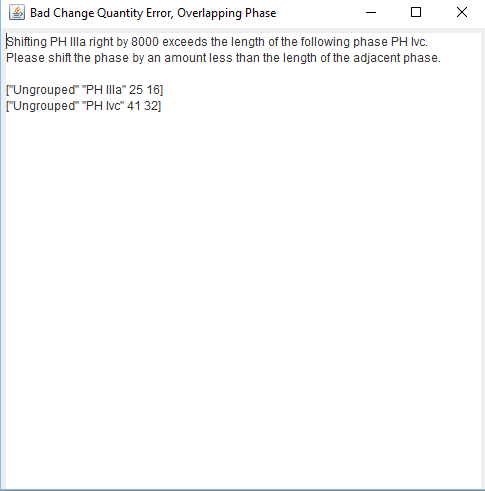


If the output file was not created, the change file needs to be adjusted based on the error message that appeared at runtime.

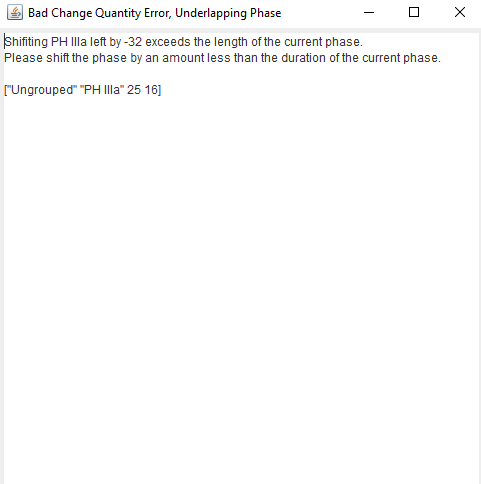
Example - Phase not divisible by 8:



Example – Phase overlap:



Example – Phase Underlap:



**Algorithm Details and Formulas**

The phase shifter always operates on the indicated phases and the phase to the immediate right. Any phase shift will also effect all other phases when a gap or overlap is created. The sum of all duration will remain constant unless the final phase is extended (An increase in duration of one phase guarantees a decrease in the duration somewhere else unless the final phase is being shifted).

The change is a change in the DURATION of a phase, however the start days can also be adjusted by shifting the phase to either the right or left.

The following method is used to update a phase:

Shift Phase i by N.

Phase i : start A, duration B

Phase i+1: start C, duration D (where C = A+B)

Phase i -> start A, duration B+N

Phase i+1 -> start A+B+N, duration D-N

It is possible that the shift will result in Phase i+1 having a negative duration or being completely overlapped by phase i. Negative duration phases and overlapped phases are handled by the program and do not need to be fixed by the user (can be handled logically, but throws error, error can be removed if this is the wanted behavior). The output will be the expected output.

The rule that the start of phase i+1 is equivalent to the sum of the start and duration of phase i is enforced after each phase shift. If there are multiple phase shifts, the first shift will occur, then each phase will be shifted by zero before moving on to the next real phase shift.

If the final phase is shifted to the left, it is possible that some records will have no value for their phase (Undefined phase after the final phase and before the initial phase). It is not possible to adjust the start date of the first phase. Other changes to either the duration or start day of all other phase will result in the expected output.

***Formulas for phase shifts:***

**Increase the duration of phase B:**

Phase B: change = +N (Phase C decreased duration)

Phase A: change = -N (Phase A decreased duration)

**Decrease the duration of phase B:**

Phase B: change = -N (Phase C increased duration)

Phase A: change = +N (Phase A increased duration)

**Add N to the StartDay of phase B (and keep the current duration):**

Phase A: change = +N, Phase B: change = +N (side effects applied to phase C)

**Decrease the StartDay of phase B by N (and keep the current duration):**

Phase A: change = -N, Phase B: change = -N (side effects applied to phase C)

**Remove phase B:**

Phase A: change = +N for N <= duration of phase B (increased phase A duration)

Phase C: change = -N for |N| <= duration of phase B (increased phase C duration)

**Remove phases B and C:**

Phase A: change = +N for N <= duration of phase B + duration of phase C (increase phase A duration)

Phase D: change = -N for |N| <= duration of phase B + duration of phase C (increase phase D duration)