## ERTAC Change Log – v3.0

No chances in operations or the algorithm occurred in ERTAC v3.0, this version is simply a port from compatibility with Python 2.7.x to Python 3.x

Known Bugs

1. **Disappearing Generation Bug**In situations when many units are curtailed a bug can occur where not all of the excess generation pool is allocated. The code calculates whether any GDUs are needed to allocate the excess generation pool by looking at the capacity of each unit minus the generation that has been assigned excluding units that had been flagged as having reached their annual capacity or operating hour cap by that hour or were retired in that hour. Later in the algorithm when excess generation is allocated units that reach their annual capacity or operating hour cap in a future hour also get excluded from consideration. If that difference is too large not all of the excess generation pool will be assigned and the annual growth rate will not be honored.
2. **Lacking Bug**Similar to the disappearing bug, the calculation for the “lacking” column in demand\_generation\_deficit also does not consider that reach their annual capacity or operating hour cap in a future hour and in cases with large amounts of curtailment will undercount the generation that is lacking in a particular hour.
3. **Decimal Places in Division**Python 3.x calculates division out to more decimal places than Python 2.x. Sometimes this results in generation and heat input being calculated that adds or subtracts generation that is of no significance (between e-10 and e-14, typically). However, these minor results can result in larger discrepancies in the calculations of the number of hours operating (especially for units higher the hierarchy). These discrepancies can also result in heat input being replaced in HIZG hours, since generation in those hours is moving from 0 to just slightly above 0, thus having the potential for substantial heat input being removed in certain hours.
4. **Quotations in ff10s**If quotations are included in the comment field of the ertac\_for\_smoke ertac\_pusp\_info.csv file it will cause SMOKE to not function when it is run.
5. **Double counting for region switches**When a unit switches regions, since this was not an expected behavior, ERTAC for SMOKE and the ERTAC post process currently adds two entries, one for each region. A fix to this will be to loosen the restriction on region switching in a future iteration of the code.
6. **ff10 hourly delay**This is more of a bug in SMOKE itself. SMOKE offsets ff10 hourly values by an hour during daylight savings time. Until this is corrected ff10 hourly results from ERTAC to SMOKE need to be adjusted by an hour to correct the adjustment that occurs in SMOKE.
7. **Non-EGU Data in ertac\_noncamd\_hourly.csv**The code removes hourly data from the hourly CAMD file if it is associated with a non-EGU, however if hourly data is included in the ertac\_noncamd\_hourly.csv it will not be removed causing errors. The fix is simple, but we chose not to include since the switch from Python 2.7.x to Python 3.x was so challenging. To fix this error replace lines 1525 to 1532 with the following code:  
     
    print("There are", len(oris\_unit\_list),  
    "units marked in the UAF as Non-EGU to be removed from the CAMD/ERTAC nonCAMD hourly data.", file=logfile)  
     
    rows\_affected = 0  
    for (plant, unit) in oris\_unit\_list:  
    rows\_affected += conn.execute("""DELETE FROM camd\_hourly\_base  
    WHERE orispl\_code = ? AND unitid = ?""", (plant, unit)).rowcount  
    print("Removed", rows\_affected, "hourly rows from CAMD data.", file=logfile)  
     
    rows\_affected = 0  
    for (plant, unit) in oris\_unit\_list:  
    rows\_affected += conn.execute("""DELETE FROM ertac\_hourly\_noncamd   
    WHERE orispl\_code = ? AND unitid = ?""", (plant, unit)).rowcount  
    print("Removed", rows\_affected, "hourly rows from ERTAC nonCAMD data.", file=logfile)

## ERTAC Change Log – v2.2

Known Bugs

1. **Disappearing Generation Bug**In situations when many units are curtailed a bug can occur where not all of the excess generation pool is allocated. The code calculates whether any GDUs are needed to allocate the excess generation pool by looking at the capacity of each unit minus the generation that has been assigned excluding units that had been flagged as having reached their annual capacity or operating hour cap by that hour or were retired in that hour. Later in the algorithm when excess generation is allocated units that reach their annual capacity or operating hour cap in a future hour also get excluded from consideration. If that difference is too large not all of the excess generation pool will be assigned and the annual growth rate will not be honored.
2. **Lacking Bug**Similar to the disappearing bug, the calculation for the “lacking” column in demand\_generation\_deficit also does not consider that reach their annual capacity or operating hour cap in a future hour and in cases with large amounts of curtailment will undercount the generation that is lacking in a particular hour.
3. **Quotations in ff10s**If quotations are included in the comment field of the ertac\_for\_smoke ertac\_pusp\_info.csv file it will cause SMOKE to not function when it is run.
4. **Double counting for region switches**When a unit switches regions, since this was not an expected behavior, ERTAC for SMOKE and the ERTAC post process currently adds two entries, one for each region. A fix to this will be to loosen the restriction on region switching in a future iteration of the code.
5. **ff10 hourly delay**This is more of a bug in SMOKE itself. SMOKE offsets ff10 hourly values by an hour during daylight savings time. Until this is corrected ff10 hourly results from ERTAC to SMOKE need to be adjusted by an hour to correct the adjustment that occurs in SMOKE.

Preprocessor

1. **Calculation of Hourly Base Year Emission Rates**.  
   CAMD hourly emissions rate data is rounded to four digits, which, especially with SO2, can result in discrepancies in calculations. A new function was added to the preprocessor so that hourly emission rates we re-calculated based on hourly heat input and emissions tons in order to retain more precision in calculations.
2. **ERTAC Fuel Unit Type Bins Input File**  
   A new file was introduced so that ERTAC fuel unit type bins can be added or edited. It includes two columns, one for the fuel/unit type bin and one for whether it should be grown. Non-grown fuel unit type bins are necessary for demand transfers to Wind or Solar for instance. The new input file is read in by the ertac\_tables.py. Given the location of this information in the code this file is stored in the same directory as the code, like the states.csv file. It should be noted that adding additional types and not using them does not create problems. Additionally, ertac\_tables.py now also reads the states.csv file in so that flexibility is available to run ERTAC in other countries or subdivide states for instance.
3. **Seasonal Control Factor Check Bug**When different seasonal control entries were included for different future years, warnings would be producednoting that several entries crossed dates, but since they were for different future years, the start and/or end dates were not in fact crossing. This has been corrected.
4. **Leap Year Utilization Fraction Bug**The utilization fraction was being calculated for a future year that is a leap year, when projected from a base year that was also a leap year was being calculated incorrectly. This is because these projections had 8784 hours in them, and 8760 was hardcoded in many parts of the code. A function was added to ertac\_lib to get the proper hours in a year for when the base and future years are both leap years and it now calculates the utilization fraction correctly.

Projection

1. **Cleaner Error Messages for Bad Base Year Data**  
   The projection code would crash with a stack trace if the calc\_hourly\_base file had no data or did not have data for every hour of the year for a region/fuel unit type bin combination. In both cases the code now aborts and explains why with a “Fatal Error” message.
2. **Hours Lacking Gload but with Heat Input**  
   Many units have hours in the CAMD hourly data file with heat input and emissions, but are not generating. It is likely that these hours are start-ups and shutdowns. The code ignored these hours resulting underestimates of emissions in projections. The code now keeps track of these hours and if after projections are complete any hour that is projected to have no generation and had no generation in the base year, but had heat input and emissions now has heat input and emissions added back in to the hourly projections. If excess generation is added into any of those hours resulting in positive generation in the projection this does NOT happen and the start-up or shutdown emissions are NOT shifted forwards or backwards. This functionality is dependent on a Boolean variable being set for each individual region and fuel-unit type bin in the ertac\_input\_variables\_v2.csv file.
3. **Hours at Max Heat Input Flag Bug Fix**In some parts of the code, only hours that a unit was above the max heat input were flagged, now hours that are above or equal to the max heat input are flagged. Other parts of the code flagged hours correctly in prior versions.
4. **10,000 GDU Break**If the GDU creation loop creates 10,000 GDUs it will break the loop and issue a warning.

Post Processor

1. **Leap Year Ozone Season Summation Bug Fix**When the postprocessor was run on years with leap years the ozone season was shifted forward one day. This has been fixed.

For SMOKE Postprocessor

1. **Suppress PUSP Percentage Distribution Warnings**A new command line option (--suppress\_pdw) was added to suppress warnings that the percentage distribution for a unit is not exactly 100% to allow for 1% tolerance.
2. **Retired Unit Warnings**The code no longer reports problems with missing data in the pusp file for retired units, nor does it report problems for emission rates that end before the projection year.
3. **Additional Header File**A new csv was created that allows one entry per line of additional lines to add to the header of the SMOKE files, both annual and hourly.

## ERTAC Change Log – v2.1.2

Known Bugs

1. **Disappearing Generation Bug**In situations when many units are curtailed a bug can occur where not all of the excess generation pool is allocated. The code calculates whether any GDUs are needed to allocate the excess generation pool by looking at the capacity of each unit minus the generation that has been assigned excluding units that had been flagged as having reached their annual capacity or operating hour cap by that hour or were retired in that hour. Later in the algorithm when excess generation is allocated units that reach their annual capacity or operating hour cap in a future hour also get excluded from consideration. If that difference is too large not all of the excess generation pool will be assigned and the annual growth rate will not be honored.
2. **Lacking Bug**Similar to the disappearing bug, the calculation for the “lacking” column in demand\_generation\_deficit also does not consider that reach their annual capacity or operating hour cap in a future hour and in cases with large amounts of curtailment will undercount the generation that is lacking in a particular hour.

Preprocessor

1. **New Non-EGU Bug**In previous version if a “new” unit was added and labeled as non-EGU it was included in several files, most importantly the hierarchy files, even though it was receiving no generation since it is a non-EGU. Non-EGUs that were not “new” were not included in these files so including “new” non-EGUs was not technically appropriate. Non-EGUs were being added primarily for record keeping purposes. This version of the code no longer carries new non-EGUs forward in files, in which existing non-EGUs had been removed from.

## ERTAC Change Log – v2.1.1

Known Bugs

1. **Disappearing Generation Bug**In situations when many units are curtailed a bug can occur where not all of the excess generation pool is allocated. The code calculates whether any GDUs are needed to allocate the excess generation pool by looking at the capacity of each unit minus the generation that has been assigned excluding units that had been flagged as having reached their annual capacity or operating hour cap by that hour or were retired in that hour. Later in the algorithm when excess generation is allocated units that reach their annual capacity or operating hour cap in a future hour also get excluded from consideration. If that difference is too large not all of the excess generation pool will be assigned and the annual growth rate will not be honored.
2. **Lacking Bug**Similar to the disappearing bug, the calculation for the “lacking” column in demand\_generation\_deficit also does not consider that reach their annual capacity or operating hour cap in a future hour and in cases with large amounts of curtailment will undercount the generation that is lacking in a particular hour.

Preprocessor

1. **Leap Year Bug**The preprocessor no longer maintains the extra 24 hours in calc\_hourly\_base.csv from leap years, which caused the projection code to crash.

Projection

1. **BY Controls on BY=FY Runs**When running base year = future year runs only control entries that have dates in the base year will no longer be ignored. They continue to be ignored when the base year and future year are different.

Python Post Processor

1. **New Flags**The hourly state level summary and the hourly regional level summaries will now only run if specified using the --include-st-hr and --include-rg-hr flags, respectively. Also added was a flag --include-unit-day to run a new feature that produces a summary for each day of the year for every applicable unit.

SMOKE Post Processor

1. **Design Capacity/Design Capacity Units**The design capacity and design capacity units columns are now populated with data.

## ERTAC Change Log – v2.1

Known Bugs

1. **Disappearing Generation Bug**In situations when many units are curtailed a bug can occur where not all of the excess generation pool is allocated. The code calculates whether any GDUs are needed to allocate the excess generation pool by looking at the capacity of each unit minus the generation that has been assigned excluding units that had been flagged as having reached their annual capacity or operating hour cap by that hour or were retired in that hour. Later in the algorithm when excess generation is allocated units that reach their annual capacity or operating hour cap in a future hour also get excluded from consideration. If that difference is too large not all of the excess generation pool will be assigned and the annual growth rate will not be honored.
2. **Lacking Bug**Similar to the disappearing bug, the calculation for the “lacking” column in demand\_generation\_deficit also does not consider that reach their annual capacity or operating hour cap in a future hour and in cases with large amounts of curtailment will undercount the generation that is lacking in a particular hour.

Preprocessor

1. **Version 1 to 2 converter**A new converter that is only needed one time has been added to convert input files from v1.02 compliant to v2.01.

Projection

1. **Disappearing Generation Bug Fix**A partial bug fix was implemented to ensure that the excess generation pool was allocated. In v1.02 in cases where a large number of unit’s annual capacity limit was reached GDUs would not be created to allow all of the excess generation pool to be allocated in later hours. The hour and capacity needed to meet the hour that lacks capacity the most are logged. The bug fix is only partial since it only accounts for units that had already reached their capacity limit by the hour being queried and it needs to also consider units that will reach their capacity limit in a future hierarchy hour.
2. **Lacking Bug Fix**In v1.02 the lacking column in demand\_generation\_deficit.csv was back calculated after a GDU was created, but this was problematic when an additional trigger for GDUs was added to fix the disappearing generation bug. This fix involved calculating the lacking generation prior to the creation of any GDUs. This is only a partial fix since it also suffers from the same lack of insight into capacity needs caused by units reaching their capacity limit in a future hierarchy hour.   
     
   Additionally a new flag was added --suppress-gdus that allows users to run the code without GDU creation in order to use the excess generation pool that remains in calc\_generation\_parms.csv as a more accurate “lacking” column.
3. **Emission Factor Averaging**  
   The new version of the model now can use varying emission factors at each hour for existing units, instead of only using one annual average factor, or using only an Ozone season (OS) average and a non-OS average. Emission factors can vary based on individual hours of operation during the base year, or can be averaged for each day, month, or quarter, as well as still having options to use OS/non-OS or annual averages.
4. **Heat Rate Averaging**  
   Similar to emission factors, the new model can also use varying heat rates at each hour for existing units. Individual hourly heat rates can be computed from base year hourly gross load and heat input, or averaged for days, months, quarters, OS/non-OS, or annually.
5. **Heat Rate and Emission Rate Outlier Flagging**  
   In conjunction with the hourly emission factors and hourly heat rates, the model now allows the specification of hard lower and upper limits for each of those values, and optional calculation of statistical limits. Computed emission factors and heat rates which fall outside of these limits will be flagged for review.
6. **Demand Transfers**  
   The model also has a new ability to read a user-supplied list of hourly demand transfers between region/fuel groups. This allows the model to show effects of shifting demand from one fuel to another within a region, and shifting demand (and emissions) from one region to another, and shifting demand from fossil fuel combustion to greener energy sources. If the demand transfer is larger than the generation in the origin region and fuel/unit type prior to the transfers the code returns without completing.
7. **Operating Hour Cap**An additional cap can be set in the UAF that allows units to be limited to operating a certain number of numbers annually. The functions act similarly to the annual capacity checks and columns were added to the hourly diagnostic file to allow for this functionality.
8. **Other Minor Changes**For units which report steam load (SLOAD) instead of gross load (GLOAD) and had fractional hours of operation during the base year, it was found that we need to scale the SLOAD value in the preprocessor to change it from a rate into total production during each partial hour. This change matches an earlier change for GLOAD in the 1.02 version of the model.  
   A typo was fixed in the projection code for determining if a unit had reached its capacity limit at the last hour of operation.  
     
   The output file prefix (-o option) is now used on preprocessor and projection log file names to be consistent and group them alphabetically with other output files from the same run.  
     
   For units which report SLOAD instead of GLOAD, the conversion to a GLOAD value is now done before temporal hierarchy ranking in the preprocessor.  
     
   The code checks to make sure growth rates were honored and if not provide a warning message.

SMOKE Post Processor

1. **Annual Summary**A new functionality allows you to include the prefix for the python post processor subdirectory using the --input-file-pp= flag. If included the SMOKE post processor will append the other pollutants to the file. The future\_year\_emission\_rate.csv file also includes a flag denoting where the emission rate was pulled from.
2. **Minor Changes**The --notz flag was changed to --runtz so it could be turned off by default since it is only necessary for unimplemented ORL file creation.  
     
   The --input-file-fs= flag was added to allow the input files to have a prefix attached to them.

## ERTAC Change Log - v1.02 (not released independently, included in v2.01)

Preprocessor

1. **Message Warning of Potential Leading Zero Issues**In cases where there are units in the UAF that do not match units in the CAMD base year file, but the unit UAF does match a unit in the CAMD file when the leading zeroes are removed a warning message is produced alerting the user that there are units that may have had their leading zeroes stripped in the UAF.
2. **Base Year Retirement Warning Message**A warning message is produced for any unit where there is CAMD data in the base year that occurs after the unit’s retirement date in the UAF.
3. **Secant Root Fix**  
   In cases where the peak and annual growth rates are set to be the same value the non-peak growth rate is set to be the same value and no warning messages concerning calculation issues are produced.
4. **Converting Generation from MW to MW-hr**  
   CAMD base year generation data is stored as capacity for the hour rather than power produced given to an hour. Now upon importing the CAMD base file the hourly capacity is multiplied by the portion of the hour the unit operated to determine the power produced in that given hour. All applicable file headers were converted to denote that gross load is now in MW-hrs rather than MW.

Postprocessor

1. **Ozone Season Generation Statistics**   
   Two columns (base year and future year) were added to calculate generation totals for ozone season.

## ERTAC Change Log - v1.01

Preprocessor

1. **Hiding Partial-Year Reporter Messages**If the argument --suppress\_pr is provided in the command line when running the program only 1 line of warning message is printed for partial year reporters indicating that the unit was processed as such.
2. **Calculate Operating Hours**Operating hours are summed based on the CAMD and non-CAMD hourly base year data. Any hour with gross load that is positive is counted. This occurs after the code calculates missing gross load based on steam load, but before partial year reporters are filled in.All hours are overwritten by this calculation unless the operating hours equal 0.
3. **Seasonal Controls**  
   An additional file is now available to input seasonal controls. It is nearly identical to the original control file, but with four columns to designate the season – season start month, season start date, season end month, and season end date. This allows a control to be turned on and off in each projected year without having a line item for each projected year. They can begin and end in certain years as well (so if the seasonal control does not start until 2014, and 2013 is run, it will not be considered applicable). The preprocessor converts this file into a regular control line item so that no additional files are necessary for the projection software.
4. **Hiding Non-EGU Error Messages**  
   The message “Warning: units with no ERTAC\_HEAT\_RATE” is no longer displayed for non-EGUs.

Projection

1. **Proxy Unit Bug Fix**  
   In the function assign proxy generation an additional check is conducted to see if the proxy generation for that hour is greater than the future projected generation. If that is the case rather than assigning all of the proxy generation that the unit is expected to generate only a portion is based on the ratio of future projected generation/total proxy generation for that hour.
2. **Placing GDUs in hierarchy after planned new units**  
   If there are new planned units GDU’s are added after the first planned new unit in the hierarchy. If there are no new planned units, GDU’s begin based on the percentile provided in the input file.

Postprocessor

1. **Python post processor developed**.
2. **ERTAC to SMOKE converter developed.**