# Tree Eater User Guide

## Background Notes

Tree Eater is one module in Neoprocessor, a set of Python scripts created and maintained by Joshua Petitmermet ([joshuapetitmermet@gmail.com](mailto:joshuapetitmermet@gmail.com)). This user guide focuses on using Tree Eater as a standalone script to calculate an average value per cubic foot for each tree size-species group bin that can be manually entered into the BioSum Processor Wood Values screen.

## Introduction

BioSum assigns value to merchantable sized wood in trees harvested as part of silvicultural treatments based on the user supplied parameters for each species group and diameter class. These values (in dollars per cubic foot) are multiplied by the FIA calculated cubic foot volume of the tree: VOLCFNET, which is from a 1-foot stump to a 4-inch top, and does not include rotten, missing, and form cull.

The challenge is that merchantable wood price information is available, regionally, by species but not by trees size. Wood is sold as logs, so log size (small end diameter) is the basis for value (given that logs of larger size can produce more valuable products, and with the right milling technologies, with less “waste”, or more accurately, less diversion to lower value products like chip board, than for the smallest trees). BioSum does not currently track logs, and does not have information on the length or small end diameter of the logs that could be produced from any felled tree.

In practice, there is an art/science to bucking tree boles (slicing them into logs) whether that is done at the stump (by a feller-buncher or chainsaw operator), the landing if using whole tree harvester (by a processor), or at the mill if whole boles are hauled there (by the head rig). The objective is always to maximize value of the products that can be milled from the log, and there are complex bucking rules that can be applied to achieve this, as well as log specific considerations such as the position of defect in a prospective log.

Tree Eater is an optimization script that, with the species, length and DBH of a tree bole, and a log price chart, optimizes the value obtainable from the merchantable sized wood in individual trees. It has the added benefit of being able to characterize sub-merchantable wood that can be utilized as biochar feedstock—a higher valued product than the “dirty chips” typically assumed in BioSum as the energy-facility-suitable feedstock that we account for as the entirety (including tops and limbs and bark) of sub-merchantable sized trees (those that can produce no merchantable logs) and trees of non-commercial species, along with the tops (where bole diameter falls below 4 inches) and limbs (if whole-tree harvested) of merchantable sized trees of commercial species.

While we hope to one day integrate Tree Eater directly into BioSum, opening a pathway for separating out feedstock suitable for biochar or pulpwood markets, for now (summer 2020), it can be used to calculate an average value per cubic foot for each tree size-species group bin. This is accomplished by running the full list of live trees that are 1-24” DBH AND that are of one of the currently supported species (or good candidates to represent one of the currently supported species).



Figure : The BioSum Wood Values tab contains the values used by BioSum to calculate merchantable and chip wood values. Tree Eater helps to more accurately estimate these values.

## Selecting and Mapping Tree Species

1. Tree Eater assigns log prices according to a Species group / log length matrix in the Python code. The structure contains six species groups that were present in the upper Klamath Basin, with available log prices, because they met each of three criteria:
   1. Being a species acceptable to local mills for producing traditional, high value wood products (not pulp or biochar)
   2. Being a species likely to be removed during a treatment aimed at fuels reduction, and
   3. Being a species with available price and sort data.

Table : Tree Eater species groups. Source: Petitmermet (2017)

|  |  |
| --- | --- |
| TE Code | Common Name |
| 1 | Douglas-fir |
| 2 | Incense cedar |
| 3 | Ponderosa pine |
| 4 | Sugar pine |
| 5 | Lodgepole pine |
| 6 | True fir |

An analyst should decide which species are merchantable in the study area and which Tree Eater species groups can serve as a proxy. Non-commercial species should be omitted from the mapping and analysis. Table 2 is the tree species mapping from the original Tree Eater implementation.

Table : Original Tree Eater species mapping. Source: Petitmermet (2017)

|  |  |  |  |
| --- | --- | --- | --- |
| FIA Code | Common Name | TE Code | Common Name |
| 202 | Douglas-fir | 1 | Douglas-fir |
| 81 | Incense cedar | 2 | Incense cedar |
| 122 | Ponderosa pine | 3 | Ponderosa pine |
| 116 | Jeffrey pine | 3 | Ponderosa pine |
| 117 | Sugar pine | 4 | Sugar pine |
| 108 | Lodgepole pine | 5 | Lodgepole pine |
| 11 | Pacific silver fir | 6 | True fir |
| 15 | White fir | 6 | True fir |
| 17 | Grand fir | 6 | True fir |
| 19 | Subalpine fir | 6 | True fir |
| 20 | Red fir | 6 | True fir |
| 21 | Shasta fir | 6 | True fir |

The tree species mapping can be adapted for study areas other than the Klamath Basin. Table 3 is a customized tree species mapping for the Blue Mountains variant.

Table : Tree species mapping for Blue Mountains Variant. Source: Fried (2020)

|  |  |  |  |
| --- | --- | --- | --- |
| FIA Code | Common Name | TE Code | Common Name |
| 73 | Western larch | 1 | Douglas-fir |
| 202 | Douglas-fir | 1 | Douglas-fir |
| 81 | Incense cedar | 2 | Incense cedar |
| 122 | Ponderosa pine | 3 | Ponderosa pine |
| 116 | Jeffrey pine | 3 | Ponderosa pine |
| 117 | Sugar pine | 4 | Sugar pine |
| 101 | Whitebark pine | 5 | Lodgepole pine |
| 108 | Lodgepole pine | 5 | Lodgepole pine |
| 119 | Western white pine | 5 | Lodgepole pine |
| 11 | Pacific silver fir | 6 | True fir |
| 15 | White fir | 6 | True fir |
| 17 | Grand fir | 6 | True fir |
| 19 | Subalpine fir | 6 | True fir |
| 20 | Red fir | 6 | True fir |
| 21 | Shasta fir | 6 | True fir |
| 22 | Noble fir | 6 | True fir |
| 93 | Engelmann spruce | 6 | True fir |
| 264 | Mountain hemlock | 6 | True fir |

1. FIA BioSum compatibility note: The TreeEater species code numbers are NOT the same as the BioSum Species Group numbers. There is no requirement that the BioSum species grouping exactly match the grouping used in Tree Eater for tree pricing, but the results are more straightforward if the two groups do match. Divergent species groupings have the potential to allow two species in the same BioSum species group with different prices per cubic foot (because they were processed differently in TreeEater). The BioSum interface only supports one price per BioSum species group.
2. To view or edit the tree species group mapping prior to running the script, update the SpeciesGroups.txt file. This tab-delimited file maps each FIA Species Code (SPCD) to a Tree Eater Species Group (SPGRP). Tree Eater is configured to support a maximum of six tree species groups.
3. The species group is used to assign a price, in dollars per MBF, to trees in that group. To view or edit the species group / log length price matrix, find the **c1-c6** dictionaries in the Python script. Example dictionary for species group 1: c1 = {8:481.25, 14:513.75, 22:526.25, 24:532.50}. Table 4 shows the prices associated with each range of log lengths. Note that the maximum length of logs is 24 feet unless changed in the Python dictionaries. For details on how the prices were calculated, refer to the original documentation for Neoprocessor (Petitmermet 2017)

Table : Price matrix for Tree Eater species group 1. Source: Petitmermet (2017)

|  |  |
| --- | --- |
| Log length | $/MBF |
| 6"-8" | $481.25 |
| 8"-14" | $513.75 |
| 14"-22" | $526.25 |
| 22"+ | $532.50 |

## Preparing the Tree Input File (MerchTrees.txt)

Depending on the underlying assumptions, an analyst can elect to include all trees from the study area in the input file, or to only include those that were cut according to the FVS simulation output. Tree records can be selected from either the BioSum master.tree table (all trees) or the FVS\_Cutlist table (trees harvested during the FVS simulation). The TreeEater tree input file must be a headerless text file that contains contain the fields and data types (delimited by commas) in the order shown in Table 5. The name of this file should be ‘MerchTrees.txt’. If the tree input file has a different name, the tar variable value in the Python script should be set to the file name: tar = "MerchTrees.txt"

Table : Required fields for tree input and their corresponding field names from the prospective input tables. Source: Petitmermet (2017)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Index** | **Field** | **Units** | **Data type** | **Description** | **Tree Table** | **FVS\_Cutlist Table** |
| 0 | Species | NA | Int | FIA tree species code | spcd | Species |
| 1 | DBH | In | Float | Diameter at breast height in inches | dia | DBH |
| 2 | Height | Ft | Float | Height of the tree in feet (estimated for broken tops) | ht | Ht |
| 3 | TrunHt | Ft | Float | Truncated height of a tree with a broken top in feet | actualht | TruncHt |
| 4 | PctCr | % | Float | Percent live crown ratio | cr | PctCr |
| 5 | Mdefect | % | Float | Percent defect in merchantable volume | ‘0’ | MDefect |
| 6 | TPA | NA | Float | Trees per acre associated with the tree record | tpa\_unadj | TPA |
| 7 | MCuFt | Ft3 | Float | Merchantable volume in cubic feet | volcfnet | MCuFt |
| 8 | StandID | NA | Text | Stand identifier associated with the tree | biosum\_cond\_id | StandID |
| 9 | TreeID | NA | Int | Tree identifier | fvs\_tree\_id | TreeId |
| 10 | Year | NA | Int | FVS model year | ‘1’ | Year |

## Example Tree Input File Queries

Example #1

When using the query below the analyst elected to value all standing trees in order to have a robust sample for every size class and species group for estimating average unit volume pricing. This query against the BioSum master.tree table also included the following logic:

* MDefect was set to a constant value of 0 so that the rough cull or cull rot estimates for a tree was not carried forward, in part because the effect of cull is already reflected in volcfnet. There were concerns that setting Mdefect to other than zero might double count the defect
* Only trees with >5” DBH were considered because only these trees had calculated merchantable volume (tree.dia).
* Only live trees were included (tree.statuscd)

SELECT tree.spcd, tree.dia, tree.ht, tree.actualht, tree.cr,   
**0 AS Mdefect**, tree.tpa\_unadj, tree.volcfnet, tree.biosum\_cond\_id, tree.fvs\_tree\_id, 1 AS [Year]   
FROM tree   
WHERE (((tree.spcd)=264 Or (tree.spcd)=202 Or (tree.spcd)=122 Or (tree.spcd)=119 Or (tree.spcd)=117 Or (tree.spcd)=116 Or (tree.spcd)=108 Or (tree.spcd)=101 Or (tree.spcd)=93 Or (tree.spcd)=81 Or (tree.spcd)=73 Or (tree.spcd)=22 Or (tree.spcd)=21 Or (tree.spcd)=20 Or (tree.spcd)=19 Or (tree.spcd)=17 Or (tree.spcd)=15 Or (tree.spcd)=11)   
AND (**(tree.dia)>=5**)   
AND ((**tree.statuscd)=1**));

Example #2

This query against the FVS\_CutList simply selects all the trees that were processed by FVS with no additional filters

SELECT FVS\_CutList.Species, FVS\_CutList.DBH, FVS\_CutList.Ht, FVS\_CutList.TruncHt, FVS\_CutList.PctCr, FVS\_CutList.MDefect, FVS\_CutList.TPA, FVS\_CutList.MCuFt, FVS\_CutList.StandID, FVS\_CutList.TreeId, FVS\_CutList.Year  
FROM FVS\_CutList

Table 5 includes the field names from both prospective input tables to add in query building if neither of the example queries is appropriate for an analysis.

## Configuration Files

All configuration files, including MerchTrees.txt must be placed in the same folder as the TreeEater Python script so that TreeEater can find them. Below is a brief overview of the supplemental configuration files. The analyst will most likely not need to edit these, unless the underlying assumptions need to be modified.

1. **dibparams.txt**: This headerless file is a tab-delimited, ordered list of parameters used for calculating the diameter inside bark during bucking. The parameters are applied in the same order as the TreeEater species groups. For example, the first line applies to TreeEater species group 1 (Douglas-fir).

The parameters for lodgepole pine are taken from the corrected Garber and Maguire (2003) and the parameters for all other species are taken from Hann (2016). The parameters are stored internally in a list (mdib) at the index appropriate for their tree eater species code (tesp).

1. **scribtable.txt**: This tab-delimited file holds the Revised Scribner board foot volumes for logs between eight and twenty eight feet in length (in four foot increments) and one and twenty four inches in top DIB (diameter inside bark) in one inch increments. These values are used during bucking to determine the volume of individual logs in each cut tree. The length index used to store Scribner values is in four foot increments, such that an eight foot log is at [2], a twelve foot log is at [3], and so on.

By default, twenty four foot and twenty eight foot logs are not considered during bucking. The source of these values is Bell and Dillworth (1988).

Table : Scribner table with board foot volumes for logs 8 - 24 feet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Index** | **Field** | **Units** | **Data type** | **Description** |
| 0 | DIB | In | Int | Diameter Inside Bark (1 – 24) |
| 1 | VOL\_8 | FBM | Float | Volume for 8 foot logs |
| 2 | VOL\_12 | FBM | Float | Volume for 12 foot logs |
| 3 | VOL\_16 | FBM | Float | Volume for 16 foot logs |
| 4 | VOL\_20 | FBM | Float | Volume for 20 foot logs |
| 5 | VOL\_24 | FBM | Float | Volume for 24 foot logs |
| 6 | VOL\_28 | FBM | Float | Volume for 28 foot logs |

1. **gwtparms.txt**: This tab-delimited file contains the green weight parameters as taken directly from the FIA Species reference table (Source: Forest Inventory and Analysis Database 2017). The green weights are used to convert harvest volumes into green ton weights for transport. Both the wood and bark specific gravities use a reference point of 62.4 pounds per cubic foot and both wood and bark moisture contents are measured on a dry weight basis.

Table 7: Definition of fields for the gwtparms.txt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Index** | **Field** | **Units** | **Data type** | **Description** |
| 0 | SPECIES | NA | Int | FIA tree species code |
| 1 | BARK\_VOL\_PCT | % | Float | Volume of bark as a percent of wood volume |
| 2 | WOOD\_SPGR | NA | Float | Specific gravity of wood |
| 3 | BARK\_SPGR | NA | Float | Specific gravity of bark |
| 4 | MC\_WOOD | % | Float | Default moisture content of wood |
| 5 | MC\_BARK | % | Float | Default moisture content of bark |

## Running Tree Eater

There are two ways to run Tree Eater. The TreeEater2.exe executable can be run using either an MS-DOS prompt or a Powershell prompt. It is easiest to set the current directory for either of these tools to the directory containing TreeEater2.exe. Type ‘TreeEater2.exe’ to launch the script. The program will print the number of stands loaded and the number of records written to the output files in the window. Note: When using Powershell, type ‘.\TreeEater2.exe’. Powershell does not load commands from the current location by default.

For either option, the configuration and tree list files described above MUST be in the same directory as the TreeEater.exe or TreeEater2.py script. The output will be also written to this folder. The required files are:

1. dibparams.txt
2. scribtable.txt
3. gwtparms.txt
4. MerchTrees.txt

Users who are proficient with Python can run the TreeEater2.py script using a Python environment of their choice. The script requires minimal external libraries and has been successfully run using Python 3.7 and 3.8. This option allows the analyst to customize the script if desired.

## Output Files

Tree Eater generates output files at the tree-level (priced\_t.txt) and stand-level (priced\_s.txt) in the same directory as the configuration files. BioSum users may find it easier to work with this files after importing them into MS Access.

1. **priced\_t.txt**: Cut trees are used to store all the data associated with trees harvested during a fuels treatment. All values are calculated for a single tree and reported in a row of this table.

Table 8: Fields, units, data types and descriptions from the Tree Eater tree-level output file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Index** | **Field** | **Units** | **Data type** | **Description** |
| 0 | SPCD | NA | Int | FIA Species code |
| 1 | DBH | in | Float | Diameter at breast height in inches |
| 2 | HEIGHT | ft | Float | Height of the tree in feet (estimated for broken tops) |
| 3 | TRUN\_HT | ft | Float | Truncated height of a tree with a broken top, in feet |
| 4 | PCT\_CR | % | Float | Percent crown ratio |
| 5 | MDEFECT | % | Float | Percent defect in merchantable volume |
| 6 | TPA | NA | Float | Trees per acre associated with the cut tree record |
| 7 | M\_VOL\_CF | ft3 | Float | Merchantable volume in cubic feet |
| 8 | STAND\_ID | NA | Long Text | Stand identifier associated with the tree |
| 9 | TE\_SPCD | NA | Int | Tree Eater species code |
| 10 | TREE\_ID | NA | Text | Tree identifier |
| 11 | HCB\_FT | ft | Float | Height to crown base in feet |
| 12 | YEAR\_CUT | NA | Int | Model year the tree was cut |
| 13 | RX | Text | Text | Prescription associated with the cut tree record |
| 14 | T\_PRICE | $ | Float | The total price of the tree in dollars |
| 15 | S\_PULP\_CF | ft3 | Float | The volume of rejected material in cubic feet |
| 16 | LOG\_PULP\_CF | ft3 | Float | The volume of biochar feedstock in cubic feet |
| 17 | SAW\_VOL\_CF | ft3 | Float | The volume of saw-quality material in cubic feet |
| 18 | LOG\_PULP\_GT | gt | Float | The green weight of biochar feedstock in tons |
| 19 | SAW\_WT\_GT | gt | Float | The green weight of saw-quality material in tons |

1. **priced\_s.txt**: This stand-level file provides per acre estimates

Table 9: Fields, units, data types and descriptions from the Tree Eater stand-level output file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Index** | **Field** | **Units** | **Data type** | **Description** |
| 0 | Rx | NA | Text | Prescription associated with the outputs |
| 1 | StandID | NA | Long Text | Stand identifier |
| 2 | Tprice | $ | Float | Gross revenue in dollars |
| 3 | MerchVol | ft3 | Float | Total saw-quality material produced in cubic feet |
| 4 | LogPulp | ft3 | Float | Total biochar feedstock produced in cubic feet |
| 5 | MerchGT | gt | Float | Total weight of saw-quality material in green tons |
| 6 | PulpGT | gt | Float | Total weight of biochar feedstock in green tons |

## Using Tree Eater Output in BioSum

After importing priced\_t.txt into an Access database as a table named Priced\_t, run the two following queries. For these queries to work, the field names in the Priced\_t table must match those in Table 8. Also, the species codes and diameters should be modified to be compatible with the current analysis. The species codes and groups in the switch statement must correspond with the BioSum species groups.

SELECT Switch([SPECIES] In (202,73),1,[SPECIES] In (108),3,[SPECIES] In (119),4,[SPECIES] In (122),5,[SPECIES] In (264,22,19,17,15),2,[SPECIES] In (93),6) AS SpGrp, Switch([DBH]<6,1,[DBH]<8,2,[DBH]<12,3,[DBH]<15,4,[DBH]<21,5,[DBH]<=24,6) AS DClass, Priced\_t.\*, [Tprice]/[MerchVol] AS DollarsPerCFFROM Priced\_t  
WHERE (((Switch([DBH]<6,1,[DBH]<8,2,[DBH]<12,3,[DBH]<15,4,[DBH]<21,5,[DBH]<=24,6)) Is Not Null) AND ((Priced\_t.MerchVol)>0) AND ((Priced\_t.SPECIES)<>101));

Save the query above as ‘TreePricePerCF2wDclass’. It will be used as the source for the query below:

SELECT SpGrp, DClass, Avg(DollarsPerCF) AS AvgOfDollarsPerCF  
FROM TreePricePerCF2wDclass  
GROUP BY SpGrp, DClass  
ORDER BY SpGrp, DClass

The results of the second query generates a price chart for entry into the BioSum wood values screen (Figure 1). Note that depending on the selection of trees in the tree input file, there may not be values for every species / diameter group combination on the BioSum wood values tab. Since BioSum requires that all of the boxes be populated, the analyst should use their best judgement. For example, if species group 4 has no trees size classes 2, 3 and 6 (and very few in cut lists) one could take the average, for those size classes, respectively, of species group 3 (which was more valuable) and species group 5, which was the opposite.

## References

Bell, J.F. and Dilworth, J.R., 1988. *Log scaling and timber cruising*. OSU Book Stores. Inc., Corvallis, Oregon.

“*Forest Inventory and Analysis Database.*” Department of Agriculture, Forest Service, Northern Research Station. St. Paul, Minnesota. [Available only on internet: <https://apps.fs.usda.gov/fia/datamart/datamart.html>] (2017).

Garber, Sean M., and Douglas A. Maguire. "*Modeling stem taper of three central Oregon species using nonlinear mixed effects models and autoregressive error structures*." Forest Ecology and Management 179.1 (2003): 507-522.

Hann, David W. "*Revised volume and taper equations for six major conifer species in southwest Oregon.*" Department of Forest Engineering, Resources, and Management. Oregon State University, Corvallis, Oregon. Forest Biometrics Research Paper 2 (2016).

Petitmermet, Joshua. “*Neo-Processor Documentation.*” Department of Forest Engineering, Resources, and Management. Oregon State University, Corvallis, Oregon. (2017).