# MAIN SCHEMATIC FOR FAME R CODE

# COMPONENTS OF THE TWO MAIN SCRIPTS

## FAME\_FMR FHanalysisTFI\_GS\_calculations

* Library imports and scripts
* Make directories and look up tables
* Fire history processing
  + FHanalysis (*EcoResFunctionsFMRv2*)
    - This output is called by most functions
* Crop rasters to polygons (either user input or DELWP area)
  + cropRasters (*EcoResFunctionsFMRv2*)
* Combine all fire sequence data
  + Calcu\_All\_Combs (*calc\_U\_AllCombs*)
  + myAllCombs (*FAME\_FMR FHanalysisTFI\_GS\_calculation*)
* Calculate time between fire intervals (TFI)
  + Calc\_TFI\_2 (*TFI\_functionsFMRv2*)
* BBTFI calculations (what is BBTFI?)
  + calcBBTFI\_2 (*calcBBTFI\_2*)
* GS Calculations (what is GS?)
  + makeGS\_Sum (*GS\_Calcs*)

## FAME\_FMR)Sp\_Calculations

* get list of taxon (either user input or state-wide data)
* read in species raster data
* make species abundance calculations
  + makeSppYearSum2 (*EcoResFunctionsFMRv2*)
* calculate changes in relative abundance
  + calcDeltaAbund (*EcoResFunctionsFMRv2*)

# DEFINITIONS

Provide a list of the abbreviations used and their definitions

(a non-exhaustive list)

RA\_Rasters

TFI\_Rasters - time fire interval?

GS\_Rasters

BBTFI\_Rasters

FPA –

APZ – Asset Protection Zone

BMZ – Bushfire Moderation Zone

LMZ – Landscape Management Zone

PBEZ – Planned Burn Exclusion Zone

myEFG\_TSF\_4GS

YSF, LBY, LFT – lby(last burnt year)

LTR

# MAIN FEEDBACK

* Grammatical consistency. More spaces make it easier to read, and shorter line lengths where possible. Also a full TRUE/FALSE is also easier to read.
  + This should be done, I have tried to put in spaces around all the <-. == etc to provide this, and gave functions with long input lists newlines to make it easier to see what is in them.
* Provide concise explanation at the top of each script and function within these saying what the script/function are intended to do and output type
* It appears the firetypes are set are repeatedly set independently. I suggest the look up table can be used to make it clear that that is where the 1 or 2 came from.
  + E.g. FIRETYPE\_LUT$TYPE[FIRETYPE\_LUT$FIRETYPE == 'BUSHFIRE'] instead of a 2. I know it looks like more code, but there is no room for confusion as to what the 2 means
* Script naming convention. The naming is hard to follow given you have nested scripts. Maybe a numerical prefix, followed by something that makes a little more sense will make it more obvious how it all goes together.
  + E.g. something like below that has a little bit of a hierarchy in it that is obvious but also has the flexibility to be added to (e.g. you can add in a ‘131’ down the track if you need to more TFI functions)
  + 001\_FAME\_Run whole analysis
    - * 010\_FAME default settings
      * 020\_FAME\_Function\_generic functions
    - 100\_FAME\_FIRE HISTORY ANALYSIS
      * 110\_FAME\_Function\_Fire History (incl FHanalysis/sequence functions)
      * 120\_FAME\_Function\_calculate all fire-spp combinations
      * 130\_FAME\_Function\_TFI
      * 140\_FAME\_Function\_BBTFI
      * 150\_FAME\_Function\_GS
    - 200\_FAME\_SPECIES ABUNDANCE ANALYSIS
      * 210\_FAME\_Function\_Spp calcs (move from EcoRes script)
* There are some duplicate functions. Suggest moving ‘generic’ functions to a new generic script, then running more specific functions within each script related to their output
  + E.g Join\_Names is duplicated in EcoRes and TFI, or something like notAllIn in Ecores would be a good candidate for a generic functions script.
* Function naming is ok, but its easier to know what is a user function vs package/base function if you give them names like ’Make\_HDM\_Values’ instead of makeHDMVals
  + This may be a little pythonic, but I find it helpful to know that this is my function vs others which generally aren’t named in Titlecase with underscore separators.
* Function input naming and declaring
  + Generally the function input variables are named as their inputs; it would be nice if these variable names were a little more informative.
  + Some inputs are defaulted to their inputs, and others aren’t. I suspect its too far gone now, but would be easier if it was all or nothing, preferably nothing with a comment saying what the input is (e.g. FHanalysis output dataframe, specific dataframe column, integer, raster, Boolean etc). A descriptive comment could be inline, or a block of comments straight after the function declaration saying what the inputs are.
* There are a few times where you read in the shapefile as a dataframe and remove the geometry. A generic function would be good for this.
* Tidy up the commented out lines etc when this revision is done.
  + Are ‘functions not used’ at the bottom of EcoResFunctionsFMRv2 really not used? E.g. makeHDMValsfromRasters is potentially called, others are not. I moved it above the line saying ‘Functions not currently used’ which all below could be deleted once the revision is complete.
* makeHDM\_Value\_Matrices.r isn’t sourced, its an option in FAME\_FMR\_Sp\_Calculations which the user would need to change in the script. Can this incorporated elsewhere or given a setting input that if True/False it gets run?
* I know its in the middle of a revision, but hopefully you can remove all the ‘\_2’ or ‘\_new’ at the end of functions and scripts.
  + E.g. "./ReferenceTables/DraftTaxonListStatewidev2.csv"