TCM User Guide

TCM – Test Case Manager, is used to manage your “File State Databases” (which are used by CART).

Usage:

./tcm [create|clear|install|edit|delete|list] <name> [path/to/myfile]

create : creates a new File State Database file (*.tar.gz)
clear : clears out all files and directories for a given version
install : unzips and untars the <name> file into ./conalaun (from testpackages/files/<VERSION>/<name>.tar.gz)
edit : extracts ./testpackages/files/<VERSION>/<VERSION>.tar.gz and replaces 'path/to/myfile' (inside the tar.gz) with file 'myfile' in ./updates directory.
delete : removes path/to/myfile from the ERAF (<VERSION>.tar.gz)
list : display list of all (*.tar.gz) files for current VERSION.

<name> : name of file to make (only for FSDBs, and only with 'create', and 'install' operations).

NOTE: 'edit' and 'delete' are only to edit the Expected Results Archive File (ERAF) - NOT FSDBs.
To edit an FSDB, just do: (a) tcm install <myfsdb>; then (b) change file system, then (c) tcm create <myfsdb>.

For 'edit' & 'delete', the version is taken from the VERSION environment variable. So do not specify <name>.

If you do not have yet set environment variable VERSION, TCM will bail. You must set VERSION environment variable.
The value of this variable must be a directory in...

./testpackages/files/$VERSION

Table of Contents

The CONFIG variable

IMPORTANT... check the value of variable called “CONFIG” in tcm.py.

CONFIG = "testpackages"

This has to match the value that CART uses to find your *.conf files.
Example

[victor@VCCK_DEV cart]$ ll testpackages/
total 12
drwxrwxr-x. 4 victor victor 4096 Dec 30 11:00 files
-rw-rw-r--. 1 victor victor 277 Dec 30 12:22 v1.000.conf
-rw-rw-r--. 1 victor victor 282 Dec 29 12:32 v1.100.conf

Right now CART uses “ testpackages” for that directory. Make sure TCM's value is same. It's just where your v1.000.conf, v1.100.conf, etc files are.

This should really never change.

Glossary

FSDB – file state database. FSDB is the old legacy name, but really it is just the set of raw ICO files to process. (in **cart_raw_ico**).

CREATE operation

To create a File State database (FSDB) , simply create directory **cart_raw_ico** and place your ICO files in it, then run TCM CREATE.
The below is assuming you set environment variable “VERSION” to “ **v1.000”**

[victor@VCCK_DEV cart]$ **tcm create mytest**

VERSION: v1.000
OPERATION: create
NAME: mytest

This will create a file in

testpackages/files/v1.000/mytest.tar.gz

It builds that path based on the VERSION and NAME parameters at the command line.

What is placed in this **mytest.tar.gz** file? Simple – all the files in **cart_raw_ico/***
Let's assume you had your testpackages/v1.000.conf file with the contents:

```
areas = x,y
[mytest.tar.gz:x,y]
“input A”
“input B”
“input C”
“input D”
```

Also assume that you want 2 ICO files to be processed after “input A” but before processing “input B”. Then you would place them in cart_raw_ico and name them as...

```
./cart_raw_ico/ico1-j
./cart_raw_ico/ico1-k
```

The charactors after the “ico-\d+” can be anything you want. All that matters is the file name starts with “ico” and then one or more integers, then a dash, then any charactors you want.

When CART runs and unzips **mytest.tar.gz**, the 'ico1-j' and 'ico1-k' will exist in cart_raw_ico.
Note that “input A” is input index [0]. and “input B” is index [1].

So CART sees “ico1-j” and “ico1-k”. The name must start with “ico-” (the letters “i”, “c” and “o” then a dash – any text can appear after and any file extension).

CART sees that the integer right after the dash is a “1” (in the filenames - “ico\d+”)... So it knows to process up to, but not including index [1] (which is “input B”). Then it will compile those 2 ICO files (ico1-j and ico1-k), THEN it will process “input B”.

CLEAR operation

If you want to get rid of all files that are part of a given FSDB, use “clear”.
This deletes all files/folders that were installed in (copied from being extracted from testpackages/files/v1.000/).
Example, if you used “create” to extract all files that are in testpackages/files/v1.000/myFsdb.tar.gz, then use “clear” to remove all those files.

All that the “clear” command does is delete all contents of **cart_raw_ico** directory.

INSTALL operation

If you issue the command..

tcm.py install myfsdb

In this example, Install simply extracts all files in **testpackages/files/v1.000/myfsdb.tar.gz** and places them in **cart_raw_ico.**

EDIT operation

What if you wanted to change the contents of the ERAF? (expected results archive file) ? (example, for version 1.000, it exists in . **testpackages/files/v1.000/ v1.000.tar.gz**) ??

Let's say the ERAF **./testpackages/files/v1.000/v1.000.tar.gz** contained the following...

./subdir_1/file_x.txt
./file_y.txt

So file_y.txt is in the root directory of the ERAF tar.gz. And file_x.txt is in a subdirectory (' subdir_1') of the ERAF tar.gz file.

How can we update these two files?? Easy.. place files with the same name in the "updates" folder (sub directory of where ./tcm.py is).

./updates/file_x.txt
./updates/file_y.txt

So ./updates/file_x.txt will end up overwriting /subdir_1/file_x.txt inside
./testpackages/files/v1.000/v1.000.tar.gz .
And ./updates/file_y.txt will end up overwriting ./file_y.txt inside
./testpackages/files/v1.000/v1.000.tar.gz .

To make the change to **file_x.txt,** run.... (assuming you set VERSION environment variable to " v1.000")..

tcm edit /subdir_1/file_x.txt

TCM will then edit the ERAF file **./testpackages/files/v1.000/v1.000.tar.gz** and replace /subdir_1/file_x.txt with file ./updates/ file_x.txt.
Of course it doesn't do that "inline" – it unzips, extracts, overwrites, and then does a "create".

To replace file_y.txt inside **./testpackages/files/v1.000/v1.000.tar.gz** with ./updates/file_y.txt... just issue...

tcm edit /file_y.txt

***** NOTE ** If you want to CREATE a new directory and file in the ERAF, simply provide the path … That is, if the path you specify to 'edit' actually does not exist, it will be created in the ERAF.***

that's it !!

Update - January 2023 - Included directories in an FSDB.

TCM now knows about other directories (other than just "**cart_raw_ico**"). Now TCM, when you do a "tcm create", it includes in the FSDB, all files in all the following directories:

```
drwxr-xr-x. 1 victor victor    0 Jan  1 18:41 alt_text
drwxr-xr-x. 1 victor victor   36 Nov 19 13:18 cart_raw_ico
drwxr-xr-x. 1 victor victor   48 Jan  1 18:39 groups
drwxr-xr-x. 1 victor victor   72 Jan  1 18:40 sr_code
```

So, "tcm create" now creates an FSDB with ALL the above files from all above directories in it. Thus, before doing your "tcm create" , make sure that all the files in these directories are sane !! Make sure that no "stray" files are inadvertently included. Make sure a stray version of "constant_strings.dat" is not along for the ride.