Chapter 7 -- FILE MANAGEMENT

This is the broad category of operations involving the use and abuse of files created by and for LTPlus. There are eight file management commands, five of which operate specifically upon the files identified by a single map name. These include **acquire, backup, create, retrieve and save**.

The remaining three functions (**catalog, purge,** and **rename**) are more generic in nature and can be implemented with any file type. Their actions can, in fact, be duplicated by standard UNIX functions outside of the LTPlus environment. They were included in LTPlus to assist users unfamiliar with UNIX commands.

CREATING A MAP

Every LTPlus map-based operation is dependent upon the existence of a current, active map. A new map is identified for use by first being created, using, as would seem proper, the **create** command. When a new map is created, its **header** data is initially set to the contents of the default **header** file in the **config** directory. This allows the user to save time in setting up the map's **header** file by setting many of the common **header** file elements for the maps within a given project. Once the map has been created, the user may use the **header** command to change one element at a time, or the **edit** command, in conjunction with the **header** file, to add several of the elements using the system editor. The most critical element of the header file which must be set prior to creating a map is the raster size (**raster_ht, raster_wd**). These elements must be established properly before creating a map which is to be manually digitized or into which vector data is to be imported. The raster dimensions are not critical if the map raster (scan) information is to be imported. Imported raster data carries with it dimensions of the scan. When a map is created, it is also acquired, negating any need to also execute a subsequent **acquire**.

ACQUIRING A MAP

Before any LTPlus map-based operations (other than **create**) can be performed, a current, active map must be identified. The **acquire** command identifies a map which was previously created and/or processed. Some operations which are not specific to a map may be performed with no current, active map present (**config, catalog, purge, ckeys_***, etc.).

ARCHIVING A MAP

Because every system has a finite amount of hard disk space, it is important to be able to archive map data when it is no longer needed on the system. The **backup** command provides a means for moving all of the files associated with a map name to a user-designated offline storage medium. This command looks for the designated backup prefix and suffix strings (identified as either 1 or 2 in the command string) and the **bkup_path** string in the **config** file to perform the operation. The first string element generally contains the system copy type command, the second contains the remainder of the copy type command (if any), while the **bkup_path** string represents the device location containing the media to which the files are to be copied. If no path name is given, the map file is assumed to have a full path name of the config file element **maps_path** concatenated with the map name supplied. For example, a Xenix-based system may have a **config** file **bkup11** string equivalency of **tar cvf**, with no string assigned to **bkup12**, and a **bkup_path** of **/dev/rfd0**. Executing the command **backup 1 big_buck** will generate the following system call:

> **tar cvf /dev/rfd0 /u[sr]/ltp/maps/big_buck**

This will cause all of the files associated with the named map (assumed to be under the **maps_path** directory) to be copied to the media in the drive identified as **/dev/rfd0**. The files will be stored on that media with the full path name (**/u[sr]/ltp/maps/[map name]**).

Executing the **backup** command with no parameters will cause the current, active map's files to be copied to the **bkup_path** file device defined in the **config** file.

CATALOGUING DIRECTORIES

The **catalog** command is very handy for determining what files are currently located under a specified directory type. If the **catalog** command is issued without a subsequent directory name, the contents of the default map directory (identified in the **config** file as the **maps_path** element) will be listed. A directory type may be included with the command; the choices are:

dictionary	**manual**	**map**
menu	**help**	**digmenu**
macro		

The catalog may be viewed on the screen or directed to the primary printer device (also indentified in the **config** file as print1).

REMOVING MAPS, FILES AND DIRECTORIES

Unwanted files may be removed from the system with the **purge** command. Keep in mind that this is a pretty permanent type of action. There is no wastebasket from which the file may later be retrieved. The user must identify the type of file which is to be eliminated with this command; the choices are:

map **maplist** **macro** **help** **menuset** **dictionary** **penfile**

Where multiple files exist under a directory, the actual file name is also required.

Any file may be **renamed**. This is not the same as making a copy. This is equivalent to a UNIX **mv** command. The user must supply the full path name of the file to be **renamed** and the full new path name. For example, to **rename** a map called **buck_mtn** to a new name of **big_mtn**, the command would be:

rename /u[sr]/ltp/maps/buck_mtn /u[sr]/ltp/maps/big_mtn

RETRIEVING ARCHIVED MAPS

To recover map data stored offline with the **backup** command, use the **retrieve** command. The source and destination **maps_path** and **bkup_path** locations must be correct, as defined in the **config** file. LTPlus assumes that the user is desiring to recover a map directory and its contents from the defined **bkup_path** device and place it in the **maps_path** directory identified in the **config** file. Using a **-t** argument with the **retrieve** command displays a listing (catalog) of the files available on the default **bkup_path** device, but does not perform the retrieval action.

SAVING MAP DATA

To avoid the pain and misery of lost map data (which is most often caused by an inappropriate or bizarre action by the user!), use the **save** command to store (and re-store) the active map's data files. The user will be prompted for this action when creating or acquiring another map or exiting LTPlus. It is a good idea to be pro-active and initiate the **save** command every 1/4 to 1/2 hour. The active map will be saved under the map name directory which will be found along the default **maps_path** as defined in the **config** file, unless an optional map pathname is entered as a parameter of the **save** command. Selected information about the current, active map is stored automatically by LTPlus in a temporary map file (**temp0**) under the default **maps_path** directory. However, much of the valuable map data is maintained in memory (where it can be lost) until the **save** command is executed by the user. The reason for requiring execution of the **save** command (vs an automatic **save** function) is to allow the user to make changes to a map without affecting its permanent appearance (from the previous **save** execution). The ultimate in flexibility can prove to be the ultimate in misery for a user who does not recognize the need to **save** desired changes!

Details about usage for all of the aforementioned functions may be found in the *Command Language Reference* section of this manual.