BCiTool_XI/XMLINI - TC-X Settings Utilities

Introduction

With TC-X, TurboCAD introduced a new format for it's settings file - TCW10_xml.ini. Problem is it has proved rather difficult to edit manually. Since the release on TC 10.1, quite a bit can be done in the TC Explorer, but there are still some occasions when it is easier to edit the settings file directly.

These functions address this problem until IMSI comes up with a better settings editor. I actually prefer using these functions to the old system of editing the ini file in Notepad (but then again, I did write them :-))

NB. These utilities are for 'power users', if you are not comfortable with general ini editing, don't use them.

New Goodies in XMLINI 1.1

- Works a little bit smoother with the changing location of the settings file - all functions are disabled until it's valid.

- Now has Backup button, so you can backup the current state at any time (in addition to the automatic backups created for the initial state of each session)... these backups are available for future sessions.

- Cleans up it's running Undo/Redo backups a bit better

Installing

- Unzip the package in the ...\TCWP100\Program folder. It should automatically create an XMLINI sub-folder

- This is the recommended location for it, though *shouldn't* really matter.

- You have to locate the TC settings file. This can be quite a challenge!, so here is a procedure finding it.

    o Run TC-X at least once, and shut down (this will create the default settings file)

    o In Windows Explorer, Search for files called "TCW10_xml.ini" with the current date. You may also need to enable Show Hidden Files in your default Folder Options.

    o Figure which is which if you get more than one hit (should be the one with the most recent date. In my case with TC Pro10.1 it is at

    ```
    C:\Documents and Settings\David Bell\Application
    Data\IMSI\TurboCAD\10\Professional\Profiles\Built-in\TCW
    10_xml.ini
    ```

You will need to know this location when you run XMLINI.

Running

<div style="border:1px solid black; padding:10px;">

Attention

The first time you run the application your anitvirus and/or firewall will probably complain about a potentially malicious script ...
you have to Authorise the script and this message should not reappear in the future.

You will also get a screen flash(es) when you run the functions ... this is just a brief DOS window running the GZip unpacking/packing functions.
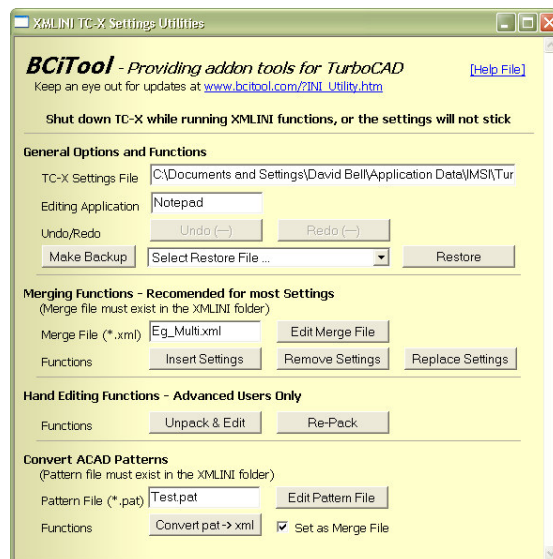
</div>

<div style="border:1px solid black; padding:10px;">

Attention 2

You must Shut Down TC-X while running these functions,
otherwise the settings will not stick!

</div>

OK, to run the program, browse to the XMLINI folder and open XMLINI_App.hta (hta = html application). I have set up a shortcut to it on my Start menu.

Then, the usual procedure goes

- (Shut down TC-X)
- Start XMLINI, run functions
- (Leave XMLINI running)
- Restart TC-X, check that changes are as expected
- If changes are not right, Shut TC-X down; switch back to XMLINI; Undo and/or run additional functions; Restart TC-X; etc

Functions explained

Before we start – Backups vs Undo/Redo

Backups and Undo/Redo work the same way, by making copies of the entire settings file and just swapping them as required. The difference is

- Backup files are only made when you click the click the Backup button, and they remain on disk indefinitely, so are available between XMLINI sessions

- Undo/Redo files are made automatically every time you use the Reload, Insert, Remove, Replace and Re-Pack functions below, so you can restore to any state within a session. However they are cleaned up at the end of a session so are not available between sessions.

General Options and Functions

(note to change the default values in the text boxes, open & edit XMLINI.hta in notepad)

TC-X Settings File	Important. You need to direct XMLINI to the TC Settings file. Since the release of TC 10.1, it is no longer possible to ship XMLINI with a default value for most installations ☹. So you have to do it.
C:\...\Program\TCW100_xml.ini	There is a procedure for finding it on Page 1 of this manual. Type or copy the full path into this text box
	You can set it as the default by right-clicking XMLINI_App.hta in Windows Explorer, opening in Notepad, and editing line 24 ... in my case I change it to ...
	``` txtTCINIPath.value = "C:\Documents and Settings\David Bell\Application Data\IMSI\TurboCAD\10\Professional\Pro files\Built-in\TCW10_xml.ini" ```
Editing Application Notepad	The application you will use for editing the settings files. Notepad is the default, Wordpad can also be used, or more specialised XML editors.
Undo/Redo Undo Redo	Undo/Redo changes made to the Settings file *within the current XMLINI session*. It is best to leave XMLINI running in the background until you are satisfied that the settings are as you would like them - so that Undo/Redo remains available.

Previous Backups	Make Backup makes a copy of the settings file that is available indefinitely. It doesn't have the fine resolution of Undo/Redo but it works *between XMLINI sessions.*
Make Backup	
Select ... ▾	The backup names are time-stamped with the form ....
Reload	Backup yyyy-mm-dd_hh-mm-ss

Automatic Merging

The following functions can be used to insert, remove or replace settings to/from TCW10_xml.ini with one click.

Merging is my preferred method for editing the TC-X setting file. Even if it's only a single setting, you can create a merge file almost as quickly as editing the setting directly, and once you have a working merge file it can be used again with confidence.

Also you can send merge files to other people that can change a single or many settings without affecting their other personalised settings - once you have them working well they are far more convenient than manual editing (IMO).

Caution: Merging has not been tested on all tables, if you find one that appears to malfunction, please let me know. DB

Merge File Name Eg_Multi.xml	This text box sets the name of the Merge file that is used with the following functions. It must be a valid xml file in the XMLINI folder.
Edit Merge File	Opens the Merge file in the editor of your choice.
Insert Settings	Insert Settings takes entries in the Merge File and **conditionally inserts or overwrites** them into the TC-X Settings file. Insert conditions Entries in the Merge file are compared to those already in the Settings file, using a Table x Var index created from the **name** attribute of both tags. Individual entries in the Settings file are then inserted/overwritten according to the following rules • If the entry does not exist, it will be inserted. • If the entry already exists, it will be overwritten*. • Insert can be used on all tables (AFAIK).

	See More on Merging below for more details of the merging process.
Remove	Remove Settings takes entries in the Merge File and **conditionally removes** them from the TC-X settings file. Remove conditions Entries in the Merge file are compared to those already in the Settings file, using a Table x Var index created from the **name** attribute and **value** the of tags respectively. Individual entries in the Settings file are then removed according to the following rules • If the entry exists, it will be removed*, however ... • Remove only works in the following tables - "BrushStyle_Table", "Paper_Table", "Color_Palette", "LineStyle_Table", "SHX_Fonts". See More on Merging below for more details of the merging process
Replace Settings	Caution: The Replace function is quite brutal, and should be used sparingly - mainly for when you want to *reduce* the number of settings. Replace Settings takes **tables** in the Merge File and **conditionally replaces** the **entire tables** in the TC-X settings file. Replace conditions Tables in the Merge file are compared to those already in the Settings file, using a Table index created from the **name** attribute of tags. Individual tables in the Settings file are then replaced according to the following rules • If the table exists, it will be replaced, however ... • Remove only works in the following tables - "BrushStyle_Table", "Paper_Table", "Color_Palette", "LineStyle_Table", "SHX_Fonts". See More on Merging below for more details of the merging process

* The 'BrushStyle_Table' table is an exception because the TC-X Setting file allows several entries with the same **name** attribute -which added together make a single pattern. In this table, if the entry already exists **all** entries with that **name** will be overwritten/removed (effectively overwriting/removing the entire pattern with that **name**).

Hand Editing functions

The following functions can be used to open/save the ini file for hand editing.

The Hand Editing functions are for advanced users. As noted previously, I believe it is better to change settings using Merge files. The only time I personally use these functions is for extracting prototypes for new merge files (more later).

Unpack & Edit	Copies and converts ...\Program\TCW10_xml.ini into a standard xml file named ...\XMLINI\TCW10.xml, and opens it in the editor of your choice. After you have finished editing you can Save in the editor then **Re-Pack** using the function below. If you don't want to save the changes, just don't Re-Pack.
Re-Pack	Converts ...\XMLINI\TCW10.xml back to ...\Program\TCW10_xml.ini. Once packed the xml file disappears, so you have to Unpack again to re-edit.

Convert ACAD .pat Files

This utility converts ACAD .pat files to an xml file suitable for using with the Merge functions above.

Pattern File Name Test.pat	This text box sets the name of the Pattern file that is used with the following functions. It must be a valid .pat file in the XMLINI folder.
Edit Pattern File	Opens the Pattern file in the editor of your choice.
Convert pat ->	Converts the .pat file into an xml file suitable for using with the **Insert Settings** function above. It also makes a txt file than can be manually edited into TC9 and previous versions. There is an example file called Test.pat that has three new brushes that I found at www.hatchpatterns.com
☑ Set as Merge File	Just a convenience, after converting the .pat, it sets the Merge file to the just created .xml

Keeping the number of patterns down.

There is nothing to stop you adding as many patterns to the settings file as you want. However, this can lead to unnecessarily bloated drawings because they all get loaded into each new drawing.

With XMLINI/TC-X you can use the following procedure to create a separate patterns collection drawing, so you can import specific patterns into drawings only when you need them.

- (Shut down TC-X)
- Set the merge file to the xml file of your choice, that contains the patterns you want to import.
- **Replace Settings** – TC-X now only has the patterns in the merge file (plus Solid & None)
- (Leave XMLINI running)
- Restart TC-X
  - New from Scratch (or open a file that already contains a pattern collection).
  - Optionally you can use the PatternSampler macro in BCiTool_PM to make a sample page from the new patterns
  - File Save
- Shut down TC-X
- Switch back to XMLINI
- **Undo Replace** – TC-X now has it's 'normal' set of brush patterns back again.
- Restart TC-X, and shut down XMLINI

Ok, so now we have TC-X with a normal set of brushes, and a saved drawing that contains an alternative collection of patterns. To use the patterns in a new drawing, do as follows

- With the drawing you are working on open …
- Open the TC-Explorer palette
- Browse to the file, then right-click on it's thumbnail and **Force Extract** – this will expose it's objects in the palette without actually opening the file.
- Still in TC Explorer, browse to and open the file's Brush Styles collection
- Drag the brushes you want into the working drawing area.

More on Merge Files

What Are Merge Files?

Merge Files are xml files that are identical in format to an unpacked TCW.ini file, though obviously they have different content. They should be a self contained fragment of the TC-X Settings file, and must be valid xml - you can check them by opening the Merge file in Internet Explorer which will report any errors if the file is not valid.

You can also think of them as a sort of script, that can either Insert, Overwrite, Remove or Replace settings - depending on the merge file contents and on which function activates it. In this sense they can be easily sent to others so that they can apply the same changes to their settings, without affecting their other personal settings.

The three merging functions **Insert**, **Remove**, and **Replace** work in the same way. Entries in the Merge file are compared to those already in the Settings file

> **Insert** inserts/overwrites **entries** line-by-line ...
> - Using a Table x Var index based on the **name** attributes of **both** tags
> - If the entry does not exist, it will be inserted.
> - If the entry already exists, it will be overwritten*.
> - Insert can be used on all tables (AFAIK).
>
> **Remove** removes **entries** line-by-line ...
> - Using a Table x Var index based on **name** attribute and **value** of tags respectively
> - If the entry exists, it will be removed*, however ...
> - Remove only works in the following tables -"BrushStyle_Table", "Color_Palette", "Paper_Table", "LineStyle_Table", "SHX_Fonts".
>
> **Replace** replaces entire **tables** ...
> - Using a Table index based on **name** attribute
> - If the table exists, it will be replaced entirely, however ...
> - Remove only works in the following tables -"BrushStyle_Table", "Color_Palette", "Paper_Table", "LineStyle_Table", "SHX_Fonts".
>
> * The 'BrushStyle_Table' table is an exception because the TC-X Setting file allows several entries with the same **name** attribute - which added together make a single pattern. In this table, if the entry exists **all** entries with that **name** will be overwritten/removed. (effectively overwriting/removing the entire pattern with that **name**)

What is the difference between Remove and Undo?

When you consider the Insert and Remove functions, it is clear that Remove is close but not exactly the same as Undo...

- Insert and Remove work on slightly different indexes - this is necessary because TC-X reindexes each time you shut down.
- Remove only works on certain tables.
  - Specifically Remove only works on those tables such as Paper_Table that can have any number of user defined entries. Entries in these tables are deemed 'safe' to remove.

- By comparison, for example, the 'Drawing_Defaults' table has a large number of entries that appear only once but may or may not be critical to TurboCAD functioning correctly. Until it is proved safe to do so Remove does not operate on these tables.
- Other tables can be added to Remove's scope when they are proved safe.
- Given compatible tables ...
  - if Insert actually **inserted** an entry, then Remove will effectively reverse this action, however ...
  - if Insert **overwrote** an existing entry, then Remove will delete the entry entirely - the original value will be lost and can only be recovered using Undo.
- Undo/Redo work in a completely different way. Each time Insert, Remove, Replace and Pack are used, they create a backup of the Settings file. Undo/Redo works by switching these backups with the current Settings file, and thus completely reverses or repeats actions taken with the other functions. (Note XMLINI does not delete these backups - they remain on your HDD until you manually delete them).
- Going back to Remove, if you kept a copy of the Merge file, Remove can be applied at any time, whereas Undo/Redo only works within a current XMLINI session.

A Few Tips for Creating Merge Files

- XML files can be edited in Notepad or any XML data content editor.
- **Check that they are valid xml**. An easy way is to try opening them in Internet Explorer. There are also a number of free xml validators/editors available on the web.
- Have a good look at the example files - merge files must be in a specific format that makes a self-contained fragment of the TC-X settings file.
- It is strongly recommended that you use the basic TCW ini file as a prototype for merge files. ie
  - **Unpack & Edit,** then extract the table(s) you want and go from there.
  - Look very carefully at the table headers ... they don't all have the same attributes in the table tag, *and frankly, I have no idea what the various table attributes signify.*
- When developing them I find the following procedure works well ...
  1. (Shut down TC-X)
  2. Start XMLINI
  3. Click **Edit Merge**, do your changes and save.
  4. Click **Insert Settings** (or Remove, Replace)
  5. (Leave XMLINI running)
  6. Restart TC-X and check changes.
  7. If it didn't work, shut TC-X down again, switch back to XMLINI, click **Undo**, and Go back to 3.
- Merge files can change as many or as few settings as you want .

Example Merge Files

Eg_Paper.xml	This Merge file demonstrates Inserting/Removing some entries into the 'Paper_Table' table.
	Used with **Insert**, all the papers in the Merge file have been assigned a new **name** attribute, and thus will be inserted (added).
	Remove will delete them, effectively reversing **Insert**.
	Note. While XMLINI always uses the **name** attribute for indexing, TC-X can use **name** or **id** or maybe both (in this case TC-X uses **id**). So when creating a Merge file for new tables you have to examine what is already in the TC-X settings file and follow it's model.
Eg_PenWidth.xml	This Merge file demonstrates Inserting some entries into the 'PenWidths' table.
	Used with **Insert**, the entries would usually already exist, and would thus be overwritten.
	Remove does not work on this table.
Eg_Color	This Merge file demonstrates Inserting/Removing some entries into the 'Color_Palette' table.
	Used with **Insert**,
	• the first entry in the Merge file (RGB65) would usually already exist, and would thus be overwritten • the last three entries (DJDB_RGB1-3) would not normally exist and would thus be inserted.
	Remove does not work on this table.
	This example demonstrates the importance of the **name** attribute in Merge files. If say, you have a color palette that you would like to distribute, it is best to come up with more-or-less unique names for each entry, so that it can be Inserted from other people's setups without affecting their other color settings.
Eg_Brush	'BrushStyle_Table' table.
	Used with **Insert**, the pattern has a more-or-less unique name, so it would usually be inserted (added).
	Remove will delete the pattern, effectively reversing **Insert**.
	Note that BrushStyle_Table is unique (I think) in that it can have several entries with the same **name** attribute (which added together make the patterns)

Eg_Multi	The example just combines all of the above files into one merge file. Applying large files such as this one is exactly the same applying all of the small files one after the other.
	There is no limit on how many or how few settings can be changed with a single Merge file.

Acknowledgements & Licence

These functions provided by David Bell (BCiTool), with some help from Tom Snape and others, Use at your own risk, no responsiblity taken, blah blah blah....

Feel free to send XMLINI to your friends, but *please* only as the entire unmodified distribution package, or better still point them to www.BCiTool.com/?INI_Utility.htm so they get the latest version.

Also in the package are GZipping functions provided by the GNU people at Free Software Foundation, under the GNU Public Licence. My reading of their licence agreement is that it's OK to include their tools in free distributions such as this one - but it's all there for you read in the GNU licence file included in the XMLINI package if you need a cure for your insomnia :-). To get a clean and complete GZip package, go to http://gnuwin32.sourceforge.net/packages/gzip.htm

Eg_Paper.xml

```xml
<?xml version="1.0"?><Settings
xmlns:noNamespaceSchemaLocation="TCSettings.xsd">
<Table name="Paper_Table" editable="true" first_id="1" type="0002">
     <Var name="P11" id="11" type="string">
          <Value>Arch1 (12 in x 18 in)</Value>
     </Var>
     <Var name="P12" id="12" type="string">
          <Value>Arch2 (18 in x 24 in)</Value>
     </Var>


     ...

</Table>
</Settings>
```

Eg_PenWidth.xml

```xml
<?xml version="1.0"?>
<Settings xmlns:noNamespaceSchemaLocation="TCSettings.xsd">
    <Table name="PenWidths" editable="true">
        <Var name="in" type="string">
            <Value>0.01,0.02,0.035,0.05,0.1,0.2,0.5,1,2</Value>
        </Var>
        <Var name="mm" type="string">
            <Value>0.1,0.2,0.35,0.5,1,2,3,5,7,10,20</Value>
        </Var>
    </Table>
</Settings>
```

Eg_Color.xml

```xml
<?xml version="1.0"?>
<Settings xmlns:noNamespaceSchemaLocation="TCSettings.xsd">
    <Table name="Color_Palette" editable="true" first_id="1" type="0012">
      <Var name="RGB65" type="string">
            <Value>OwWheat,216,216,191</Value>
      </Var>
        <Var name="DJDB_RGB1" type="string">
            <Value>DJDBWheat,216,216,191</Value>
        </Var>
        <Var name="DJDB_RGB2" type="string">
            <Value>DJDBWhite,255,255,255</Value>
        </Var>
        <Var name="DJDB_RGB3" type="string">
            <Value>DJDBYellow Green,153,204,49</Value>
        </Var>
    </Table>
</Settings>
```

Eg_Brush.xml

```xml
<?xml version="1.0"?>
<Settings xmlns:noNamespaceSchemaLocation="TCSettings.xsd">

<Table name="BrushStyle_Table"  editable="true">
      <Var name ='BCLogo' type='string'>
            <Value>T,H, 315, .1552854442, .7386485387, .7071067812,
.7071067812,2, .075,-1.3392135624</Value>
      </Var>
      <Var name ='BCLogo' type='string'>
            <Value>T,H, 0, .2083184528, .6856155301, 0, 1,2, .0353553391,-
.9646446609</Value>
      </Var>


      ...

</Table>
</Settings>
```


Eg_Multi.xml

```xml
<?xml version="1.0"?>
<Settings xmlns:noNamespaceSchemaLocation="TCSettings.xsd">

<Table name="Paper_Table" editable="false" first_id="1" type="0002">

      ...

</Table>

< Table name="PenWidths" editable="true">

      ...

</Table>

<Table name="Color_Palette" editable="true" first_id="1" type="0012">

      ...

</Table>

< Table name="BrushStyle_Table"  editable="true">

      ...

</Table>

</Settings>
```