

Ways to Merge/RemoveAreas/Stage Routes

By Chuck Rickert
September, 10, 2011

Ways to Merge/RemoveAreasFrom/Stage Routes

This document is about the various ways and methods an MSTS route builder can use in merger and removal projects.

Such as whenever a builder wants to:

- Use part of a route or the whole route in another route in a merger project.
- Remove an area from a route in preparation for a merger, or to re-do it or eliminate excess in a removal project.
- Use a modular approach for building a route in stages.

The most crucial challenge

- The most crucial challenge a builder faces in such projects is synchronization of data bases for the changes being made.
- There are various manual and automated ways and methods that can be used to do this synchronization.
- That's what this document is all about - preparing for, choosing, and using the best method for meeting that challenge.

There are the usual methods

- The usual methods are suggested in forums, and rely on use of RE rebuild/selects/deletes in merger and removal projects.
- These methods have advantages, but also have significant drawbacks which may put a painful burden on the builder.
- The usual method suggested for dealing with whole route mergers relies on TSUtil MERGE and related functions.
- This method has many advantages which can remove burdens on the builder, but requires a learning curve beyond RE.

And there is a seemingly unknown and unusual method

- This method appears to be unknown and unusual if lack of forum coverage is any indication.
- It relies on some lesser known functions along with some very familiar ones, and uses them in lesser known ways.
- It can dramatically reduce the burden on the builder, opening the door to projects not thought possible or feasible.
- It is much more powerful than the usual methods, and can be used in removals and whole or partial route merger projects.

With this method a builder can:

- Extract a city/yard/branch (or whatever) of any size from a route and turn it into a complete route by itself
- Remove an area from a route to open up space to receive "whatever" in a merger, to re-do it, or just to remove excess
- Merge "whatever" into another route
- While retaining essentially all of the original work in the wanted portions in every step along the way
- And do it with very little activity in the Route Editor. It can even be done without using RGE (or Demex).
- An easy example, covered later, can be used to illustrate all of this using a route everyone has.
- The method deserves more coverage if it works for others as I think it may.

Thus this document

- It describes the various methodologies and tools used in merger/removal/staging projects.
- It is a replacement for my original basic guide for combining routes at the hawkdawg.com site.
- This one covers all the usual methods along with special emphasis on the unusual one including examples.
- The document is appropriate for those who have at least some experience with creating and/or modifying route(s).
- It spans from basics to extensive details as a boost up the learning curve for anything here that might be unfamiliar.

Document topics are:

- Part 1. Merger And Removal Projects
- Part 2. The Usual Methods
- Part 3. The Unusual Method
- Part 4. Examples Using This Method
- Part 5. How To Do A Merger Project
- Appendix

Appendix topics are:

- Why/When/How To Convert (Horace) A Route
- How To Get Two Routes In One RGE Or AE Display And Find Overlap
- How To Use RGE To Reshape A Route
- How To Use TSUtil and Route_Riter In Merger/Removal Projects

So, FWIW:

- If you want to merge "whatever", remove "whatever", or build a route in stages

- Then read on and see what works for you.

PART 1. Merger And Removal Projects

Merger And Removal Projects

If you are thinking about doing a project to merge a whole or partial route with another, remove an area from a route, or build a route in stages, there is a lot to know, figure out, and do. Such as:

What are merger projects?

- Merger projects are those in which files from one route are used in another via manual operations or automated functions.
- The route being merged into is referred to below as the main route, the primary, with the other as the secondary.
- The files can be a subset of world/tile/scenery files (part of a route), or essentially all files (a whole route).
- Changes being made have to be properly reflected (merged) into the primary route's data bases and folders.
- For partial routes, the builder prepares the routes, copies over wanted files, and uses RE to merge data base changes.
- For whole routes, the builder prepares the routes and uses TSUtil MERGE to merge files and folders from both routes.
- The latter can also be exploited to allow building a route in stages, but more about that later.

What are some of the things to know in merger projects?

- Understand the fundamentals of the MSTS world and the locations and shapes of routes within it.
- Know the routes sufficiently enough to determine their locations, shapes, and where they can be joined.
- Understand that preparation may be necessary to ensure routes will be compatible and fit with minimum gap and no overlap.
- Preparation may include reshaping, moving, and adjusting the altitude of one or both routes to achieve proper fit.
- Be aware of the various methods, techniques, and tools available to reshape, move, adjust altitude, remove, and merge.
- Understand how data bases can be impacted by anticipated changes to world files.
- Be aware of possible conflicts between two routes that may require additional special attention and effort.
- These include track systems, signaling, distant mountains, and files with same names but different content.
- Understand that glitches will likely occur that will require digging into technical issues to find causes and fixes.

What are some of the things to figure out in merger projects?

- View and analyze each route to pick a "JctWF" and assess the shape of the border at or near the "JctWF" for each.
- "JctWF" is the world file name representing the junction from which tracks will later be extended into the other route.
- "Assess the shape..." means to address overlap/gap and select world file/tiles to delete/add to achieve proper fit.
- Choose the best method for synchronizing track and road data bases with desired changes.

What are some of the things to do in merger projects?

- Convert (Horace) a route if necessary so that track systems are compatible within the routes and the global MSTS system.
 - Reshape the route(s) as necessary, using the best method for removals of unwanted stuff. See removal projects below.
 - Move a route and/or adjust its altitude if necessary using TSUtil.
 - Merge desired files and folders from one route into the other using the best method.
 - Provide scenery, textures, and other files from one route into the other if not already present, including .ref file.
 - Address possible differences in .dat files, signals, etc. that may require manual merging or editing.
 - Join tracks, add scenery, and adjust terrain at the borders to get a smooth transition from one route to the other.
 - Fix any conflicts caused by shapes/textures and tertex files having same names but different content.
 - Repair or replace activities that may have been affected by track changes in world files.
-

What Are Removal Projects?

- Removal projects are those in which files are removed from a route.
- Those files are usually world files and/or tile files selected as unwanted.
- Changes being made have to be properly reflected (removed) from the route's data bases and folders.
- This can be done to reshape a route to remove excess, prevent overlap, open space for a merger, or to start over in it.

What are some of the things to know in removal projects?

- Understand the fundamentals of the MSTS world and the shape of the route within it.
- Understand how data bases can be impacted by anticipated changes to world files.
- Be aware of the various methods, techniques, and tools available to reshape the route via removal of world files/tiles.

What are some of the things to figure out in removal projects?

- View and analyze the route to spot the area(s) to be removed.
- Choose the best method for synchronizing track and road data bases with desired changes.

What are some of the things to do in removal projects?

- Select the specific world file/tiles that need to be deleted or added to achieve shapes for proper fit.
 - Reshape the route(s) as necessary, using the best method for removal of unwanted stuff.
 - Repair or replace activities that may have been affected by track changes after deleting world files.
-

For Both Merger and Removal Projects:

- The above is not a complete list of all aspects of such projects, but many key ones are included.
- It is meant to give a perspective about the scope of knowledge and effort involved for those who don't already know.
- It should be evident that one should already have some experience in creating or modifying a route.
- And be willing to learn and try the techniques and tools before undertaking a major project for real.
- For newbies and middies, the initial challenge is the learning curve. The document is designed to educate and show how.
- For middies and beyond, the document is designed to explain and show how for the few items that may be unfamiliar.
- The overriding challenge for a builder is to ensure that track/road data bases will be in sync with the changes.
- Much of this document is devoted to choosing and using the best method to meet that challenge.

There is much more that follows which will explain the above in much more detail, starting with the usual methods.

PART 2. The Usual Methods

The Usual Methods

This section contains explanations and "how to"s for the methods usually used in merger and removal projects.

The usual methods are:

- Route Editor "Rebuild" to create a new track data base (TDB) and road data base (RDB) from world files.
- Route Editor "Select" to "activate" tracks and roads from copied world files into the main route's TDB/RDB.
- Route Editor "Delete" to remove unwanted interactives/tracks/roads from data bases for the world files being removed.
- TSUtil "MERGE" to merge data bases and other files of two whole routes.
- Rarely used is manual editing of world files and data bases, It's done by gurus only, and not covered here.

RE Rebuild

- The RE Rebuild method relies on RE Rebuild to merge track and road items into a brand new initially empty TDB and RDB.
- It can be used for merger projects in which world files from a secondary route are copied into a primary route.
- It can be used for removal projects in which unwanted world files are deleted before its use.
- It can also be used in normal build projects to rebuild a new TDB/RDB from scratch out of necessity, or as a checkout.

RE Rebuild can be the easiest method to use in merger and removal projects because it provides an automated mechanism, but it does have some serious inherent drawbacks as will be evident next.

How to use RE Rebuild in merger projects:

- Prepare each route to be in its proper location/shape/altitude so that they would fit with minimum gap and no overlap.
- Refer to the "Preparation" section above and "How To Do A Merger Project" in Part 5 below as necessary.
- Delete any interactives in the portion from the secondary. RE Rebuild does not process interactives.
- Delete any interactives in the primary.
- Failure to delete interactives will result in RE error messages when near their locations. Take option to delete if so.
- Delete any dynamic tracks in the portion from the secondary unless they were originally laid in the primary.
- Failure to delete dynamic tracks in the secondary can cause fatal corruption. See "But beware of some issues" below.
- Copy the desired world files from the secondary into the primary. Any in the primary with same names will be replaced.
- Add/replace tiles to the primary route to cover the new world files if not already present.
- Copy shapes, textures, and ".ref" entries used in world files from the secondary that are not already present in primary.
- Delete all *.tdb, *.rdb, *.tit, *.rit, and their *.bk files from the primary route's main folder.
- Run RE "Rebuild" by clicking the Advanced button when selecting the primary, and checking "Rebuild..."
- Re-lay any deleted interactives originally in the primary, and any deleted dynamic tracks originally in the secondary.

How to use RE Rebuild in removal projects:

- Analyze the route to identify world files you want to remove.
- Delete any interactives in the wanted portion. Don't worry about interactives in the world files that will be deleted.
- It is not necessary to delete dynamic tracks in the wanted portion, nor from the unwanted portion.
- Delete the unwanted world files and tiles using Explorer and RGE.
- Delete all *.tdb, *.rdb, *.tit, *.rit, and their *.bk files from the route's main folder.
- Run RE "Rebuild" by clicking the Advanced button when selecting a route to load, and checking "Rebuild..."
- Re-lay any deleted interactives originally in the primary as appropriate. It's dynamic tracks should be okay.

But beware of some issues:

RE Rebuild does not handle dynamic tracks from copied world files as hoped or expected.

- Dynamic tracks in a copied portion are not added to the Local tsection.dat file of the main primary route.
- Instead, the Local tsection.dat is used as a reference when dynamic tracks are encountered.
- If they are from a different route, an S/DS will occur if one of their IDs can't be found in the Local tsection.dat.
- Or if from a different route, they will have invalid data for IDs that can be found in a highly unlikely coincidence.
- Valid dynamic track IDs with invalid data cause corruption that will be discovered at some point in time.
- If copied in dynamic tracks were originally laid in the primary AND REMAIN UNCHANGED from it, they should be okay.
- If unsure, safest bet is to delete all dynamic tracks in the portion to be copied before RE Rebuild.

RE Rebuild won't jump across empty world files

- When visiting world files, RE Rebuild won't jump across a gap of one or more world folders to get to a copied portion.
- Thus tracks and roads on the other side of the gap in the copied portion are ignored.
- However, they can be "activated" via the "RE Select" method below.

RE Rebuild doesn't properly handle XTrack crossovers

- After RE Rebuild, XTrack crossovers may be missing their tracknodes.
- This can be resolved by using AE to spot tracks that cross with missing black dot, then noting the latitude/longitude.

- In RE, each crossover can be found by latitude/longitude, then select/deselect/Save to activate it. See "RE Select" below.

Things can go wrong

- Usually when something goes wrong, the "Track database error: remove...vector.." message occurs.
- It may be caused by border tracks that straddle the border, cross multiple borders, or are too long.
- Note the world file name posted in the Camera window. you can use it as a clue for the general problem area.
- Restore a backup and look for problem tracks along nearby borders and replace tracks to remove the problem.
- Try again. It may take several iterations at several areas, or it just may not complete successfully.

- Also:
- It may be caused by RouteStart in the .trk file pointing to an empty or missing world file/tile. Change it.
 - It may be caused by too many items in a tile, or a route too big. Install MSTSBin or reduce the number of items.
 - It may be caused by a turnout from a switch or derail with no track attached to it. Attach a section.
 - Check in forums for other possible causes and solutions. There are a variety of causes for this message.
 - Look into <http://www.trainsim.com/vbts/showthread.php?303056-Track-Database-error.-vector-on-non-vector>

In Summary:

- RE Rebuild may be the best method for merger and removal projects because of its automated nature, if it works right.
- It may not be best if there are lots of interactives anywhere, and lots of dynamic tracks in the secondary portion.

RE Select

- The RE Select method relies on the builder to select/deselect each track and road section from copied world files.
- This merges the new items into the TDB and RDB of the main route in a synchronized manner.
- It can be used for merger projects in which world files from a secondary route are copied in.
- It doesn't apply for removal projects.

RE Select is not as easy as RE Rebuild in merger projects because it is not an automated process, but it does have some advantages over RE Rebuild, and some drawbacks as will be evident next.

How to use RE Select in merger projects:

- Prepare each route to be in its proper location/shape/altitude so that both would fit with minimum gap and no overlap.
- Refer to the "Preparation" section above and Part 5 for an approach on how to do this.
- Delete any interactives in the portion from the secondary. Do not delete any in the primary.
- Delete any dynamic tracks in the portion from the secondary. Do not delete any in the primary.
- Copy the desired world files from the secondary into the primary. There must be no overlap of active world files.
- Add/replace tiles to the primary route to cover the new world files if not already present.
- Copy shape and texture files from the secondary for scenery that doesn't already exist in the primary.
- Perform "RE Select" by selecting/deselecting each desired track/road in the copied portion to "activate" it.
- If for some reason you want to move a section, you can move it until you deselect. You cannot select multiples.
- Re-lay any deleted interactives and dynamic tracks originally in the copied portion as appropriate.

But beware of some issues:

- RE Select does not handle dynamic tracks from copied world files as hoped or expected.
- The same situation exists with RE Select as it does for RE Rebuild when encountering dynamic tracks.
- An S/DS will occur if one of their IDs can't be found in the Local tsection.dat as the world file is loaded.
- Or will have invalid data for IDs that can be found when selected, unless the data matches by coincidence.
- If copied in dynamic tracks were originally laid in the primary AND REMAIN UNCHANGED from it, selection should be okay.

In Summary:

- RE Select may be the best method for merger projects because it keeps the primary route's stuff intact.
 - No need to delete interactives in the primary.
 - It may not be best if there are lots of interactives and dynamic tracks in the portion from the secondary.
-

RE Delete

- RE Delete relies on the builder to delete interactives, tracks, and roads from no longer wanted world files.
- This removes them from the data bases in a synchronized manner.
- It can be used for removal projects.
- It can be used for a removal sub-project in conjunction with merger projects.

RE Delete is not as easy as RE Rebuild in removal projects because it is not an automated process, but it does not require interactives to be deleted from the kept portion.

How to use RE Delete in removal projects:

- Analyze the route to identify what you want to do.
- You are going to be deleting world files and tiles that you no longer want.
- Use RE to select and delete each interactive, track, and road in unwanted world files to remove it from the data bases.
- Always delete an interactive first before deleting the track or road section under it.
- Delete the unwanted world files and tiles using Explorer and RGE. No longer needed scenery files remain.
- Repair or remove any activity related files affected by removal of the world files.

In Summary:

- RE Delete may be the best method for removal projects when there are few items to delete.
- It has no issues of note.
- Otherwise it may not be best if there are lots of interactives, tracks, or roads to delete in the portion to be removed.

OUCH!

- Any of these methods can mean tedious if not painful RE sessions to delete and re-lay lots of objects.
- There may be so many to do that you may never start the project or abandon it after burnout.
- Unless there is a better way.

A Better Way?

- A better way to synchronize track and road data bases may be what I'll refer to as the TSUtil CLRDB/MERGE method.
- The tasks related to it can be used for both merger and removal projects.
- Nearly all extra RE deletion/relaying tasks and issues of the RE Rebuild/Select/Delete methods can be eliminated.
- Thus retaining essentially all original work in the wanted areas, including interactives and dynamic tracks
- Regardless of the size of the area involved.
- If unaware or have doubt, proceed to Part 3 to find out about the method, and then try the examples.

The TSUtil CLRDB/MERGE Method

This method relies on TSUtil CLRDB and MERGE functions to automatically synchronize the track and road data bases with changes made. The method gets its power from TSUtil CLRDB and MERGE, thus the basis for its name. Other TSUtil, Route_Riter, MSTs, and Windows functions are also used. When used in a methodology not normally associated with them, these functions and the method in which they are used opens the door to significant benefits not available in the other methods.

The Functions

The key TSUtil and Route_Riter functions pertinent to the TSUtil CLRDB/MERGE method are described below. Detailed instructions on how to use them are included in the Appendix.

TSUtil FILTER

- When mentioned in forums, it is usually in the context of removing excess tiles.
- FILTER provides an extremely easy way to remove all the tiles associated with world files that no longer exist.
- There may be cases in which it may be desirable to keep certain tiles that have no world files associated with them.
- They may act as terrain filler such as desert, mountains, and water to extend the white void horizon further away.
- FILTER will delete them, bringing the white horizon closer, maybe too close for camera views outside the cab.
- An interesting tutorial about this is at http://msts.steam4me.net/tutorials/remove_tiles.html.
- In those cases, RGE may be a better choice to delete tiles because of the precision available in it.
- Or FILTER and RGE can be used in combination - FILTER to remove tiles automatically, RGE for minor adjustments.
- FILTER provides an extremely easy way to remove huge amounts of unwanted tiles.

TSUtil CLRDB

- When mentioned in forums, it is usually in the context of resolving "Adjacent..." and blue pole issues in a world file.
- It checks each chain (path) in the TDB to see if associated entries can be found in world files.
- If an entry is found in any world file, the whole chain is considered used and is retained, even for missing world files.
- If none is found anywhere, the whole chain is considered unused and entries for it are removed from all data bases.
- The trick is to break chains that cross borders between wanted/unwanted world files, then give CLRDB only wanted files.
- This breaks each single chain into two different ones, one ending in a wanted world file, the other in an unwanted one.
- Running CLRDB will then prune and reorganize the data bases for just the world files given it in the World folder.
- CLRDB provides an extremely easy way to remove data base items from data bases for any number of world files.

Route_Riter COMPACT

- When mentioned in forums, it is usually in the context of preparing routes for distribution.
- It can be used to delete unused scenery files.
- Here it will be used for the exact same purpose, but now in conjunction with CLRDB.
- COMPACT provides an extremely easy way to remove the shapes and textures that existed only in deleted world files.

Route_Riter DUPLICATE

- When mentioned in forums, it is usually in the context of duplicating a route with a different name.
- Here it will be used for the exact same purpose, but now in conjunction with CLRDB.
- DUPLICATE provides an extremely easy way to rename an altered route to prevent confusion with the original route.

TSUtil MERGE

- When mentioned in forums, it is usually in the context of merging two whole and different routes.
 - Here it will be used for the exact same purpose, but now in conjunction with CLRDB and the other functions.
 - And instead of two different routes, MERGE can be used to merge the "same" route, as when building a route in stages.
 - When a route is built in stages in a coordinated manner, "sameness" allows easy merger of fully completed sections.
 - MERGE provides an extremely easy way to merge data base items from two routes along with other files and folders.
-

The Methodology

- The TSUtil CLRDB/MERGE method consists of two portions - a CLRDB based portion and a MERGE based portion.
- The CLRDB portion can be used to remove data base items that are no longer wanted in an area of any size.
- Thus it can be used to extract an area as an independent route, or carve out an area to open up space or remove excess.
- The MERGE portion can be used to automatically merge files and folders of two routes.
- Thus it can be used in traditional mergers and for building a route in fully completed stages.

Using the CLRDB portion in a removal project or merger sub-project:

- The CLRDB portion can be used to reshape a route by removing unwanted tiles, world files, and related scenery files.
- Complete the analysis from which the world files/tiles to remove have been identified, and "JctWF"(s). Defined above.
- Use WORDPAD to change RouteStart in the .trk file to a wanted world file. A "JctWF" would be a good choice.
- Use RE to delete only those interactives/tracks/roads that cross border(s) from wanted into unwanted world files.
- Use WINDOWS EXPLORER to delete all of the .w, .ws, and w.bk files for the no longer wanted world files.
- Use RGE and/or FILTER to delete all unwanted tiles.
- Run CLRDB to purge and synchronize data bases to the new set of world files.
- Use RE to see if the route loads and looks okay.
- Use AE to see if it loads and looks okay in the layout, and create a new activity if necessary to checkout in the Sim.
- Use Route_Riter/COMPACT ROUTE (optional) to delete shapes and textures that were used only in unwanted world files.
- Use Route_Riter DUPLICATE (optional) to change the new route's name to prevent conflict with the original.
- At this point you have a new but smaller fully functional route.
- If the route is to be used only by itself, you are done here and can proceed with normal route building activities.
- Otherwise, when doing a merger project, repeat the above steps on the other route if reshaping is necessary for it.

Using the MERGE portion in a merger project:

- The MERGE portion can be used to merge two routes by merging their files and folders.
- Complete a preparation phase after which the two routes are in their proper shape/location/altitude.
- Run MERGE to do the actual merge.
- Use RE to see if the route loads and looks okay.
- Use AE to see if it loads and looks okay in the layout, and create a new activity if necessary to checkout in the Sim.
- Proceed with route building activities to join tracks, terrain, and add scenery at the border to complete the project.

Want to build a route in stages?

- If you want to build a route in stages, either by a single builder or by multiple builders, coordination is critical.
- Coordination can lead to 100% compatibility in shape/location/altitude/terrain/scenery/signals/etc among portions.
- If 100% compatibility is achieved, issues related to overlap, gap, moving, and altitude adjustment can be eliminated.
- That would narrow down the major staging tasks to possible reshaping and merging, using the method chosen as best.
- Staging can be done with the usual methods of RE Rebuild/Select/Delete, but not with completed modules.

Want to build a route in stages using completed modules?

- If you want to build a route in stages using completed modules, the TSUtil CLRDB/MERGE method can do it.
- It is the only method that allows the use of fully completed modules without deleting/relaying lots of data base items.
- If a module is an extension, it can be done with a few selections and button clicks in just the MERGE portion.
- If a module is a replacement for an area with the same shape/location/altitude, then it can be done with both portions.
- After that, the project can be completed by joining the tracks and roads at the border.
- Example 1 in the examples below illustrates this.

Don't want to build a route in stages?

- Of course for most merger projects, two different routes are involved. High degree of compatibility cannot be expected.
- In this more typical case, all preparation tasks may apply in order to deal with lack of coordination between routes.
- This applies regardless of method used to synchronize data bases.

What about working on routes with interactives?:

- Two inherent aspects of this method are keeping original interactives and working on a route with them present.
- Some builders warn against working on routes with interactives, saying their presence will likely cause corruption.
- Forum postings by others indicate that merging/extending/enhancing routes with interactives is quite common.
- In all my messing with many routes having interactives, the fact they pre-existed never caused corruption.
- Differences in experiences/options may be related to the type of RE activity going on, the nature of the route, and luck.
- Extending a route while building or joining two routes after a merger is usually okay with interactives present or not.
- Deleting interactives or deleting interactives along with their associated track/road section(s) is usually okay.
- Manipulating lots of stuff before a Save is not okay when it goes too far for RE to handle, inviting bad luck corruption.
- RE can handle huge amounts of stuff (see PRR-Eastern Region) as long as it makes data base changes in small chunks.

So, what should one do?

- If you want 100% assurance that interactives won't haunt you while building a route, install them last.
- If present, you can delete them. When removing tracks or roads, delete their interactives first before deleting them.
- Otherwise, don't let interactives stop you from projects like these, and don't delete them unless the method demands it.

- Just be sure to perform prudent backup procedures and check things out along the way, interactives or no interactives.

Which method is best?

- Consider the type of project and the quantity of deletion and re-laying tasks facing you.
- If none or few, RE Rebuild and RE Delete may be best for removal projects, and RE Rebuild and Select for merger projects.
- If lots to do, then CLRDB/MERGE may be best for both removal and merger projects.
- With the RE Rebuild, RE Select, and RE Delete methods, the data base synchronization burden is on the builder and RE.
- With the TSUtil CLRDB/MERGE method, the data base synchronization burden is on TSUtil.
- The best method to use depends upon the routes, what you want to achieve, techniques/tools, and your ability to do it.
- Frankly, I trust TSUtil much more than I trust RE, and the TSUtil author is still with us, unlike MS and Kuju.

Is it smart to expect success when relying on the TSUtil CLRDB/MERGE method instead of RE?

- Odds are on the high side when using released routes. Most were built with little or no tricks and have no severe errors.
- Odds are on the medium to low side when using routes with data bases that were manually edited to get around issues.
- Odds are zero when using routes that are huge or contain serious errors that stop TSUtil but still run in RE/AE/Sim.
- My TSUtil experiences on lots of routes tell me that odds for success would be high for many if not most released routes.

Fear of the unknown

- If unfamiliar with using these TSUtil and Route_Riter functions, don't be afraid to learn and use them. They're easy.
- Benefits can far outweigh the effort to learn and use correctly. In most ways TSUtil is far easier to use than RE/RGE.
- If previous attempts at using TSUtil functions were unsuccessful, see the TSUtil and Route_Riter topics in Appendix.
- Also, you can't ruin your prized possessions if you always back them up just prior to using data base functions/methods.

To get beyond any lack of firsthand knowledge, doubt, or fear of the unknown, try the examples next.

Examples Using The TSUtil CLRDB/MERGE Method

- This section includes examples the reader can perform, aided by detailed step-by-step instructions.
- The first example is a merger project using a route you have/had, and which uses all of the method's steps/functions.
- The second example is a removal project on a route of your choice.
- Their purpose is to demonstrate the power of the TSUtil CLRDB/MERGE method and to educate the reader in its use.
- Don't be deterred by the number of tasks. Lots of handholding for first timers. They are easy to perform.
- It is critical to have Route_Riter at 7.6.18 (26 Aug 2011) and TSUtil at 3.6.02, or later. Install/update accordingly.

Before using the TSUtil and Route_Riter functions:

- Print the "How to use TSUtil and Route_Riter ..." in the Appendix for more detailed instructions as needed.
- Refer to the printout when performing the functions. Proper use is critical for increasing the chances for success.

=====

Example 1.

- Example 1 is a merger project using a route everyone has or had at one time. It is designed to:
- Extract a city (Baltimore) from the middle of a route (default USA1) to turn it into a functional route by itself.
- Remove an area from a separate route to create space to receive Baltimore. Same area and route for simplicity.
- Merge Baltimore into that route.
- Complete the exercise and run in the Sim as the ultimate test to see if everything worked as advertised.

Admittedly, no one would want to do this particular example for real, but:

- It does illustrate all of the functions and steps of the TSUtil CLRDB/MERGE method
- It does illustrate its applicability and power as a viable method for merger and removal projects
- Including its use to build a route in stages using completed modules.
- Give it a shot.

Extract Baltimore as a standalone route

This is a removal portion of a merger project. It removes everything from the original USA1 route except Baltimore, thus extracting it as a route by itself.

Convert the route if necessary

- USA1 is a default route.
- You may have to convert it or swap Globals depending upon your current Global tsection.dat file.
- If you don't know what to do or how, this is a great opportunity to find out in a guided manner and gain experience.
- Convert or swap Globals as appropriate. Refer to "Why/How to convert.." in Appendix for explanations and "how to"s.
- After doing the above line, proceed to "Use WORDPAD" below if conversion was NOT necessary or a Global swap was made.
- Otherwise if the route was converted, copy it elsewhere because it will be needed in a later step.

Use WORDPAD

This is to change RouteStart so that the route will start in Baltimore

- Place a copy of the original or converted default USA1 into the Routes folder.
- In RE/AE/Sim, it is displayed as "Northeast Corridor". In RGE and Explorer it is displayed as "USA1".
- For this example, Baltimore resides in two world file/tiles: -11059/14283 and -11058/14283. Both are "JctWF"s.
- Use WORDPAD to open USA1.trk and change RouteStart to -11058 14283, the eastern "JctWF" of Baltimore.

Use RE

This is to delete border crossing data base items between both sides of Baltimore and the rest of the route.

- Use RE with F7 to find the blue line border on the east side of -11058/14283. Tracks are in tunnels until near there.
- Use F2 to spot and delete the following four track sections between -11058/14283 and -11057/14283.
- (North to south below the marker - A1t870r4dlft, a dynamic track, A1t870r4drgt, and A1t10mstrt)
- Jump or move to the blue line border on the west side of -11059/14283, Baltimore's western "JctWF".
- Tracks are below ground, so use the "W" key and "/" key as necessary.
- Use RE with F7/F2 to spot the A2t500mStrtTun buried between -11060/14283 and -11059/14283, and delete it. Then Save.
- If you see some small white areas in the terrain within 11058/14283, just ignore them. They are Kuju defects (Gasp!).

Use EXPLORER

This is to remove all files from the World folder except those for Baltimore.

- Delete all files in the World folder except the .w and .ws files for Baltimore, those with -11058/14283 and -11059/1428.
- Or create a World folder elsewhere and do the next three lines.

- Copy the files (6) with -11058/14283 and -11059/14283 from the original World folder into the new World folder.
- Delete the original World folder.
- Cut and paste the new World folder with the files back into the route.

Use TSUtil FILTER

This is to remove the tile files belonging to the world files just deleted.

- Run FILTER to delete unwanted tiles. It is the "Remove .t where no .w exists" button. Respond "OK" to messages.
- Don't forget to copy the contents of "newRoute" back into the USA1 folder.
- Take a look at the route in RE. Notice that terrain and scenery outside of Baltimore are gone, but track paths remain.

Use TSUtil CLRDB

This is to purge and synchronize data bases to only the Baltimore world files.

- Run CLRDB to remove all items from data bases except those for Baltimore.
- Don't forget to delete the contents of the route's Activities/Paths/Services/Traffic folders before copying "newRoute".
- Also, because RE doesn't like it if it can't find an activity somewhere, make one or copy in a route that has one.
- Take a look at the route in RE. Notice that everything is gone outside of Baltimore.
- It's smart to select any one track and one road section and Save. This forces an update as a check for any hidden S/DS.
- If not successful, see Appendix for possible causes and resolution.
- If successful, data bases are only for Baltimore. Scenery files used exclusively for areas outside of Baltimore remain.

Use Route_Riter COMPACT

This is to remove scenery files not needed for Baltimore

- Run COMPACT to remove unused scenery files.
- If successful, unused shape and texture files have been removed.

Use Route_Riter DUPLICATE

This is to rename the new route to Baltimore

- Run DUPLICATE to duplicate the route to rename it. Specify "Baltimore" for route folder and route name.
- Delete USA1. It served its purpose. It will interfere with next steps if still present. Keep Baltimore for MERGE.
- Optional: Create a backup of Baltimore in a new folder elsewhere named "BaltimorePreMerge", but probably won't need it.

What you have now is Baltimore as a route. For the real world, this illustrates how an area can be extracted for use as a standalone route, or for use in a merger with another route.

Remove Baltimore from another copy of USA1

This is a removal sub project to carve out Baltimore from a separate copy of USA1 to open up space so that the extracted Baltimore route can be merged into that location.

Perform the same steps as above, but use the variations below to remove Baltimore and keep the rest.

- Place a copy of the original default USA1 into the Routes folder, or use a copy of the converted USA1 from CVRT above.
- Use WORDPAD to open USA1.trk and change RouteStart to -11057 14283 so that it will start just east of Baltimore.
- Use RE to delete the same border crossing items as above.
- Use Explorer to delete only the Baltimore world files, which are those with -11058/14283 and -11059/14283.
- Use FILTER to remove Baltimore's tiles. Don't forget to copy "newRoute".
- Run CLRDB to remove Baltimore's data base items. Don't forget to copy "newRoute".
- Don't bother to use COMPACT to remove Baltimore's scenery files unless you want to.
- Use DUPLICATE to rename the modified route. Specify USA1Mod for its route folder and route name.
- Take a look at the route in RE. Notice that terrain, scenery, and track paths for Baltimore are now gone.
- This is a good time for a backup in a folder named "USA1ModPostCLRDB" in case you need to rerun the next step.
- Delete the USA1 route from which the duplicate was made.

What you now have is a route with a hole in it. For the real world, this illustrates how an area can be removed to open up space to receive a whole or partial route in a merger project, or to remove an area as excess or to re-do it in a removal project.

Merge Baltimore into the non-Baltimore USA1

This is the merge portion to actually merge the Baltimore route into the modified USA1 route, and complete the project.

Use TSUtil MERGE

This is to merge the two routes.

- Both Baltimore and USA1Mod should be in the Route's folder.
- Run TSUtil MERGE to merge the two routes. Select USA1Mod as the primary.
- Use RE to check things out around the border areas of Baltimore.(-11058/14283 and -11059/14283).
- If successful, you now have USA1Mod with Baltimore merged back into it, ready for their tracks to be joined.

- This is a good time for a backup in a folder named "USA1ModPostMerge in case you need to redo the next step.

What you now have is a merged route. For the real world, this illustrates how the data bases/files/folders of two routes can be merged as one.

Complete the exercise - ultimate test

- You could stop here without joining tracks, and just create an activity to checkout things out in the Sim.
- Or you could perform the steps in the next section to join tracks to complete the project and do the ultimate test.
- Do the following for the ultimate test, otherwise you are done.
- Add the following tracks into the appropriate places in the gap between -11058/14283 and -11057/14283:
- (From north to south- A1t870r4dlft, a dynamic track with a length of 11.620, A1t870r4drgt, and A1t10mstrt)
- Be sure to use "T" as necessary to align. Check that path lines are continuous with no blue poles after laying them.
- Add A2t500mStrtTun in the gap between -11060/14283 and -11059/14283. Get below terrain by using "W" and "/".
- You now have USA1 back together again if you correctly performed the instructions here and in the Appendix.
- Check out in RE and AE to see if the route loads and looks okay.
- If not successful, see Appendix for possible causes and resolution.
- Create and run an Activity in the Sim as the ultimate test to see if you/method/tools did everything right.

What you now have is USA1 after taking it apart and then putting it back together again.
 Clean up after the example by deleting Baltimore and the newly merged USA1 when ready.
 If you renamed your Standardized Global to GlobalS at the beginning, delete default Global and rename GlobalS to Global.

How Example 1 relates to staging with fully completed modules

- The Baltimore that was extracted as a route could be thought of as a fully functional replacement module.
- The removal of Baltimore in the second part could be thought of as the removal of a previous module.
- The merger could be thought of as the inclusion of the new module into the rest of the route.
- Because of 100% compatibility, MERGE can take place immediately without analysis/reshaping/moving/altitude issues.
- The power - No need to remove/re-lay lots of interactives/dynamic tracks as would be necessary with other methods.

=====

Example 2

- Example 2 is a removal project using a route chosen by the reader.
- It's much simpler than Example 1 because of fewer steps.
- It demonstrates CLRDB's ability to easily remove the interactives/tracks/roads from a portion of any size of a route.
- The objective is to remove the eastern portion of your route, then seeing if CLRDB did its job as advertised.

Remove a portion of a route

- Pick one of your working routes that is compatible with your Global tsection.dat.
- Use AE to pick a longitude/latitude at some track point in the middle of your route suitable for a split east and west.
- Keep in mind that you will be deleting interactives and tracks that cross the border. The fewer the better.
- Use RE with that longitude/latitude to jump to the world file at that location.
- Write down the name of that world file. In a merger project, it would be your eastern "JctWF".
- Use RE with F7/F2 to delete all interactives/tracks/roads that cross the border from the "JctWF" towards the east.
- Check above (north) and below this world file and remove any data base items in them that cross the border eastward.
- Use WordPad to change RouteStart in the .trk file to the world file you picked as "JctWF".
- Use Explorer to delete all eastern world files, those with filenames that are east of (less than) the name in "JctWF".
- Use RGE or FILTER to remove all eastern tiles, those east of your "JctWF" - an example of achieving a straight edge.
- Run CLRDB to remove the data base items from data bases for all those deleted world files.
- Don't forget to delete the contents of the route's Activities/Paths/Services/Traffic folders before copying "newRoute".
- Don't bother with Compact and Duplicate unless you want to.
- Check the route in RE and AE to see if things look right. You should see nothing east of your "JctWF".
- Everything else should remain intact, including any activities with paths exclusively in the western portion.
- If not okay, tasks may have been done improperly, or the route and TSUtil are incompatible. See Appendix if so.
- Delete the route when done.

Hopefully the successful completion of the examples will provide a firsthand understanding and appreciation of the power of the TSUtil CLRDB/MERGE method as a viable and significant method when undertaking merger and removal projects. An approach on how to do a merger project is next.

Part 5. How To Do A Merger Project

How To Do A Merger Project

This is an approach on how to do a merger project. It is not error free or all-inclusive, nor does it contain the best or only ways to do things. But it does offer one builder's interpretation of what one needs to know, figure out, and do when undertaking a merger project. The topics are:

- Understand The Fundamentals
- Prepare A Route For A Merger
- Perform The Merger
- Complete The Project

=====

Understand The Fundamentals

The more one understands the fundamentals of the MSTS world as depicted in the Route Geometry Extractor (RGE), the higher the odds for success in removal and merger projects. Fundamentals include how the MSTS world relates to the real world in terms of the locations and shapes of routes, the manner in which they are referenced and manipulated, and the factors that must be considered. Certain factors can be particularly complex and difficult to address. However, by understanding the fundamentals more fully, and employing certain techniques that can help deal with complexities, the odds for success can be increased.

Maybe what follows will help one gain that understanding if currently lacking. The best place to start is the MSTS world itself, where a route begins.

The MSTS world

- The default MSTS world is a flat area at one meter above sea level from which the shape and location of a route is set.
- RGE is used to display this world and create initial files.
- RGE can also be used to reshape an existing route with additions and deletions of terrain, called tiles.
- The user makes this choice with "File/New Route..." for a new route, or File/Select Route..." for an existing one.
- RGE displays the whole world in an initial window, but cannot load all the data necessary for user actions.
- It requires the user to "Zoom Region" twice to get to the desired region containing the data for a route's area.
- Regions have demarcation lines (grid) to differentiate one from the other.
- After that, the user can "Zoom Window" in or out to facilitate selection of the area (tiles) to add or delete.
- Zooming is done by placing the cursor in a desired location and right clicking.
- A route may span from one region into another. Each requires its own region and window zooming, and user selections.

World grids and tile structures

- Like the real world, MSTS uses longitude(east/west) and latitude(north/south) to indicate world position from zero.
- With the MSTS world being flat, values used for it's longitude/latitude will not precisely match the real world's.
- While both use degrees to indicate position from zero, MSTS also uses a 2048x2048m square as a location reference.
- This square is the familiar world file/tile, with its name indicating position east/west and north/south from zero.
- For design purposes, MSTS superimposes three types of grids with row and column names over its world to form squares.
- These grids provide coordinates which MSTS and the builder can use for identifying location/distance.
- The first is an invisible grid used as a base to translate cursor position to longitude/latitude and world file name.
- It is preloaded and consists of the squares that are used for defining "world" files with names displayed at bottom.
- A second grid (default) is for user selection of squares to become "regular" tiles used as close-in ground terrain.
- A third grid is Distant Mountains (DM) tiles, which are mountains seen in the distance as created by Demex not RGE.
- The user selects which grid to load by choosing the fourth icon at the top left for regular tiles, the fifth for DMs.

Tile size

- Tile size determines the amount of terrain area covered by the tile.
- The size of the grid square as set by the user determines the size of the tile that will be generated (created).
- Ideally regular tiles should be the same size as world files at 2048m on each edge (side), thus on a one-for-one basis.
- A builder can mistakenly create regular tiles that cover multiple world files, with 2, 4, 8, or more on a side.
- An example can be seen in an RGE display of the default USA1 route, near the bottom on the east side.
- (For those with little RGE experience, this would be a good time to try the steps above and see the example.)
- The issue can be prevented by always setting grid squares to the smallest possible size when creating regular tiles.
- Distant Mountain tiles cover many world files, and are usually eight or sixteen world files on a side.

- When tiles are larger than world files, they can have significant impact in merger projects. More about this later.
-

From grid to regular tiles

- First, an altitude must be set via Edit/Set Height Offset". The value specified should be the desired value -1.
 - The general process is to set the grid squares to the smallest size, then select/populate/generate tile files.
 - All of this can be done via a cursor drawn selection box, right click, then "Add All..." and "Extract Tiles...".
 - Or it can be done in separate phases as follows.
 - Setting the grid square to the smallest size is done via right click then "Split", repeated as necessary.
 - The builder selects the grid squares he wants for his routes by "populating" them to flag his choices.
 - To "populate" selected square(s), they must be toggled via right clicking and selecting "Toggle Populated...".
 - Toggling will put an "X" into each selected square to populate it, or remove it to depopulate it.
 - To create actual tile files for populated squares, they must be generated via clicking "Edit/Generate...".
 - Regular tiles are stored in the "Tiles" folder. (Distant Mountains are stored in the "Lo_Tiles" folder by Demex)
 - The regular and DM tile structures, its quad-trees, are created and saved in the TD folder upon "File/Save Quad-Tree".
-

The world files/tiles

- MSTs uses grids with column and row names as its primary location indicator for world files/tiles.
 - Columns represent the distance east/west from the MSTs starting point, and rows represent the distance north/south.
 - The MSTs starting point locations are different than the real world's starting point locations.
 - Columns start at zero at the line (like a longitude) through Tokyo, Japan, with negative values to the west.
 - Rows start at zero which is the circle (like a latitude) somewhere in/near/at the South Pole. Positive to the north.
 - Distances in both cases are expressed as units, not degrees, with a unit being a single world file/tile (2048x2048m).
 - These distances are reflected in the file names of world file/tiles.
-

World file/tile naming format

- World file names represent the distance from the starting point. Each name reflects its location in the MSTs world.
 - (The naming of terrain tiles is not covered here. Never saw the need to figure it out.)
 - For world files, the first part in the name represents a column for east/west, the second part a row for north/south.
 - Thus world file "w-11069+14261" represents 11069 units west and 14261 units north from the starting points.
 - Because of this design, routes can be moved by changing world files names and updating data bases accordingly.
 - This design, and others that allow reshaping a route and adjusting altitude, is what makes merger projects feasible.
 - Tools that exploit these designs to implement changes to a route is what makes merger projects possible.
 - As will be seen, these are intertwined and are performed in a preparation phase before an actual merger of two routes.
-

Moving a route:

- A route can be moved by TSUtil MOVE which manipulates world file names and adjusts references to them in data bases.
 - It's like moving a puzzle piece next to another, but one or both may require some reshaping for them to fit properly.
 - The distance it moves is based upon two values that must be derived by the user.
 - A route can be moved in two directions (east/west and north/south) for a specified number of world files in each.
 - The specified values must be whole numbers representing the number of world file units (distance) to move, or zero.
 - As will be seen next, certain specific world file names will be used to derive the values to use with TSUtil MOVE.
-

Moving what to where

- It may make it easier to visualize a move by thinking of it as shooting an arrow to a target.
- Here, the terms "Arrow" and "Target" will be used to provide a mechanism to determine how far to move a route with MOVE.
- Arrow and Target can be determined by using the "JctWF" for each route.
- "JctWF" is the world file name representing the junction from which tracks will later be extended into the other route.
- "JctWF"s are selected during analysis of the routes using knowledge of the routes and RGE/AE/RE.
- Arrow is the "JctWF" of the route chosen to be moved.
- Target is adjacent to the "JctWF" of the other route, the open space to be occupied by the Arrow after a MOVE.
- The amount to move is derived by calculating the arithmetic differences in the two parts of two world file names.
- The difference between the first part of the Arrow and Target names determines the distance east or west.
- The difference between the second part of the Arrow and Target names determines the distance north or south.
- Positive values (with no sign) move east and north, negative values with a minus sign move west and south.

- Zero means no move in that direction.
- After providing the values to TSUtil, it shoots the Arrow to the Target, dragging the whole route with it.
- The objective is to get the routes to fit together as closely as possible with no overlap before a merge.
- An example of a move will illustrate.

An example of an unrestricted move:

- A route is to be moved a short distance east and south.
- All tiles are at the smallest edge size of one world file, thus it can be moved anywhere.
- World file -11057/14283 on the east side of the route to be moved is designated as its "JctWF", thus the Arrow.
- World file -11052/14270 on the west side of the route to be moved to is designated as its "JctWF".
- The open space adjacent to the "JctWF" of the route being moved to is -11053/14270, and designated as the Target.
- Thus the Arrow is -11057/14283 and the Target is -11053/14270.
- The value for the move east would be the difference between -11057 and -11053, which is 4.
- The value for the move south would be the difference between 14283 and 14270, which is 13. To move south, value is -13.
- Therefore the values of "4 -13" would move the whole route east and south to the desired Target.

You can't always move where you want

- If the largest tile in a route is one world tile on a side, the route can be moved anywhere in the MSTs world.
- But if the largest is larger than one world tile on a side, then the specified values must be a multiple of that size.
- For example, if the largest is a DM with eight world files on a side, the specified values must be a multiple of eight.
- Or if the builder mistakenly created a large tile at two world files on a side, then values must be multiple of two.
- If a restriction exists, it means that a route may end up near but not at the desired location.
- Default USA1 has a restriction. It can be easily removed with no harm because that area is not actually used.
- If a restriction can't be eliminated, then a gap is likely which will require filling in with more world files/tiles.
- This restriction is not a TSUtil deficiency. A large tile covers multiple world files, thus all must move as a unit.
- An example of a restricted move will illustrate the impact that large tiles have, and how to deal with it.

An example of a restricted move:

- A route is to be moved a short distance east and south, as above and with the same initial "JctWF"s.
- But this time, the route has DMs with the largest at 8 world files per side.
- Because of this, valid choices to move this route are restricted to 0, 8, 16, etc.
- The values "4 -13" from above are invalid and must be changed if want to keep the original Arrow or Target.
- If used "8" for the move east, Arrow would end up at -11048 (-11057 + 8 = -11049), east of the target route's "JctWF".
- Overlap will occur, thus not allowed. See more about overlap in the next section.
- Because of this overlap, the only choice for the move east is "0", which will cause a gap. See next section.
- The gap to fill would be 4 world file/tiles (the difference between -11057 (after MOVE) and original Target's -11053).
- If used "-8" for the move south, Arrow would end up at 14275 after MOVE. (14283 - 8 = 14275)
- This is north of the target route's "JctWF", which means no overlap, but now a gap to deal with.
- The gap to fill would be 5 world file/tiles (the difference between 14275 (after MOVE) and original Target's 14270).
- The proper values to use for this restricted MOVE are "0 -8" instead of the originally preferred "4 -13".
- In merger projects using routes with large tiles, reshaping of the route(s) is often required to deal with overlap/gap.

Overlap/Gap = reshape/move

- Overlap is when both routes have one or more tiles/world files with the same name, thus occupying same areas, a no-no.
- Overlap can be resolved before a merger by reshaping one or both routes via deletions, or by moving a route (via TSUtil).
- If overlapping tiles have world files with data base items, the data base items must be removed with one of the methods.
- Overlapping tiles can be deleted via RGE, FILTER, or both.
- Gap is when there would be empty space between routes, often created because of restricted moves.
- Gap can be resolved by moving and/or reshaping a route before or after a merger with added tiles to fill in the gap.
- Whenever reshaping is required, it often means a different set of Arrow/Target from initial choices.
- Ultimately, shape/Arrow/Target must be found that provides MOVE values that can achieve no overlap and minimum gap.
- One way to alleviate overlap/gap issues is to achieve a "straight edge" along the whole joining border, if possible.

A whole straight edge

- Sometimes routes can be reshaped to have a straight edge along the whole joining border where overlap could occur.
- Achieving a straight edge means removing all world file/tiles beyond the "JctWF" towards the other route.
- Many routes have excess tiles beyond their last tracks which can be easily removed with no discernable harm.

- If this can be achieved for both routes, then overlap can't occur - nothing would stick out into the other route.
- An example would be a route that is to be extended by another route, with both having excess tiles at their ends.
- If an eastern route's "JctWF" on the west side is at -12345/XXXXX, then all from -12346/XXXXX would be removed.
- If a northern route's "JctWF" on the south side is at -YYYYY/12345, then all from -YYYYY/12344 would be removed.
- Of course the same approach would apply to the other routes involved, but in the opposite directions.
- If straight edges can be achieved, the effort to reshape to prevent overlap can be significantly reduced.
- Otherwise, maybe a partial straight edge can be achieved near the "JctWF"s to help make things a little easier.

A partial straight edge

- It may not be feasible to have a straight edge along the whole border for both routes.
- This is when world files/tiles other than an "JctWF" might stick out beyond it, potentially into the other route.
- This is likely to happen when a route is to be merged into another at one or more "JctWF"s within it.
- Figuring things out and spotting/selecting/deleting tile/files in this case would be more difficult.
- Reshaping to at least a three world file straight edge with an "JctWF" in the middle might make it easier.
- Otherwise, individual tiles/world files will have to be dealt with on an individual basis, making things harder to do.

Regardless of border edge shape:

- Whenever reshaping includes the removal of tiles, any existing corresponding world files will have to be removed too.
- If so, then consider reshaping as a removal project. This means: pick one of the removal methods to use to remove areas.
- When lots of world files to remove, the CLRDB removal portion of the TSUtil CLRDB/MERGE may be best.
- When few to remove, the RE Delete method may be best.

Viewing, analyzing, and identifying

- In order to address overlap/gap, the builder has to be able to view the routes in displays, on paper, and in his mind.
- Unfortunately, RGE displays only one route, making it hard to determine where world files and tiles might overlap.
- To assist, one can create and use a small hand-drawn RGE-like matrix as a worksheet to mark pertinent info onto.
- It could be a box matrix consisting of smaller boxes with five along each side, one matrix for each route.
- Pertinent rows and column names from the RGE display and the "JctWF" can be marked onto the worksheet matrix.
- Tiles that can be deleted and other notes can be marked on it too.
- Thus the matrix worksheet could be a subset for the border area around the "JctWF" of the real RGE display.
- I found that such worksheets used in conjunction with RGE/RE helped to view/analyze/identify, but it's still not easy.
- In the following link there are some other explanations and techniques that might help when viewing routes.
- <http://www.trainsim.com/vbts/showthread.php?235096-Quad-Tree-file&highlight=quad+tree>
- The next section includes a way to view two routes in one RGE and one AE display, and identify any overlap.
- In fact, this is done automatically in one of the first steps of the approach described below.

When one has a sufficient understanding of the fundamentals, he is in a position to learn more about how to prepare a route for a merger, the hardest phase of merger projects.

Prepare A Route For A Merger

Preparing a route for a merger is doing the necessary tasks to ensure that each route will be in its proper location, shape, and altitude before the actual merger of files and folders of the two routes. To do so, a route may have to be reshaped, moved, and have its altitude adjusted. Any reshaping involving removals of world files with data base items in them will have to be done in conjunction with one of the methods in order to synchronize data bases for those world file changes. For help when using methods, tools, and functions described below, refer to the "how to"s in:

- Part 2 for the usual methods
- Part 3 for the TSUtil CLRDB/MERGE method
- The Appendix for RGE and the TSUtil and Route_Riter functions

Here is an approach for preparing a route for a merger, starting with an overview:

Overview

- Check to see if both routes use Mosaic files
 - Conduct a trial run of TSUtil MERGE
 - Conduct an initial analysis of the routes
 - Find out if a MOVE restriction exists
 - Conduct an unrestricted move, then skip the next two
 - Eliminate or reduce a restriction if possible
 - Conduct a restricted move
 - Adjust the altitude of a route if necessary
-

Check to see if both routes use Mosaic files

- Use these steps to see if there is a possible showstopper caused by the use of Mosaic files in both routes.
- Check the Terrtex folder of each route. Look for sequentially numbered files such as "mosaic_0001", "mosaic_0002", etc.
- If both routes contain such files, they will have same names but different content in most cases.
- If so, terrtex files from the primary route will prevail. Their use in the secondary portion may not appear correctly.
- No need to continue until/unless resolved, unless you want to address the issue after the actual merger.
- Otherwise, proceed to "Conduct a trial run of TSUtil MERGE", next.

Conduct a trial run of TSUtil MERGE

- Use these steps to see if there is a showstopper caused by data bases that won't merge. No use continuing if so.
- Back up the routes, and have Route_Riter at its most current level, at least 7.6.18 (26 Aug 2011).
- Run MERGE. The only purpose is to view results. Either route can be primary. Don't worry about preparation.
- Check the bottom of the Report Form. It may indicate a need to "Horace", some other problem, or program completion.
- If need to "Horace", do so and repeat trial run. Use TSUtil CVRT, global swapping, or Horace as desired.
- If some kind of a problem, refer to "When Things Go Wrong" and repeat above if fixed. You may or may not be successful.
- After successful MERGE completion, copy the contents of "newRoute" folder back into the route's main folder.
- Load the merged route into AE to see if the data bases load okay. Activity related errors can be ignored.
- If doesn't load, data base errors exist in one or both routes that prevent successful processing by TSUtil.
- If fails to load, try again after restoring the primary and using original secondary as primary, or troubleshoot.
- When a merged route loads in AE, this means that it is worthwhile to continue, though there still may be issues.
- You can now view two routes in a single RGE or AE display. Also, the merge log contains a list of any overlaps.
- Refer to "How to get two routes in one RGE or AE display instead of just one, and find overlap" below.
- With RGE and AE displays and worksheets available, an initial analysis can be done.
- Proceed to "Conduct an initial analysis of the routes", next.

Conduct an initial analysis of the routes

- Use these steps to get a quick indication of what to expect in terms of degree of effort and difficulty.
- View and use RGE and AE displays and any worksheets to determine the relative locations of the two routes.
- If a gap, reshaping may be necessary to fill the gap, and/or a route may have to be moved.
- If no or slight overlap, maybe some trimming along the borders will suffice, and not a move.
- If more extensive overlap, maybe it can be removed via reshaping without a move, especially if overlap is excess tiles.
- Once you decide you are done with the initial analysis of the merged route, restore the primary.
- You want to ensure that both originals are in place before doing analysis and reshaping on them which follow.
- Now you can use the RGE/AE displays of the original routes and worksheets from the trial merge for analysis.
- If you decide a move is not necessary, reshape as necessary and proceed to "Adjust the altitude of a route..." below.
- Otherwise a move is necessary.
- Reminder: The terms "JctWF", "Arrow", and "Target" are used below. They are defined in "Moving What To Where" above.
- Proceed to "Find out if a MOVE restriction exists", next.

Find out if a MOVE restriction exists

- Use these steps to see if a move restriction exists in a route to be moved, thus determining which type of move to do.
- Before proceeding, be sure that both original routes are in the Routes folder, and not the one from the trial merger.
- Pick a route to be moved.
- If you already know the route to move has a restriction, skip to "Eliminate or reduce a restriction if possible".
- If you already know the route to move has no restriction, skip to "Conduct an unrestricted move" below.
- Otherwise, use RGE or a trial move to determine if there is a large tile that forces a move restriction, or not.
- In RGE, look for larger "X"s including DMs. If see any, proceed to "Eliminate or reduce a restriction if possible".
- Otherwise, if no larger "X" in RGE, proceed to "Conduct an unrestricted move" below.
- If don't want to use RGE, do a trial run of MOVE. It will quickly and easily disclose a restriction if any.
- You can do this by doing a MOVE using "1 1" as values. The only purpose is to see the Report Form.
- Check the bottom of the Report Form. It will report the route moved, a move restriction, or a problem.
- If a problem, see if you can fix it and repeat the trial MOVE above.
- If a restriction, write down the multiplier value, exit with "No", and proceed to "Eliminate or Reduce..." below.
- Do not respond "Yes" to Route_Riter's offer to copy "newRoute" into the route. If do, restore the route after MOVE.
- If moved (thus no restriction), exit with "No" and proceed to "Conduct an unrestricted move", next.

Conduct an unrestricted move

- Use these steps to move a route when it has no move restriction. It can be moved anywhere. Reshaping may be desirable.
- Analyze each route to choose its "JctWF" and the tiles/world files to delete/add for reshaping if necessary.
- This is where RGE/AE displays and worksheets (single route or both routes) are used to conduct the analysis.
- To help identify a JctWF", latitude/longitude in AE can be used with RE or Route_Riter to find its world file name.
- During analysis, see if able to achieve a straight edge along the whole border towards the other route.
- Reshape the route accordingly. Use one of the removal methods when deleting world files containing data base items.
- If you have little experience with RGE, refer to "How to use RGE to reshape a route" in the Appendix.
- Designate the "JctWF" of the picked route as the Arrow.
- Use the "JctWF" of the target route to derive Target. Target is an open space next to the target route's "JctWF".
- Derive the values for MOVE by getting the east/west and north/south arithmetic differences between Arrow and Target.
- Run MOVE using the derived values. Check bottom of Report Form.
- The move should complete without restriction. If it has one now, this means one was created during reshaping. Fix it.
- Respond "YES" to let Route_Riter copy "newRoute" into the route assuming successful completion of MOVE.
- That's it. The moved route should be at its proper shape/location a for merger, but may need an altitude adjustment.
- Proceed to "Adjust the altitude of a route if necessary".

Eliminate or reduce a restriction if possible

- Use these steps to see if you can eliminate the restriction value to make subsequent steps much easier, or to reduce it.
- If you encounter a restriction, consider picking the other route as the one to move if it has no restriction.
- If you decide to pick a route to move that has no restriction, proceed to "If no MOVE restriction" above.
- If you already know that the move restriction can't be eliminated or reduced, skip to "Conduct a restricted move" below.
- See if DMs exist by loading its quad-tree in RGE, and assess if they overlap with the other route.
- If DMs exist and overlap, then reshaping or complete removal may be necessary
- To delete all, use RGE or just delete the LO_Tiles folder and the "lo_td_idx.dat" and "*.tdl" files in the TD folder.
- Otherwise analyze the route to see if you can remove or reduce the restriction by reshaping.
- If can, reshape the route accordingly, using one of the removal methods as appropriate for world files.
- Always reshape to the smallest size possible.
- If you were able to eliminate all large tiles by reshaping, proceed to "Conduct an unrestricted move" above.
- Otherwise you still have a restriction, but you need to know its size.
- If you were unable to reduce the size of the restriction, it should be the same as you noted during the trial move above.
- If you were able to reduce it, you should know the value for the current largest tile after your changes.
- If you don't know that value, don't proceed until finding out what it is via RGE or another trial run of MOVE.
- Proceed to "Conduct a restricted move" when you know the restriction value, next.

Conduct a restricted move

- Use these steps to move a route that has a restriction that can't be eliminated.
- The distance to move has to be a multiple (divisible by) the restriction value determined from RGE or a trial move.
- Analyze both routes to pick an initial pair of "JctWF"s, one for each route.
- "JctWF" is the world file name representing the junction from which tracks will later be extended into the other route.
- This is where RGE/AE displays and worksheets (single route or both routes) are used to conduct the analysis.
- During analysis, see if able to achieve a straight edge along the whole border towards the other route.
- To help identify a JctWF", latitude/longitude in AE can be used with RE or Route_Riter to find its world file name.
- Do not do any reshaping now. You may need some tiles for filler in case a gap is forced by a restriction.
- Designate the "JctWF" of the route to be moved as the Arrow.
- Use the "JctWF" of the target route to derive the Target. Target should be open space next to the target route's "JctWF".
- Derive the values for MOVE by getting the east/west and north/south arithmetic differences between Arrow and Target.
- Check to see if the values are divisible by restriction size. See "An example of a restricted Move" above if necessary.
- If not, recycle through the analysis process to find a Target or other "JctWF"s that will satisfy the restriction.
- Reshape the route(s) accordingly. Use one of the removal methods when deleting world files containing data base items.
- Run MOVE using the derived values. Check bottom of Report Form.
- The move should complete without restriction. If one is reported, this means you need to recycle thru analysis again.
- Respond "YES" to let Route_Riter copy "newRoute" into the route assuming successful completion of MOVE.
- That's it. The moved route should be at its proper shape/location a for merger, but may need an altitude adjustment.
- Proceed to "Adjust the altitude of a route if necessary", next.

Adjust the altitude of a route if necessary

- Use these steps to adjust the altitude of a route to match the other at the border.
- Plant a scenery object in the "JctWF" near the border towards the other route.
- Note the "Y" value in the RE Object for the selected object. It is the altitude at that spot.
- If you use a track for this, its "Y" may be at the connected end, not the free end. Makes a difference if on a grade.
- Delete the object.
- Repeat the process above for the other route to get its altitude near the "JctWF" border.
- The arithmetic difference between the two "Y" values provides the difference in altitude in meters at the border.
- If don't need to adjust either route's altitude, proceed to "Prepare the other route if necessary" below.
- Pick the route whose altitude you want to adjust.
- Run TSUtil ADJH using the nearest whole number of meters from the "difference" determined above, up or down.
- Respond "YES" to let Route_Riter copy "newRoute" into the route assuming successful completion of ADJH.

Prepare the other route if necessary

- At this point one or both routes should be prepared to be in proper shape, location, and altitude.
- If both not yet prepared, perform the appropriate sections above for the route needing it.
- That's it for the preparation, ready for merger.

Perform The Merger

Performing the actual merger is using one of the methods described in this document to actually merge data, files, and folders from one route into the other while synchronizing data bases with the changes being made. The merger could be for copied world files/tiles, or for both whole routes.

To perform the merger

- Decide which method you want to use for the actual merger, one of the usual methods or TSUtil CLRDB/MERGE.
- To merge part of a route into another via RE Rebuild or RE Select, refer to the "how to"s in the "Usual.." section above.
- To merge two whole routes, refer to "MERGE" in the "How To Use TSUtil And Route_Riter Functions" in the Appendix.
- That's it for the merger, ready for completion of project.

=====

Complete The Project

Completing the project is performing the final tasks to join the routes into one.

To repeat from a previous section:

- Provide scenery, textures, and other files from one route into the other if not already present, including .ref file.
- Address possible differences in .dat files, signals, etc. that may require manual merging or editing.
- Join tracks, add scenery, and adjust terrain at the borders to get a smooth transition from one route to the other.
- Fix any conflicts caused by shapes/textures and terrtex files having same names but different content.
- Repair or replace activities that may have been affected by track changes in world files.
- That's it.

Closing commentary

Results from projects like these can span from complete success, success with sacrifices, or complete failure.

- How a project will turn out depends upon many items, including interactions between builder, routes, tools, and methods.
- If one lacks skill/experience in building routes, such projects shouldn't be started until after gaining some.
- If one has that skill/experience but not in these methods, maybe this document will help one get started.
- If one has skill/experience in such projects but not all methods, maybe this document will provide new option(s).

As said before, this is FWIW and is just one builder's version about these kinds of projects and methods.

- There is lots of stuff along with lots of opportunities for errors, omissions, oversimplifications, and disagreements.
- If one is spotted, replace it with your own knowledge/skill/common sense. Serious ones may prompt revisions to this.
- If the TSUtil CLRDB/MERGE method works for you, great.
- If not, hopefully the other stuff will.
- Good luck.

That's it, except for some recognition of others, and the Appendix which follow.

- Thanks to all past/present contributors to the Route Design forum. They keep the route building engine alive and well.
- Thanks to Carlos-Heinz for TSUtil and to Mike Simpson for Route_Riter.
- Their remarkable contributions are truly indispensable to the route building community.

THE APPENDIX

Appendix

The appendix provides a much deeper level of detail on how to use the tools and functions in merger and removal projects.

Appendix topics are:

- Why/When/How To Convert (Horace) A Route
- How To Get Two Routes In One RGE Or AE Display And Find Overlap
- How To Use RGE To Reshape A Route
- How To Use TSUtil And Route_Riter In Merger/Removal Projects

Why/When/How To Convert (Horace) A Route

A merger project increases the likelihood that there may be the issue of incompatible dynamic track ID numbering systems to deal with. This is because routes typically considered for merger were built by different builders using different MSTs systems. If the issue exists but not properly addressed, the routes cannot be merged by normal means. Following is an explanation of what this is all about, and how to deal with it.

Track and road numbering systems

- The ID numbering system used for regular tracks and roads is different from the numbering system used for dynamic tracks.
- This was not an issue in the early days when the only tracks and roads were defaults from MSTs.
- The advent of extended tracks and roads such as XTracks/Newroads/others introduced a greatly extended numbering system.
- This introduced the potential for incompatibility between numbering systems for regular tracks and dynamic tracks.

Regular tracks

- There are two numbering systems for regular tracks/roads - default from MSTs and Standardized from XTracks, etc.
- Default tracks and roads use a pair of IDs that range from 1/1 to 375/262.
- Standardized tracks and roads use a pair of IDs that range from 1/1 to 39999/39999.
- The Standardized includes default regular tracks using the same values, and goes up from there for the extended ones.
- A numbering system master file and the related shape files are stored in the MSTs Global folder.
- The master file for regular tracks/roads is known as the Global tsection.dat file. It is used only as a reference by RE.
- RE refers to the current contents of Global as it creates entries for the TDB and world files as tracks/roads are laid.
- The Global tsection.dat is either "default" or "Standardized", depending upon which Global the builder provides.

Dynamic tracks

- Dynamic tracks have their own numbering system, and their range begins where the range for regular tracks/roads ends.
- Dynamic tracks initially laid in a default Global system are permanently numbered in a range starting at 376/263.
- Dynamic tracks initially laid in a Standardized Global system are permanently numbered starting at 40000/40000.
- These numbers and associated data are stored in a tsection.dat file in the routes main folder. There are no shape files.
- This controlling file for dynamic tracks is known as the Local tsection.dat. It also contains all pertinent shape data.
- The IDs and entries in the Local are unique to the one route, with values derived as the builder lays each track.
- RE is wholly responsible for the Local tsection.dat (shudder). It creates the Local, and adds/modifies/deletes entries.
- RE assigns IDs as each dynamic track is laid, using the next available values above previous ones found in the Local.
- RE also refers to the Local when dynamic tracks are modified or deleted. IDs are not reassigned after deletions.
- The Local tsection.dat is either "default" or "Standardized" depending upon which Global was being used when initialized.

Note: the IDs for all tracks and roads appear in and are crosslinked among the Global tsection.dat file and the route's Local tsection.dat file, the TDB, and world files. Proper synchronization is absolutely essential, otherwise S/DS messages, failure to load, or other weird stuff will occur.

The problem

- The problem is corruption that can occur when the range of numbers in the Local is incompatible with the range in Global.
- A range starting at 376/263 for dynamic tracks would overlap inside the range reserved for Standardized regular items.
- For that matter, a Local starting at 40000/40000 would be incompatible with a default Global that ends at 375/262.
- This is not a problem when a route with dynamics was initially built and remains under the same type of Global.
- Also it is not a problem when such a route is used only in the Sim and AE, regardless of the Global.
- But a problem will exist when an action occurs with dynamic tracks in RE/TSUtil when Global and Local are incompatible.

- When a default Local is under a Standardized Global, the route can be converted to put its dynamics above the Global.
- This can be done by adding 39624/39737 to each dynamic track ID in every location it is stored.
- This will get them into a range starting at 40000/40000, above the Standardized Global range that ends at 39999/39999.
- That is what the utility known as Horace does to convert a route, and so does TSUtil CVRT.

How do I know what I have?

- Use Explorer to view the MSTS Global folder and look for the tsection.dat file.
- If the size is about 156KB, the Global is default.
- If the size is much larger than that, the Global is Standardized.
- Use an editor to open the local tsection.dat file in the route's main folder.
- If you don't find a tsection.dat, the route does not have dynamic tracks. No action is required, you can skip the rest.
- If you see number sequences near the top starting with "376", the Local is default.
- This means that the route's dynamic tracks were initially laid while under a default Global. It may need conversion.
- If you see number sequences near the top starting with "40000", the Local is Standardized.
- This means that the route's dynamic tracks were initially laid while in a Standardized Global, or already converted.

What do I do?

- Check to see which combination you have
- If the Global is default and Local is default, no action is required. There is no conflict or mismatch.
- If the Global is Standardized and the Local is Standardized, no action is required. There is no conflict or mismatch.
- If the Global is default and the Local is Standardized, you need to install one of the required extended track systems.
- If the Global is Standardized and the Local is default, you need to convert the route, or you could swap Globals.

How do I convert a route?

- You can use TSUtil CVRT. Refer to it in "How to use the TSUtil and Route_Riter functions" in the Appendix.
- Or you could use Horace from "tk_utils.zip" or "hrc11cab.zip" from the trainsim.com file library. Not covered here.
- In both cases, you will have to navigate to a default Global folder somewhere and select the tsection.dat file from it.
- That's it.

How do I swap Globals?

- If you use a Standardized Global but don't want to convert a default route with dynamics, you could do a temporary swap.
- Instead of converting, you can merely swap Globals so that the desired one is named Global.
- Rename Global (if Standardized) to GlobalS.
- Find a default Global folder somewhere and copy it and its subfolders into MSTS.
- When you are completely done, you can delete Global (or rename as GlobalD), and rename GlobalS back to Global.
- That's it.

Why can't I convert a particular route?

Sometimes routes having dynamic tracks may have a mix of numbering systems resulting from manual manipulation of data and files by the builder.

- The Global may be Standardized and the Local may be default.
- The TDB and world files would contain mixed ID numbers too.
- If so, Horace won't convert correctly. It considers anything above 375/262 as a dynamic, it thus would bump extended too.
- With TSUtil CVRT, it stops processing and issues a warning when it detects a mix.
- If such a route can't be converted, then dynamic tracks can't be used within RE, and TSUtil functions can't be used.
- Otherwise such mixed routes can be used in the Sim and AE. It can also be used in RE for working on regular tracks.

How To Get Two Routes In One RGE Or AE Display And Find Overlap

How To Get Two Routes In One RGE Or AE Display And Find Overlap

- Normally, a builder uses RGE and AE displays to view the shape, location, and track layout of each route.
- Unfortunately, each display contains only one route, making it difficult to visualize their relative position/layout.
- It would be nice to view both together to see how close they are to each other, and find any overlap.
- If you want to do this as an aid for working on the actual routes to be merged, it's easy as follows.

Conduct an immediate trial MERGE

- Immediately merge the two routes. Either one can be primary. No preparation is necessary except possible CVRT/Horace.
- This would be the trial merge if you are following the steps in the "Prepare A Route For A Merger" section above.
- You aren't going to use the merged route, just some displays from the route and some information from MERGE itself.
- So, just run MERGE, but you may want to take a look at "Some hints/tips" below before doing the trial merge.
- If the newly merged route loads okay in AE or RE, then it and the log from MERGE can be put to good use.

Decide if the merged route will be of any value

- Use the RGE/AE displays of the merged route to see if routes are so far apart that displays from it would be useless.
- If so, you could move a route as described in ("Some hints/tips" below) so that it would be closer.
- Or you may have to delete the merged route and rely on the normal displays of single routes instead.
- If the routes are in acceptable locations relative to each other, continue in order to make use of the merged route.
- Remember, the purpose of the merged route is the information from it as an aid in dealing with the two separate routes.

Create an RGE worksheet

- A printed worksheet of the RGE display of the merged route can be used as an aid during analysis/reshaping efforts.
- It will show the tiles of each route relative to the other.
- If you want a worksheet, display the merged route in RGE and then capture the display with "AltPrtScr".
- Paste the image into PaintShop Pro or equivalent.
- Optional, crop to the populated tiles, with some extra space around the edges to allow room for annotations.
- Optional, replace the RGE background colors with white for more clarity.
- Print it for use as a worksheet.
- Don't forget, the route may span across multiple RGE regions. If so, a worksheet should be created for each region.

Annotate the RGE worksheet(s)

- Transpose a few column/row names from the RGE display onto the corresponding positions in the RGE worksheet.
- As you add additional items of information, you can add more column/row headings as needed by counting and marking.
- You can mark "JctWF"s and tiles to remove on it, and mark tiles that overlap as shown next.

Identify overlapping tiles

- Open the log from MERGE that is located in Route_Riter's Reports folder. It is the most recent "**_merge.log" file.
- Search inside it for "both routes".
- At the hit, there's a list of overlapping tiles if any exist. Those in the secondary route are excluded from the merge.
- If no overlap, you'll see "0 valid Tile-Definitions of 'Merge'-route will not be included in target-route".
- If it appears, you're fortunate, skip to "Create an AE worksheet".
- If not, each entry means a tile appears in both routes. A corresponding world file may or may not exist in one or both.
- Fortunately for each overlapping tile, its corresponding world file name is listed, not the tile name.
- This makes it much easier to track down overlap and decide which to keep or remove by creating a worksheet from the list.

Create and annotate an overlap worksheet

- A printed worksheet of overlapping world file/tiles can be used as an aid during analysis/reshaping efforts.
- Select/copy the list paste it into a new text page, and print it.
- In the blank area to the right, make two column headings using your route names.
- If lots of lines, you may want to use the list while looking at the ORIGINAL routes to see if most are unused tiles.
- Many routes have excess unused tiles at the extremities that can be easily deleted in chunks to get a straight edge.
- The world file names provide RGE display rows and columns to spot if they are in or outside the desired area.
- If so, you can use a highlighter to flag them as excess tiles with no world files, or leave unmarked to flag as excess.
- For each of the other lines, look in each ORIGINAL route (not the merged one) for the world file name in World folder.
- Add the indicated markings under the appropriate column heading as instructed below.
- If there is no world file, mark as "-" to flag as no world file to worry about.
- If there is a world file but it's empty, mark as "E" to flag as empty, harmless, but wasting space.
- If there is a world file with items in it, mark as "P" to flag as a tile with a potential data base overlap.
- Do this for both routes so that there are markings under each column for each line that is not considered excess.

Use the overlap worksheet

- The worksheet now contains a list of overlapping tiles with markings indicating if their world files are factors or not.
- And it also contains a list of overlapping excess tiles having no world files, those you highlighted or left unmarked.
- During analysis of original routes, you can decide which tile/world file to delete from which route, and circle it.
- If you want, you can locate and highlight overlapping tiles from the list onto the RGE worksheet.
- When doing actual reshaping, the circled items and those indicated as excess are items that can/should be deleted.

Create an AE worksheet

- A printed worksheet of the AE display of the merged route can be used as an aid during analysis/reshaping efforts.
- It will show the track layout of each route relative to the other.
- If you want a worksheet, display the merged route in AE and then capture the display with "AltPrtScr".
- Paste the image into PaintShop Pro or equivalent.
- Print it. There is no need for color replacement nor cropping unless you want to.

Annotate the AE worksheet

- Use AE's latitude/longitude and RE's camera to get world file names for various landmarks for each route.
- Landmarks can include "JctWF", route extremities, and other locations that might help visualize. Do this for each route.
- Annotate them on the AE worksheet at appropriate locations.
- The worksheet can be used to do a rough drawing of tracks onto the RGE worksheet for the border areas.
- If you have Mosaic, you can display the tiles and tracks of the merged route in the same display,

With the worksheets:

- Use these worksheets of the trial merged route as you analyze and reshape the originals via RGE and the methods.
- Worksheets from a trial merge don't necessarily make the reshaping process easy, but they may make it easier.
- If you are doing the trial merge from "Prepare A Route For A Merger", return to that section.
- Otherwise, be sure to delete the merged route or move it elsewhere, and restore the primary when done with it.
- You want to ensure that both original routes are in place before doing actual analysis and reshaping on them.
- That's it.

Some hints/tips:

- If you already know the routes are too far apart, you may move one closer to the other, but no need for precision.
- Precision will come when you work with RGE/AE displays of the original routes, with or without these worksheets.
- So, with this move, aim to end up with a small or no gap between them. You may have to deal with a move restriction.
- If doing a move, select a world file in the middle of populated boxes along the border towards other route as "JctWF".
- If not satisfied with the move because of too much overlap or gap, you can do additional moves/merges until satisfied.
- Be aware that if you do a move to get a trial merged route, complete the actual move and use this version from now on.
- Otherwise the column/row references in the display and worksheets of the merged route would be erroneous with originals.

How To Use RGE To Reshape A Route

How To Use RGE To Reshape A Route

RGE can be used to delete or add tiles when reshaping a route during the preparation phases of removal and merger projects. Demex has some tile manipulation capabilities too, but it is not addressed here because all would have RGE. RGE can be used to delete a single tile via cursor selection, or a few or a huge amount at a time via box selection. RGE and AE displays and/or worksheets from them can be used to help keep track of areas needing attention.

To Delete A Single Tile:

- Perform the normal steps in RGE to select the route, load quad-tree, and zoom the region/windows.
- Locate the desired tile by referring to world file name at bottom of display window while moving cursor around.
- Numeric values of world file names will change as cursor is moved, giving a clue as to location.
- Find the unwanted tile and place the cursor over the tile then right click and delete. Files are deleted then.
- Repeat as necessary.
- Save Quad-tree along the way or when done. This updates the structure. The tile(s) are already gone.

To Delete A Group Of Tiles:

- A small or big chunk of tiles can be deleted at a time, and repeated for other areas until all unwanted ones are gone.
- This is done by using a cursor drawn selection box that starts at the upper left and ends at the lower right.
- After the box is drawn, place the cursor inside it then right click and Delete.
- Sometimes it's difficult to line up the selection box so that an edge is exactly between wanted and unwanted squares.
- If so, see next section.
- Repeat selections and deletions as necessary.
- Save along the way or when done.

To Make It Easier To See The Borders:

- Delete single unwanted tiles immediately adjacent to wanted ones to open up space for easier box selections later.
- Or could delete small groups at a time until sufficient space opens up.
- Then hack away at the remaining unwanted areas with large selection boxes until done.

To Add Tiles:

- Perform the normal steps in RGE to select the route, load quad-tree, and zoom the region/windows.
- Set the altitude for the tiles to be added by Edit/Set Height Offset. Enter desired value -1.
- Locate the area in which tiles are to be added. It should be unpopulated, no "X"s.
- Use "Split" to create the smallest possible squares as necessary.
- Place cursor in a single unpopulated square, or inside a cursor drawn box of unpopulated squares, and right click.
- Select "Toggle Populated State", which places an "X" in the square to flag it, or remove it.
- When finished with selections, generate and store the new tiles by Edit/Generate Flagged Tiles.
- When finished, save the Quad-tree by File/Save Quad-tree.
- Repeat as necessary.

How To Use TSUtil And Route_Riter In Merger And Removal Projects

How To Use TSUtil And Route_Riter In Merger And Removal Projects

- Route_Riter includes TSUtil, and both provide extremely powerful functions useful in merger and removal projects.
 - All of their functions covered in this document can be performed via some simple selections and button clicks,
 - Certain actions before/during/after using them must be performed in the proper manner to achieve successful results.
 - The proper manner is described next.
-

There is a right way to install Route_Riter/TSUtil

- All instructions in this document are based upon Route_Riter Version 7.6.18 (26 Aug 2011) or later.
- Certain instructions would be incorrect with prior versions. The capabilities are there, just used differently.
- Install the latest version of Route_Riter from <http://www.rstools.info>
- Also install the latest 32 bit version of java from <http://www.java.com>. Current at this time is java release 6 (jre6).
- Refer to <http://www.rstools.info/RRInstaller.pdf> for instructions on how to install Route_Riter.
- Refer to <http://www.rstools.info/faq.html> if you get error messages or have other difficulties.
- Be sure to create a "Classes" folder under "c:\Program Files\Java\jre6\" or "c:\Program Files (x86) \Java\jre6\").
- Then copy the contents of Route_Riter\TSUtil into the "Classes" folder. Route_Riter and TSUtil run better this way.

After every installation and future update, always be sure to copy the contents of the TSUtil folder into "Classes".

Note: If you are on a 64 bit version of Windows 7 and get a "Java runtime error..." message, you might try this:

- Download/install the latest 64 bit version of java at <http://www.java.com>. Both 32 bit and 64 bit should be installed.
- Follow the previous instructions above to do the same for the 64 bit version.
- There may be other issues unique to Windows 7. Check the above links and the trainsim.com Route Design forum.

There are two basic ways to use TSUtil

- One way is as a standalone program via a java command line in a "cmd" window. It provides the most control and options.
- The other way is via a user interface (UI) provided by Route_Riter. UI is easiest to use.
- The Route_Riter/TSUtil UI approach is covered here. The java command line approach is not.

There are three ways to use TSUtil via Route_Riter UI

- Most users use the UI provided in the TSUtil tab. It is the easiest way. via buttons. The defaults will suffice here.
- Another way is yellow button at bottom left. This provides a UI for entering command line parameters/options directly.
- Another way is yellow button at bottom right. This provides a UI for entering command line parameter/options indirectly.
- The first way with the function buttons is covered in this document, not the yellow buttons at the bottom.

Some aspects are common for all functions

- All the TSUtil function buttons covered here are in the TSUtil tab of Route_Riter.
- The upper left area is used to navigate to and double click route, always to be followed by clicking the Confirm button.
- After clicking a function button, processing is indicated via scrolling, hour glass, or a status line at the bottom.
- Upon completion of processing, a Report Form is displayed. At the bottom is a "Program ended" message, or error message.
- This report is stored as a .log file in the Reports folder of Route_Riter.
- By sorting the contents of that folder on "Date modified", the latest is brought to the top.
- Output files and folders from all functions are placed in a newly created "newRoute" folder within the route.
- The contents of the "newRoute" folder have to be copied back into the route for changes to take effect.
- As seen next, Route_Riter issues a message regarding "newRoute", and the response is slightly different by function.

Some aspects differ by function

- FILTER differs from other functions in the handling of its output. This is described in its specific section below.
- For ADJH, CVRT, and MOVE, there is a choice of Route_Riter doing the copy, or the user. Response is "Yes" or "No".
- The suggested user response is to use "Yes" if "Report Form" indicates successful function completion.
- For FILTER, CLRDB, and MERGE. there is a notice of the need for the user to do the copy. The only response is "Ok".
- For CLRDB, certain folders should be deleted before the user copies "newRoute" after successful function completion.
- For FILTER and MERGE, the user should copy "newRoute" when ready after successful function completion.

Note: The aspects above are reflected in the corresponding instructions below.

How To Use The TSUtil And Route_Riter Functions

How To Use The TSUtil And Route_Riter Functions

- These are the TSUtil and Route_Riter functions used in the TSUtil CLRDB/MERGE method.
- They are in the order in which they normally are performed.
- Refer to the "When Things Go Wrong" section below if you have difficulties using any of them.

TSUtil CVRT

- Use this to convert (Horace) a route.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Run CVRT by clicking the "Modify Route For New tsection.dat (cvrt)" button.
- You will have to browse to a default tsection.dat in a default Global somewhere in your system when asked.
- After selection, you should see scrolling as processing occurs.
- Wait until a Report Form is generated, then look at the bottom for "Program ended..."
- If you didn't get "Program ended...", you navigated incorrectly or incorrectly determined the need to "Horace".
- If so, verify that your Global tsection.dat is "Standardized" version, and that the route is "default", and try again.
- Otherwise, if program ended, exit and respond "Yes" to the message about a choice for "newRoute" appears.
- The contents of "newRoute" are automatically copied back into the route's main folder.
- That's it. The route has been converted for use under the Standardized Global.
- An alternative is to use Horace in "tk_utils.zip" or "hrc11cab.zip" from the trainsim.com file library.

TSUtil FILTER

Be aware

- TSUtil FILTER issues a unique warning message after clicking its button but before processing begins.
- It suggests running ICHK to see if the route needs to be converted (Horaced) and to use CVRT if so.
- Heed this warning to convert the route if necessary. See "Why/When/How to convert (Horace) a route" if necessary.
- The message also warns of the possible effect removal of tiles would have on terrain on outlying water/desert/hills.
- An informative tutorial about this is at http://msts.steam4me.net/tutorials/remove_tiles.html.
- If missing terrain will be a problem, then use RGE to remove tiles, or use FILTER in conjunction with RGE.
- Otherwise, if you want to take advantage of FILTER to remove tiles automatically, continue.

FILTER:

- Use this to delete regular tiles having no world files. Distant Mountains are not affected by FILTER.
- At this point, all unwanted world files should be gone, with only wanted ones remaining in the World folder.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Run FILTER by clicking the "Remove .t where no .w exists" button. This is the button for FILTER.
- The warning message will appear about ICHK, CVRT, and terrain.
- Click the "OK" button on a warning message, assuming the route has been converted if necessary, and no terrain issues.
- Wait until a Report Form is generated, then look at the bottom for "Program ended..."
- If it doesn't appear or a problem is indicated, proceed to the "When Things Go Wrong..." section.
- Exit and respond "OK" when a message appears about "newRoute" after exiting the form.
- Tiles are removed from the route's Tiles folder during processing. A TD folder with a changed quad-tree is in "newRoute".
- Copy the contents of the "newRoute" folder back into the route's main folder if FILTER completed successfully.
- Note: The files for the removed tiles are in the Reports folder of Route_Riter, as backup.
- That's it. Tiles for no longer wanted world files have been removed.

TSUtil MOVE

- Use this to move a route.
- At this point, the two values to specify for the move east/west and north/south should already be known.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Run Move by clicking the "Move a Route to new Lat/Long (MOVE)" button.
- Enter the two derived values for east/west and north/south. Use zero if not to move in that direction.
- You should see scrolling and/or an hourglass as processing occurs. If not, or for short time, there may be errors.
- Wait until a Report Form is generated, then look at the bottom for "Program ended..."
- If it doesn't appear or a problem is indicated, proceed to the "When Things Go Wrong..." section.
- Exit and respond "Yes" when a message appears about a choice for "newRoute" after exiting the form.
- The contents of "newRoute" are automatically copied back into the route's main folder.
- Note: When a route is moved, the marker file no longer applies, the changes of locations are not reflected in it.
- That's it. The route has been moved.

TSUtil ADJH

- Use to adjust the altitude of a route.
- At this point, the value for the number of meters up or down should already be known.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Run ADJH by clicking the "Change Route Altitude (ADJH)" button.
- Enter the derived value to use for adjustment up or down. The altitude for everything in the route will be adjusted.
- You should see scrolling and/or an hourglass as processing occurs. If not, or for short time, there may be errors.
- Wait until a Report Form is generated, then look at the bottom for "Program ended..."
- If it doesn't appear or a problem is indicated, proceed to the "When Things Go Wrong..." section.
- Exit and respond "Yes" when a message appears about a choice for "newRoute" after exiting the form.
- The contents of "newRoute" are automatically copied back into the route's main folder.
- That's it. The altitude has been adjusted.

TSUtil CLRDB

- Use this to reorganize (synchronize) the track and road data bases.
- At this point, all border crossing data base items between wanted/unwanted world files should already been deleted.
- And normally the route should already be in its proper location/shape/altitude with only wanted tiles/world files.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Run CLRDB by clicking the "Reorg .tdb/.rdb (CLRDB)" button.
- You should see scrolling and/or an hourglass as processing occurs. If not, or for short time, there may be errors.
- With CLRDB, you will get a "newRoute" notice before Report Form is displayed. Respond "Ok".
- After Report Form is displayed, look at the bottom for "Program ended..."
- If it doesn't appear or a problem is indicated, proceed to the "When Things Go Wrong..." section.
- Delete the contents of the route's Activities/Paths/Services/Traffic and world folders.
- Copy the contents of the "newRoute" folder back into the route's main folder if CLRDB completed successfully.
- That's it. The interactive/track/road data bases have been reorganized to match only the existing world files.

Be aware:

- CLRDB does not produce empty TIT and/or RIT files if the route had interactives but the world files given CLRDB don't.
- This is a highly unlikely event in most situations and should not discourage the use of CLRDB.
- This could occur if so many world files were deleted that the remaining ones have no interactives of either kind.
- The issue is that the old TIT/RIT files will still exist after copying "newRoute".
- They cause no harm but their size would be misleading when analyzing results. There is an easy fix.
- Just select any track or road section in RE and Save. RE will rewrite them as empty if none exist.
- Also, this action serves as an early test to see if there is any serious problem that would cause a S/DS.

Also be aware:

- CLRDB does not produce an empty RDB file if the route had roads but world files given CLRDB don't.
- This is a highly unlikely event in most situations and should not discourage the use of CLRDB.
- This probably means that so many world files were deleted before using CLRDB that the remaining ones have no roads.
- The old RDB will still exist after copying "newRoute".
- This does cause a problem because as far as RE/Sim are concerned, road paths will remain though visual sections won't.
- One resolution is to plant a road section in a wanted world file before CLRDB, then delete or keep it after CLRDB.
- If at least one road exists in the world files given CLRDB, no problem because CLRDB will create a new RDB in "newRoute".
- Otherwise resolution will require knowing if any exist in the new route or not.
- If none, then you must remove the old RDB before adding new roads. Keep it elsewhere as backup in case you were wrong.
- The same issue would apply to the TDB too if no tracks appear in the wanted world files. Such a route would not load.

Of course, either of the above may change when using subsequent versions of Route_Riter/TSUtil.

Route_Riter COMPACT

- Use this to remove scenery files that existed only in world files that were deleted.
- At this point, all unwanted world files should have already been deleted.
- Tab to "Route Utils".
- Navigate to the route to be processed in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Select the route to be processed by using the upper left window to navigate to it and double clicking it.
- Click the "Confirm Route" button.

- Run COMPACT by clicking the "Compact Route" button.
- Respond "OK" to the first message.
- You should see periodic indications of processing such as window changes, hour glass, and messages.
- Respond "No" to a message about viewing, unless you want to.
- Respond "Yes" to a message about confirming the "MOVE" of the files out of the route, unless you want to quit.
- Wait until a Report Form is generated. It is a listing of moved files.
- Respond "Exit" when done.
- That's it. The extraneous files have been removed.

Route_Riter DUPLICATE

- Use this to duplicate a route in order to rename it.
- At this point, a route folder name and a route name should already be known.
- The route folder name (RouteID) is used in the RGE selection window, a parameter in activities, and Explorer.
- The route name is the used in RE/AE/Sim selection windows.
- To prevent confusion, you could use the same name for both, thus consistency across MSTs, Explorer, and tools.
- Tab to "MSTS File Utils"
- Pick the route to be processed, navigate to it in the upper left window, and double click it.
- Click the "Confirm Route" button.
- Select the route to be processed by using the upper left window to navigate to it and double clicking it.
- Click the "Confirm Route" button.
- Run DUPLICATE by clicking the "Duplicate route with new name" button.
- Enter the names you picked when asked.
- You should see scrolling and window changes as processing occurs.
- The function does not issue a Report Form. It just quits.
- That's it, the route has been duplicated with a new folder and route name.

Note: A .trk file contains other informational entries about the route, including a narrative description. Such entries are not changed by Duplicate, thus they should be edited if desired.

TSUtil MERGE

- Use this to merge two routes.
- At this point, both routes must be in their proper location/shape/altitude.
- In the TSUtil tab, navigate to the route to be processed in the upper left window, and double click it.
- Click "Confirm Route"
- With MERGE, the upper right window is used to navigate to and specify the secondary route.
- Click "Same" in the upper right window. This puts the same path of the left window into the upper right window.
- In it, navigate back to Routes and then to the secondary route and double click it. Do not click Confirm.
- Run MERGE by clicking the "Merge Routes (Merge)" button.
- Exit and respond "OK" then Exit when a message appears about "newRoute" after exiting the form.
- Copy the contents of the "newRoute" folder back into the route's main folder if MERGE completed successfully.
- That's it. The two routes are merged, but certain files are not.

Be aware:

- MERGE copies activity files from the secondary route to the primary, but does not adjust RouteID entries accordingly.
- This would cause problems whenever they are used.
- To fix, edit the RouteID entry in each activity from secondary to change it to the merged route's folder name.
- After that, activity files from the secondary may or may not work in the merged route.
- May have to delete all activity/Path/Service/Traffic files from the secondary, and create new ones.

Also be aware:

MERGE does not merge certain files. Most are used only during RE activity. Not a problem while using the Sim or AE.

The ones ignored are:

- carspawn.dat
- deer.haz
- forests.dat
- gantry.dat
- marker (.mkr)
- reference(.ref)
- speedpost.dat
- spotter.haz
- telephone.dat

Some can be merged manually by copying entries from the files in the secondary into the bottom of the corresponding ones in the primary, and adjusting entry counts at the top. One that cannot is the marker file because of its Lat/Long references.

Also, MERGE does not physically merge the sigscr.dat files of the two routes.

- Instead it concatenates them, which has the same effect as a merger.
- It dose this by adding an additional sigscr.dat file, but with its filename modified with a "\$"
- "sigscr.dat" would be the original.
- "sigscr\$.dat" would be added after the route was merged again.
- "sigscr\$\$.dat" would be added after the route was merged again.
- This can continue up to a maximum of nine times.

Problems during TSUtil Processing?

Look for the following if problems occur during TSUtil processing.

If TSUtil hangs or quits during processing:

- This can happen if a route has too many serious or abnormal issues in it, or is too big.
- If occurs, use TSUtil ICHK and/or other means to check for errors and fix if can. Otherwise may have to give up.

If you get TSUtil message "Continuation between 'tsection.dat'(local) and global 'tsection.dat'...":

- This means the route's local tsection.dat is out-of-sync with your Global tsection.dat. The route should be converted.
- If occurs, use TSUtil CVRT or Horace on the route for a permanent fix, or swap Globals between default and Standardized.
- If occurs during CVRT, it probably means you didn't browse to a default Global tsection.dat, or Global not Standardized.

If you get TSUtil message "Route: Route-path '...\Routes\XXXXX' is NOT a route-definition! ('XXXXX.trk' not found):

- This means the name of the route's main folder is different from the name of the route's .trk file.
- If occurs, change the route's folder name to match the name of the route's .trk file.

If you get TSUtil message "(Path-Reference is not a valid DIRECTORY)" during MERGE:

- This may mean you have a missing Path folder and are using a Route_Riter version less than 7.6.18 (26 Aug 2011).
- If occurs, update to Version 7.6.18 or later and try again. This or a later version can handle a missing Path folder.

If you get TSUtil message "Route-definition: World-File-Number of Start-Tile (XXXXX YYYYY) is missing":

- This means you forgot to change RouteStart to a wanted world file in the .trk file.
- If occurs, use WordPad to change it to a wanted world file.

If you get "Value of 'dx' must be a multiple of ? -- Function terminated!" during MOVE. (The "dx" pertains to east/west

Or if you get "Value of 'dy' must be a multiple of ? -- Function terminated!" during MOVE. ("dy" pertains to north/south).

- This means that the route has a MOVE restriction caused by one or more large tiles.
- Don't worry about it being "dx" or "dy". Any restriction for one would also apply to the other at the same value.
- The "?" in the line is the move restriction value. You need to know this value during analysis.
- The values you'll specify for a MOVE have to be a multiple of (or divisible by) this "?" restriction value.
- See sections above about restricted moves if necessary.

----- **Problems after TSUtil processing?**

Look for the following if problems occur while checking out the results after TSUtil processing.

If you see no change in the route in RE or AE:

- This may mean that the contents of "newRoute" were not copied back into the route.
- If so, copy the contents of "newRoute" back into the route, or start over and use more care.

If you see path lines crossing the border and extending into the white void area beyond wanted areas after CLRDB:

- This means that not all border crossing data base items were deleted.
- If so, start over with more care with border crossers.

If you see path lines existing independently in white void areas after CLRDB:

- This probably means that so many world files were deleted before CLRDB that the remaining ones have no roads.
- Remove the old RDB before adding new roads. Keep it elsewhere as backup in case you were wrong.
- If you already added one or more road sections before discovering, it is too late. Route is corrupt. Re-do CLRDB.

If you see terrain in areas that were supposed to be removed by FILTER:

- This means that not all unwanted world files were deleted beforehand.
- If so, start over with more care in identifying and deleting unwanted world files.

If you see white void at the start location:

- This means you forgot to change RouteStart in the .trk file from a no longer existing world file to a wanted one.
- If so, use WordPad to change it to world file that exists.

If you get lots of path and service errors in AE during loading of the route after CLRDB:

- This may mean you forgot to delete the route's Activities/Paths/Services/Traffic folders before copying "newRoute".
- Restore latest version prior to CLRDB, rerun CLRDB, and delete the above folders before copying "newRoute".
- If you get train database errors, activity related consists are missing from your Trains\Consists folder.

If you get "Send/Don't Send" when loading the route in RE:

- This may mean that RE can't find any activity anywhere in the Routes folder.
- RE doesn't care if the activity is for the route being loaded or not, it's not happy without one from somewhere.
- If so, create an activity for the route being worked on, or copy in another route having at least one activity.
- If that's not the problem, see "If you get stuck..." below.

If you get "Failed to load track database" when loading the route in RE following CLRDB:

- This can be caused if there are no tracks in the world files given CLRDB, probably by mistakes in the deletion process.
- If so, try again after restoring the route and re-doing the analysis or deletion process.

If you get "Failed to load track database" when loading the route in RE/AE/Sim following MERGE:

- This usually means the data bases and world files are out of sync.
- This may be caused by careless mistakes while performing method tasks. Try again after restoring the primary.
- Or by an error in the data bases of one of the routes that TSUtil couldn't resolve, even though it runs in RE/AE/Sim.
- Try again after restoring the primary and swapping routes so that the original secondary becomes the primary.
- If still occurs, may have to give up or perform some deep troubleshooting starting with TSUtil ICHK.

If you see only one route's portion in the AE display of a merged route instead of both portions:

- This means the RouteID in the activity has a name that also exists in the .trk file of another route in Routes.
- Open each .trk file of the other routes until finding the duplicate.
- Change the name in one of the routes to eliminate the duplication.

If you get stuck on something other than the above:

- It may mean that TSUtil is not installed correctly or is at the wrong release level. Check it out and fix if so.
- It may mean that TSUtil reported something wrong in Report Form but it was ignored. Check bottom for an error.
- It may mean that "newRoute" should not have been copied because of an ignored error.
- It may mean that an old "newRoute" from a previous run was accidentally copied. Checking date/time can prevent this.
- It may mean that folders should have been deleted before copying "newRoute". See "TsUtil_en.txt" in the TSUtil folder.
- It may mean that one of the situations described under CLRDB above applies.
- Or it may mean that serious technical issues exist. Running TSUtil ICHK integrity check might provide clues.
- Regardless of cause, resolve if possible, restore previous backup(s), and rerun the function(s) with great care.

That all folks, hope this helps.

Chuck Rickert