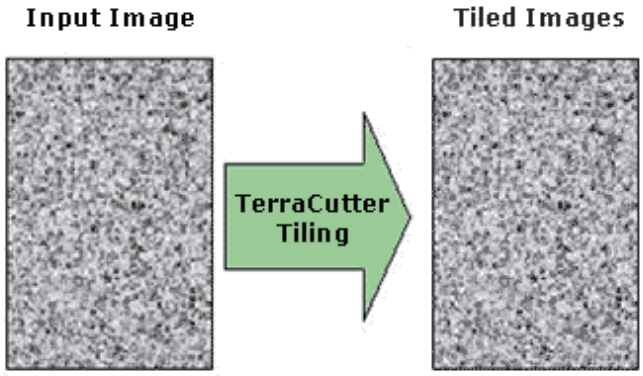
TerraServer Image Loading and Cutting Process



Conceptually, the process of loading data into TerraServer is simple. Tapes arrive from the USGS and SPIN-2 containing uncompressed image files. The files contain too much data to be downloaded over the Internet and are not in a format recognized by Web browsers, so they must be cut and merged with other images and compressed in the JPEG file format.

In reality, however, the intensive nature of preparing the files for loading into TerraServer's database requires a workflow system using several applications to manage the cutting and loading process. This enables many steps of the process to run in parallel. Each step of the process is recorded in the TerraServer database in a set of relational tables called the Load Management Schema. The Load Management Schema schedules and monitors the process of loading new imagery into TerraServer's database. As TerraServer loads and cuts, it fills in the table rows and cells as it processes the images. A set of Active Server Pages (or Web interface) is used to observe and manage the workflow.

Name	Started At	Server	Source Path	Destination Path	Last File	Completed	Files Done
doq-Tape229	7/19/99 6:50:44 AM	TERRACUT1	1:\usgstape229*.*	1:doq\Tape229\	3808362SW	7/20/99 2:48:16 AM	600
doq-Tape231	7/18/99 5:44:39 PM	TERRACUT1	1:\usgstape231*.*	1:doq\Tape231\	4310902SW	7/19/99 6:47:35 AM	462
doq-Tape233	7/17/99 10:41:53 AM	TERRACUT1	g:\usgstape233*.*	g:doq\tape233\	4611735SW	7/18/99 5:41:31 PM	620
doq-tape232	7/18/99 1:23:30 AM	FIBRE1	d:\usgstape232*.*	d:doq\tape232\	4410958SW	7/18/99 5:10:06 PM	360

Above is an example of data from a **LoadJob** table. A **LoadJob** table row is created when a load program is instructed to process a directory or a specific list of imagery received from a data source. The **LoadJob**

program ran on, the date the job started and completed, and the job's current status. Load programs update the **LoadJob** record each time they complete an input file found in the source path and insert a row into another Load Management Schema table called the **ScaleJob**. The **ScaleJob** causes the scale program to create an image's pyramid.

It is the responsibility of the load programs to sort out the GIS details and present each scene as a seamless mosaic of tiles. All knowledge of projection systems, re-sampling of pixels, edge alignment, merging of pixels from multiple images to one, etc., is implemented in the load programs.

There are two image load programs in the TerraServer system - TerraCutter and TerraScale. TerraCutter is responsible for re-formatting imagery received from our data sources, tiling it to attributes acceptable to the TerraServer web application, and inserting it into the imagery database. TerraScale computes the lower resolution tiles and creates the various levels of resolution by using the tiles created by TerraCutter. Both programs leave a "popcorn trail" (an indication or notice) in the Load Management database tables so administrators can monitor progress on loading new data. This section highlights TerraCutter. TerraScale is featured in the next section of the TerraServer Story.

TerraServer receives data from its sources in various formats:

USGS Digital Ortho-Quadrangles (DOQ) data is shipped to Microsoft via Digital Linear Tape (DLT) media written in UNIX "tar" format. DOQ files are in a custom USGS format. Meta-data and image pixels are contained in one file. Data is 8-bit grayscale or 24-bit, RGB color infra-red. TerraCutter converts color infra-red to 8-bit grayscale. DOQ files cover a USGS "standard quarter-quadrangle", which is a 3.75 minute by 3.75 minute square area. The order of DOQ files on tape is random, and adjacent DOQ files can arrive in any order.

USGS Digital Raster Graphics (DRG) or topographical map data is shipped to Microsoft on CDROM media. All 1:24,000, 1:100,000, and 1:250,000 scale maps for a square degree are contained on one CDROM. Images are in the GeoTiff format and generally have a common color map.

SPIN-2 data is shipped to Microsoft on DLT media written in Windows "NT Backup" format. SPIN-2 files are in a custom "Kodak/Microsoft/Aerial Images" format. Meta-data and image pixels are in separate files. Data is 8-bit grayscale.

TerraServer System Administrators use the appropriate "off-the-shelf" program to download a tape or CDROM to a directory on one of six image editing systems. Image editing systems are multi-processor Windows NT Server systems with 500 GB or more local disk. Four servers are 4-processor 200 Mhz Intel Servers donated by Intel. Two servers are 4-processor 300 Mhz Alpha Servers donated by Compaq. Two Intel Servers are connected to 1 TB of Fiber-Channel disk array donated by CLARiiON, a subsidiary of Data General. The other two Intel servers are connected to two Symmetrix SCSI based disk arrays donated by EMC. The two Alpha servers are connected to a 250 GB StorageWorks disk array donated by Compaq. Each system has 4 to 6 100 GB stripe-set disk volumes.

The TerraServer System Administrators launch the TerraCutter image-editing program against a directory containing the image and meta-data files downloaded from tape or CDROM. TerraCutter refers to its Load Management Schema to make sure the job has not been processed previously. Or if a previous run was aborted, TerraCutter will pick up where it had left off. TerraCutter also uses the Load Management Schema to catch duplicate files sent on previously processed tapes or CDROMs. When a directory has been successfully processed, the download directory is deleted, the tape is physically marked as "processed" and shelved. All further processing - sub-sampling to create lower resolution scales, correlating tiles with named locations, merging pixels between tiles, etc. - occurs within the memory of a custom program or T-SQL database statements.

TerraCutter is a fairly complicated C program. The simple part is formatting tiles suitable for the TerraServer web application and inserting them into the database. The TerraServer web application expects tiles to be in one of three formats:

- 8-bit Grayscale, JPEG compressed
- 24-bit RGB, JPEG compressed
- Color, GIF compressed

The ground size covered by a pixel must also be fixed to multiples of 1-meter resolution - for example

resolution as the image is read in. As tiles are produced, TerraCutter saves the tile image into a temporary file, computes the Image table meta-data fields, and inserts the new tile into the database using Open Database Connectivity (ODBC) Application Programming Interface (API) calls. A single image tile is inserted in the scope of one transaction.

The tiling process is the most difficult part of TerraCutter. Depending on the "theme" provider, the input images that form a scene may all be grouped on one tape or may arrive randomly on different tapes. The TerraCutter program must figure out where to look for the original imagery and how to line up the pixels for the database input imagery. It must also know where to start extracting pixels and how to map from input imagery tile to input imagery tile to form a complete scene.

SPIN-2 data is very convenient to work with. A SPIN-2 tape contains all physical images that make up a complete scene. Thus, TerraCutter can cut tiles, merge pixels from multiple physical images, and form a complete tile in one pass over the data.

USGS scenes are more complicated and require the TerraCutter program to handle the merging of pixels to form a USGS image differently than its handling of SPIN-2 data. Thousands of files form a complete USGS scene, and these files do not arrive together on one tape. Hundreds of tapes are necessary to form a complete scene, and the tapes arrive randomly rather than in order.

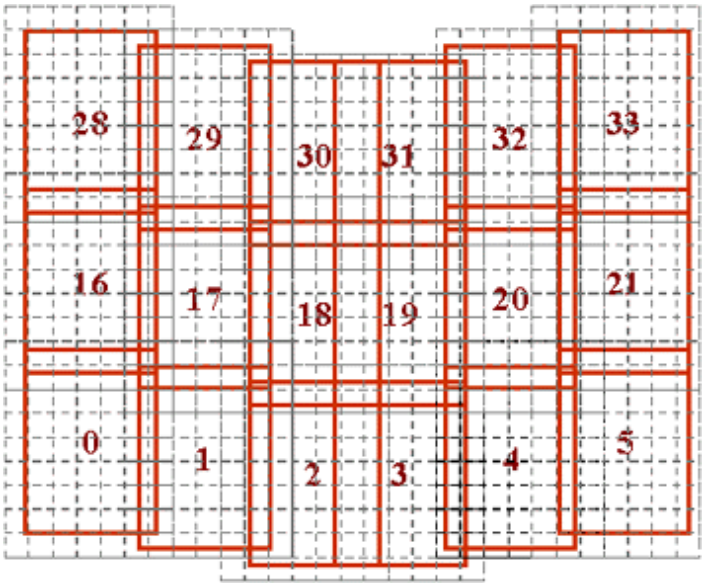
As each tile is cut from USGS data, the TerraCutter program checks the database to see if a tile has already been extracted from one or more previously received USGS physical images. If a physical image has not been previously received, a tile is simply inserted into the database. If, however, a physcial image already exists in TerraServer's database, any of three scenarios can take place:

1. The tile extracted from a tape completely forms a physical image and is accepted. The most recent complete tiles are deemed most desirable and replace the older version.

2. Only part of a tile is extracted and there is a complete tile already in the database. TerraServer keeps the complete tile and throws the partial tile out.

3. Only part of a tile is extracted and a partial tile already exists in the database. The pixels from the new tile and the already existing tile are merged. Eventually, TerraServer will receive a complete tile which will replace the merged version.

Input image files will overlap other image files along the edges. TerraCutter must choose which input image to take a duplicate pixel from. The amount of overlap varies from file to file in each data-set. The diagram below depicts how input imagery files, numbered and outlined with solid thick lines, overlap each other within the UTM coordinate system. The tiles, outlined with light dashed lines within the numbered rectangles, depict the challenge in edge matching.



by 50 to 1500 pixels. However, only one file will contain "map data" while the others will contain map notes and tick marks found along the border of USGS topographical maps. SPIN-2 physical image files overlap each other by a varying amount of pixels depending on the actual photographic rendering process.

TerraCutter tiles each input image independently. White space is added around the input image edge to align to the TerraServer grid system and the input data is re-sampled to the appropriate TerraServer resolution. Tiles are then cut and compressed to a temporary disk file.

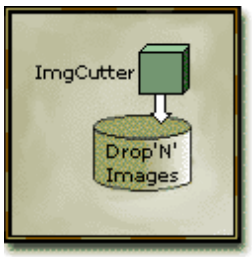
After compressing each tile, TerraCutter looks for a tile with the matching Theme, Scale, X, Y, and SceneID properties in the appropriate TerraServer database imagery table. If there is not an existing tile, then TerraCutter inserts the image into the table and sets a "visibility flag" to "visible".

If a tile does exist in the database, TerraCutter compares the "blankness" of the newly cut tile with the tile in the database. If the new tile does not contain any white space from the input image edges, then the new tile is inserted, made visible, and the old image is set to "invisible". If the new tile does contain some amount of white space, but the tile in the database does not, TerraCutter discards the new tile and does not load it. If both tiles contain white space, TerraCutter fetches the old tile from the database, decompresses it, and does a pixel level merge with the old and new tile. The "blankness" of the resulting tile is computed, the merged tile is inserted into the database and made visible, and the old tile is marked invisible.

TerraCutter performs all four steps in one transaction - (1) check for an existing image, (2) merge pixels, (3) insert new tile row into the appropriate table, and (4) update old tile's visibility flag. Other executing TerraCutters are blocked from modifying the same tile, but can be updating other tiles in the same table. The TerraServer web application performs "dirty reads" of the imagery tables and is not blocked from reading the currently visible row. Thus, we are careful to change the visibility flag of the old tile as a last step so that the web application can get to a valid, but soon to be replaced tile, when TerraCutter is at Step 2 or 3.

Once TerraCutter completes the tile insert, it deletes the temporary on-disk copy of the compressed tile. The program proceeds on to the next tile and repeats the process. When all tiles are cut from an input image file, TerraCutter updates the production status field in the **Theme Original Meta** row to indicate that the input image has been completely tiled. TerraServer Administrators monitor the progress of the TerraCutter program through database queries against the **Theme Original Meta** table.

Should the TerraCutter program abort or be terminated before completion, the program will restart and pick up the tiling process where it left off. The program uses the *ProdStatus* field in the **Theme Original Meta** table to determine if it finished an input image file. It skips through all the completed images until it finds the input image it was working on previously. It repeats the tiling process, but skips loading all tiles that were previously loaded.



Click on the thumbnail to view the load process diagram.

Please continue with the next section for information on TerraCutter's partner, TerraScale.

Microsoft Research

TerraServer Built by Microsoft Research

Contributors