Reveling in Raster Data

Jonathan W. Lowe

Raster-oriented technology supports public Internet applications that manage large raster collections.

According to ESRI's John Baleja, Steve Kopp, James Neild, and Jason Willison, the size of a raster dataset when managed in a spatial database is immaterial to the performance of the applications manipulating it. Specifically, "manipulating" in this context means loading, managing, visualizing (in two and three dimensions), analyzing, and distributing data subsets. ESRI reports that these functions apply to raster datasets measuring 1 GB, 10 TB, or more.

But how big is big? When discussing image size, it's easy to forget what the units really represent. We all know the phrase, a picture tells a thousand words. Measured in units of computer memory, it's actually possible to quantify that phrase more precisely. A byte stores a single character, such as the letter "A." Based on an informal study of my local newspaper, the average number of characters per word is 5.5 (closer to five for sports stories, closer to six for politics!). One gigabyte is 1,073,741,824 bytes (approximately 1 billion bytes). On average, then, a 1-GB image tells 195 million words (1 GB divided by 5.5 bytes, or characters, per word). Or, if the average 400-page hardcover novel contains 100,000 printed words, then that 1-GB image tells 1,950 hardcover novels worth of words. That's a fair estimate of how many hardcopy books the average person might read in a lifetime.

Net Results columnist **Jonathan W. Lowe** covers the role of emerging technologies in the exchange of spatial information. Lowe is the owner of Local Knowledge Consulting (Berkeley, California), where he designs and implements spatial Web sites. He can be contacted at info@giswebsite.com.

If this sort of comparison begins to communicate the vast size of today's imagery collections, next consider how quickly they can be aggregated. For instance, Baleja reported that, for a demonstration at this summer's ESRI International User Conference, one person loaded 4 TB of image data in one week. A terabyte is 1,024 gigabytes — by our little equivalence comparison, that's almost 2 million hardcover novels. So, some solitary technician at ESRI coaxed a single desktop computer into reading and remembering the equivalent of 8 million novels in just one week's time.

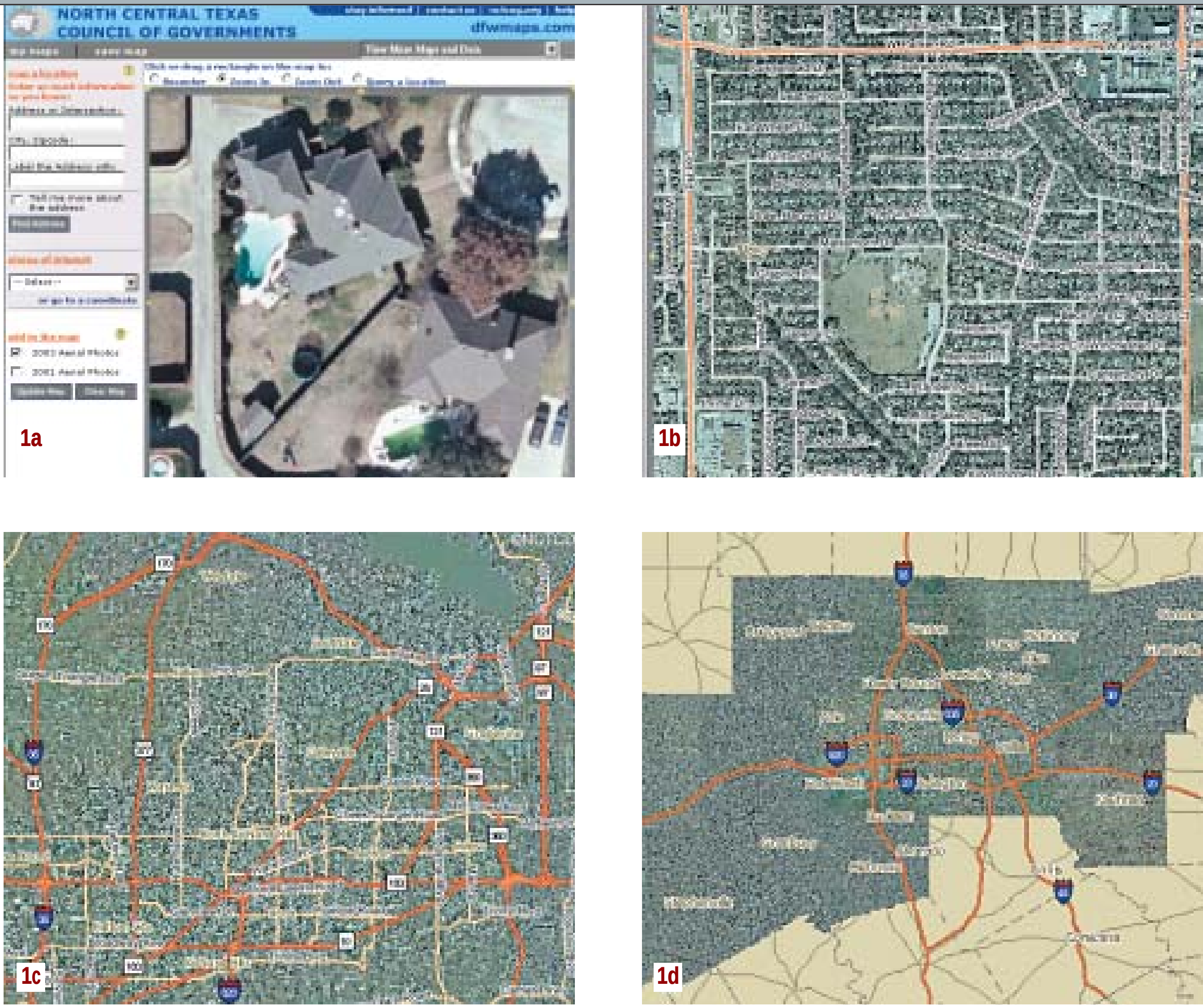
Verbal comparisons, however, only go so far. To get a better sense of size, why not directly explore some of the imagery collections now appearing online because of the advances in raster storage and retrieval technology. For instance, the North Texas Council of Governments Web site (www.dfwmaps.com) serves imagery sharp enough to resolve swimming pool diving boards, automobile windshields, and pipe vents extruding from the roofs of houses (see Figure 1a). The amazing thing is not so much the fine grain of the images, but how much fine-grained imagery is packed into one comfortable collection. Zooming out from raster-resolution scale to the limits of the imagery's extent puts its magnitude in perspective (see Figures 1b, 1c, and 1d).

Partial Pyramiding

Supplying such a large collection of raster data through the North Texas Council of Governments' single system is possible because of advancements in technology performance and management. But technologies for quickly extracting only the requested extent and resolution raster-data clip from a larger collection are by now widespread. The real news is about dataset size breakthroughs combined with enhancements that simplify or speed administration and analysis. One such new ESRI arrival is *partial pyramiding*, a technology that allows databases to update only the changing areas of an imagery collection in response to the insertion of new imagery. For instance, adding a new "patch" image to a larger collection, like replacing one square of a quilt, does not require reprocessing of the whole pyramid structure.

To understand the enhancement requires clarity on the base capability — raster pyramiding. *Raster pyramids* are multiple copies of an image dataset, but with each copy saved at a different resolution, from the original highest resolution to down-sampled low resolution. This set of copies is the pyramid. If the North Texas Council of Governments dataset uses raster pyramids, then each of the zooms in Figure 1 could be rendering a different copy or level of the pyramid. As users zoom out, they see resampled copies of the same original raster data at coarser spatial resolutions.

Pyramids improve overall performance by generalizing away unnecessary detail for a zoomed-out view before the request ever arrives. Then, when a user does zoom to the full data extent (as in Figure 1d) he or she sees the top of the pyramid — a single, low-resolution image of the whole raster collection. At tighter zoom extents (as in Figure 1a), the database clips the appropriate subset of the high-resolution pyramid base imagery (or, depending on technical implementation, might extract a pre-existing tile by referencing its indexed polygon footprint against the rectangle of the view extent). At all levels, though, the resampling is



Figures 1a, 1b, 1c, and 1d. Views of the North Texas Council of Governments Web site when zooming from backyard scale (**1a**) to regional scale (**1d**) illustrate the large coverage of this fine-grained raster dataset.

already done, so the viewer quickly receives an image with appropriate detail for the requested scale.

Clearly, end users benefit from pyramids, but how do they impact data administrators? When inserting imagery into a database for the first time, the spatial data management software must duplicate the whole collection at enough resampled pyramid levels to perform well at any zoom scale. The larger the collection, the longer this task takes. Before partial pyramiding, the entire process had to be repeated with any imagery update. Now, rebuilding the pyramid takes less time, especially if the update represents only a small partial area of the imagery extent. So this technological advance is a time-saver feature for raster-collection administrators.

Analysis Without Paralysis

Raster data are not all images, of course. Images, mosaics, snapshots, raster catalogs, and data organized by a regular grid such as a digital elevation model (DEM) or temperature grid all qualify as raster data. And because of the recent increases in performance, ESRI reports that even very large raster datasets are now fully supported by such standard tools as clipping, reprojecting, mosaicking, warping, georeferencing, calculating polynomial transformations, and applying ESRI's extensions (3D Analyst, Spatial Analyst, and ArcGlobe) for raster analysis. What are the implications to application design?

Large-performant raster data in combination with these familiar raster analysis tools encourage not just data storage and access on local machines, but analytical transformation and access via Web services. For example, the U.S. Geological Survey (USGS) has distributed DEM data from a public Internet site for years. Coverage was nationwide, but with data broken into quads or counties and stored as files. More recently, USGS began providing seamless national coverage of DEM data stored in an ESRI SDE-based database and available on the public Internet (http://seamless.usgs.gov). This shift in storage from multiple files to a single repository means that analysis can also be seamless. In other words, USGS can now provide Web services not just to display raw DEM data, but also to render them seamlessly as a slope or aspect map. And the area for rendering can bridge quad map edges or county boundaries.

For example, the seamless USGS site already provides a simple online analysis service: the elevation profile tool. This tool allows users to create a line segment by clicking points on a map and then graphing that transect's elevations based on any one of a long list of seamless DEM datasets. To test the tool, I began a transect in a familiar baseball diamond in Berkeley, continued across a steep ridge bordering the ball field, and ended in a valley on the other side of the ridge (see Figure 2a). The tool accurately graphed my transect from left to right — a flat field, then nearly a vertical cliff face, then stepping more gradually down into the valley (see Figure 2b). Convinced that the tool worked on a small area, I zoomed nationwide and drew a transect line from the Pacific to the Atlantic Ocean (see Figure 2c). The collection is truly seamless and performant — the site generated a graph in the same amount of time as the first test, returning a topographic profile spanning multiple states, dozens of counties, and who-knows-how-many quad sheets (see Figure 2d).

Other organizations with similar database-centric raster analysis, distribution, and Internet visualization approaches include the North Carolina Floodplains site (www.ncfloodmaps.com), the Louisiana Department of Environmental Quality (http://map.deq.state.la.us), and Earthsat (www.earthsat.com).

Faster Raster

Raster analysis requires efficient number crunching. For ESRI's implementation, database features and ArcSDE enable such performant number crunching of large raster collections. Analytical operations isolate a subset of the larger collection using database indexing techniques to speed subset extraction. A typical processing flow might begin when an analysis tool, such as an aspect map request, pulls a raster subset out of the database, performs the aspect operation, delivers the results to the user, and then, possibly, puts the result back into the database for later retrieval without the need for a second calculation.

In addition to database indexes, software, such as ESRI's ArcSDE, also enhances performance of analytical operations by buffering input and output queues. Often misunderstood, ArcSDE is not a database unto itself. Rather, it is a translator and "traffic cop" between a third-party database (DB2, Informix, Oracle, or SQL Server) and an ESRI

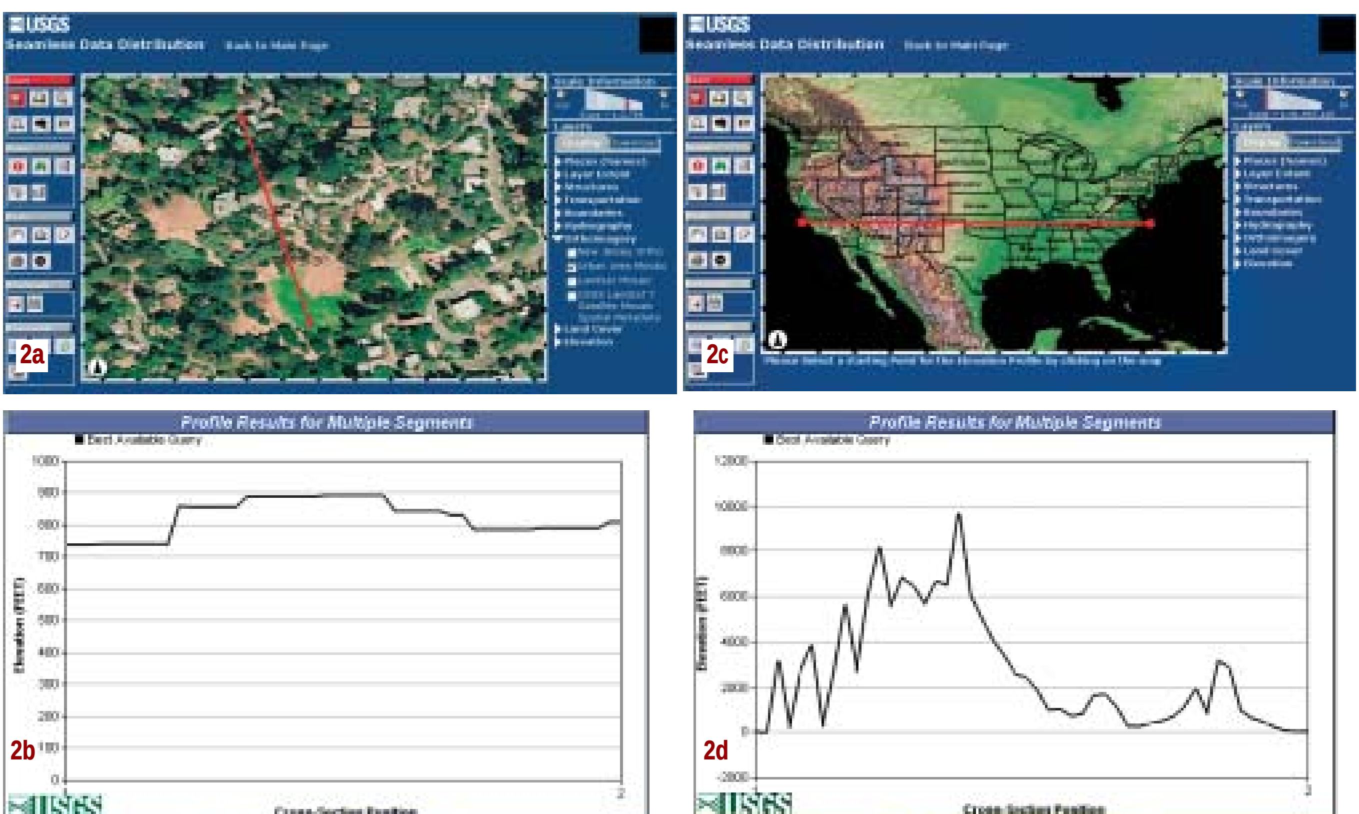


Figure 2. The USGS seamless server Web site demonstrates that a local transect (**2a**) and corresponding elevation graph (**2b**) take the same time to process as a national transect (**2c**) and its elevation graph (**2d**) — both using the same seamless DEM and imagery database.

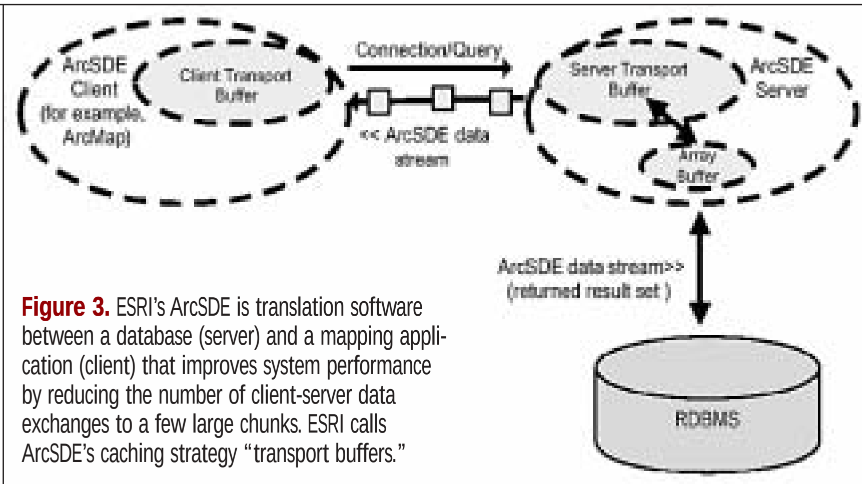


Figure 3. ESRI's ArcSDE is translation software between a database (server) and a mapping application (client) that improves system performance by reducing the number of client-server data exchanges to a few large chunks. ESRI calls ArcSDE's caching strategy "transport buffers."

Diagram courtesy of ESRI

client application (ArcMap, ArcCatalog, and so forth). ESRI's approach relies on the database (server) to store data, the application (client) to store analysis code, and ArcSDE to exchange data efficiently between them. For any given data exchange or stream between the database and processing code, ArcSDE allocates transport buffers on both the server and client (see Figure 3). When the server is responding to a client request for data, ArcSDE fills the server's transport buffer before sending its resultantly large chunk of data to the client. This strategy — called either array buffering or ArcSDE caching — reduces the number of exchanges between client and server by fetching and inserting data in larger chunks. And fewer exchanges yield lower "overhead" and consequently improve performance.

Some question the value of databases for raster data management. After all, raster data don't change the way vector data do. Once they're captured and rectified, nobody edits digital orthophotos. Why store them in an environment (the database) designed specifically to handle multi-user editing transactions? In addition to performance boosts over the file-based systems mentioned earlier, there are administrative advantages to putting all of your data eggs into one database basket.

For example, after investing considerable time to build a single large dataset on some centralized machine, administrators occasionally have to move it elsewhere. This kind of operation — moving large datasets between physical machines — is the sort of problem that database vendors (rather than spatial software makers) compete with each other to solve. Spatial data are not always at the top of the big database vendors' list of features to support, but now and then, spatial data get lucky and are carried along by one of the competing solutions. For instance, Microsoft's SQL Server supports a capability called *portable tablespaces* whereby administrators can dump a database's entire contents onto cheap large-capacity storage hardware, and then move that dumped tablespace to another machine. Upon arrival, the receiving SQL server database embraces the new portable tablespace like a lost child, and the job is done. Though a fairly standard feature (by various names) in all professional databases, SQL server's support for portable *spatial* tablespaces is noteworthy.

Security is another good reason to consider databases for raster storage — our government's concerns about homeland security include sharp focus on protecting potentially sensitive raster data. Storing raster data in a database offers an additional security barrier to that of the file system. Even if a cracker compromises your organization's outer defenses and gains access to the command line of your server's operating system, he still has to chisel past the security measures guarding the database itself, which are independent of the operating system.

Raster Masters' Dreams

In 1968, science fiction novelist Philip K. Dick asked, "Do androids dream of electric sheep?" Wondering the same of ESRI's raster dream team, I asked what excites them most about the future of raster data management.

Confident with their recent advances in terabyte-size raster data management, they are excited by the challenges of even larger datasets. Willison elaborated, "At last year's ASPRS [American Society of Photogrammetry and Remote Sensing] conference, a speaker predicted that we would soon be working on pedobytes of imagery data, and the crowd gasped. This year, ASPRS speakers talked about breaking the exobyte barrier." (At some point, size really does matter!)

All are eyeing the Holy Grail of lossless compression. For instance, JPEG image-processing routines discard original data to compress images more tightly. When done well, the changes are undetectable to human perception. The JPEG specification refers to this strategy as *lossy compression* because, once so processed, the original data are lost. Some raster specialists believe that compression ratios of 2:1, 5:1, or 20:1 will be possible with lossless (rather than lossy) compression. Since smaller storage footprints result in faster database extraction and processing, it's understandable why compression is a hot topic.

The most compelling future vision doesn't pivot on the raster engineers' delight in pushing size or performance envelopes. Instead, what sent shivers down my spine was Kopp's description of how raster management technology supports global research. Today's raster data management enables analysis and visualization of a seamless sea surface-temperature range for all of Earth's oceans over a two-year sequence. What will that analysis tell us about climate change or humanity's choices for future planetary stewardship? That's the right kind of dreaming. Dream on, raster masters! ⊕