WinSAR Data Archive
Serving the Western North America Interferometric Synthetic Aperture Radar Consortium

Background
The western part of North America is the focus of intensive scientific research into a variety of plate boundary processes including earthquakes, volcanism, mountain building, and micro-plate tectonics. The technique of space-borne Interferometric Synthetic Aperture Radar (InSAR) provides an excellent means of observing deformation over broad areas. It is capable of tens of meters spatial resolution at monthly or greater intervals. The sheer number of InSAR data sets, along with the size of each set, requires that researchers be able to accurately search for and quickly download the appropriate data. The result of our work was an integrated data system, including:
- a flexible backend MySQL database containing the metadata for each entry
- the ability for peer users to add data on their own computers to the archive catalog
- client/server code to negotiate a fast, parallel socket file transfer
- an improved web-based GUI

Purpose
The WinSAR data archive has been under development since 2002, yet prior to this summer it lacked several fundamental functionalities. The four tasks remaining were (1) to migrate the metadata entries from an ASCII text file to a MySQL database, (2) to allow consortium members to add entries to the database through a simple Web interface, (3) to create high-speed file transfer software for faster data set downloads, and (4) to improve the look and feel of the site.

In addition, we wanted to redesign the WinSAR website, so that processes such as searching the database, submitting entries, and setting up the download client would be more intuitive.

Web Solutions
The main purpose of the WinSAR website is to serve data sets to the radar remote sensing community. This summer, we enhanced the search capabilities of the site to provide more efficient searching as well as future scalability.

- The existing implementation used a flat ASCII file to store the archive's metadata. Our implementation uses a modern SQL database. We performed the migration using a PHP populator script.
- We created a new search interface, using HTML, Java, and JavaScript, that allows the user to specify whatever criteria the returned entries should satisfy (including map location, satellite orbit number, etc.).
- Once the user submits these search criteria, a PHP script converts the form data into a MySQL query, connects to the database, and retrieves the metadata for each "hit".
- The search results show up in a PHP-generated table, one entry per row, along with download links to the corresponding data files.
- A separate section of the website allows WinSAR Consortium members to add new entries to the database. Once again, a PHP script uses form input to build the appropriate MySQL database query.

Check us out at: http://winsar.stanford.edu

Parallel File Transfer Solutions
TCP/IP connections are typically limited to 1-2 MB/s rates over long-haul lines. We developed special client and server code to implement parallel-socket downloads, so users could avoid this speed limitation.

- We chose to code the client and server in Python, because of the portability of wxPython GUI applications to multiple platforms.
- Using MIME types, our client can be configured to run automatically when the user's web browser encounters a "Fast download" link on the WinSAR search results page.
- The client requests a specific data file from the server. After the server authenticates the request and tells the client the file size, multiple socket connections are opened and the download proceeds.
- The socket connections are handled in separate threads to boost performance. Data locks prevent threads from interfering with each other during key processes such as file writing.
- Tests show that the speed upgrade factor over standard browser downloads is on the order of 10x, up to 20-30 MB/s.

Data Set Thumbnails
To provide a better user experience, we decided to provide the option of visualizing the data before downloading it. To do this, we ran the data files through a script that translated the raw data pulses into a topographical image. The end result: a convenient thumbnail-sized preview associated with each data set.