THE INTERACTIVE 4-D MCIDAS WORKSTATION

Bill Hibbard and Dave Santek
Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin U.S.A.

Marie-Francoise Voidrot-Martinez
French Meteorological Office
Paris, France

1. MCIDAS UNDER UNIX AND X WINDOWS

We have implemented a version of McIDAS (Man-computer Interactive Data Access System) running under UNIX and X Windows, on the Stardent GS-1000 workstation. This system goes a significant ways toward satisfying the goals described two years ago in "A Next Generation McIDAS Workstation" (Hibbard, 1988).

This McIDAS implementation is quite similar to previous McIDAS implementations, particularly PC-McIDAS. The file structures of this system for grids, images and other forms of meteorological data are identical with those of PC-McIDAS. All of the FORTRAN applications code is identical with PC-McIDAS, except for compiler differences. The system level code is written in C rather than assembler and consists of two C modules of 900 lines each. This McIDAS implementation supports variable sized frames dynamically created by the user, pan and zoom operations, and the flexibility to allocate video system bits to images versus graphics when McIDAS is started. Thus GOES IR data can be viewed in ten image bits overlayed by one graphic bit.

This McIDAS interfaces with the user through X Windows, providing a pseudocolor window for images and graphics and a number of Xterm windows for command input and text output. The new McIDAS function for interactive model visualization uses a true-color window for graphics and numerous widget windows for user control. McIDAS commands are entered to a standard UNIX shell, so all of the shell tools are available, including command history and shell scripts. The X Windows mechanism for cutting and pasting text is available to edit McIDAS output into text files. Numerous graphical user interface widgets are available.

An implementation of McIDAS using UNIX and X Windows, and avoiding the use of assembly language, should be easy to transport to other computer systems and to connect with other systems. The critical question for porting to another system is the performance of the X server on the target system for supporting animation.

2. INTERACTIVE 4-D VISUALIZATION

We have exploited the power of the Stardent workstation to create new McIDAS functions for interactive visualization of multivariate four-dimensional data sets, such as those produced by numerical weather models. These data sets are stored in five-dimensional arrays, composed of three space dimensions, time, and one dimension for enumerating different physical variables. For example, we have applied our system to visualize the ECMWF data set of 4 February 1988, which shows an extratropical cyclone moving across the North Atlantic during a one week simulation. Our software stores this data set in a five-dimensional grid of 24 latitudes by 46 longitudes (2.25 degree spacing) by 14 vertical levels by 168 time steps (one per hour for a week) by 8 variables (pressure, temperature, specific humidity, vorticity, divergence, potential temperature, vertical wind speed, and horizontal wind speed). This is a total of over 20 million grid points. Note that a workstation with 128MB of memory could be used to visualize a data set of 50 million grid points.

The basis of our visualizations is real-time animation. That is, the ability to generate the images as fast as they appear. This allows the user to interact with the display by controlling the contents of the images and getting immediate response. The user sees a three-dimensional box containing moving scenes generated from the weather data set. The scene may contain a topographical map, trajectory lines, and iso-level contour surfaces of scalar variables. The user can interactively select which of these visual elements to display at any moment. For example, the user can select the topographical map with a 280 Kelvin potential temperature surface and a 5 cm/sec vertical velocity surface. These will be animated to show the interactions between these variables. The user can interactively control the animation--start it, stop at an interesting time, and single step forward or backward to watch dynamics slowly. The user can interactively change the iso-levels for contour surfaces to see different values for variables. The surfaces for a selected variable are redrawn for all time steps, and this is done asynchronously with the animation, typically at a rate of about two per second. The user can also interactively rotate, pan and zoom the images in three dimensions. This is a very powerful way to understand complex geometries in three-dimensions.

3. COMMUNICATIONS

The Stardent GS-1000 supports the TCP/IP protocol over Ethernet, as well as NFS (network file system). These can be used to transfer files from another computer, or to provide a byte stream between applications on different machines. The OS/2 McIDAS includes applcations for transferring all the McIDAS file structures to and from the McIDAS mainframe. We will port these OS/2 communications programs to our UNIX implementation and link them with the raw data transfer via TCP/IP. This will be a simple task and will provide access to all the data sources of the McIDAS mainframe. We have no plans to port the McIDAS satellite data ingest software to the GS-1000.

TCP/IP is supported on almost all supercomputers, so this will provide an easy way to get model data into the GS-1000 for interactive four-dimensional visualization. The primary difficulty here is to write an application for translating the model output data format to our McIDAS file structures, a task which we have done for many different data sources.

Stardent has a commitment to support higher rate communications standards as the are defined. FDDI at 100 million bits per second is nearly settled, and there is active movement toward communications at one billion bits per second.

4. VISUALIZING DATA SETS OF BILLIONS OF POINTS

Our visualization work is based on the idea that the problem is the size of meteorological data sets and that the solution is highly interactive access to those data sets. The enormous size of the data sets gives the important information too much room to hide in. Also, the data sets are cumbersome to manage, hindering the effort to look at them in a variety of ways. However, advances in hardware and software systems are making it possible to process lots of data quickly, letting scientists make choices and get fast visual response to those choices. These tools let the user hunt quickly through a lot of data. Our software tools running on the Stardent workstation enable a scientist to interactively visualize a data set of 50 million grid points.

We have done a preliminary design for an extended version of our system which would provide interactive visualization of data sets of 5 to

10 billion grid points, 100 times larger than our current capability. This design distributes our current software onto both a Stardent workstation and a CRAY supercomputer, with the gridded data set residing on fast disks attached to the CRAY, and a very high speed link between the CRAY and the Stardent.

The distributed system could be used to visualize a data set such as the hurricane simulation which Greg Tripoli of the U. W. Meteorology Department is planning to create using his thunderstorm model (Tripoli, 1989). The hurricane data set will consist of 4 million spatial grid points by 10 physical variables by 2520 time steps (every 4 minutes for a week), and the simulation will require about 500 hours on a CRAY 2. The output data set will consist of 100 billion grid points which would need 400 billion bytes to store as 32 floating point values. This is an unrealistic amount of storage, so we would reduce the resolution of the model's output by 2 in space and time and compress the values to 8 bit integers. The resulting data set would have 1260 time steps and 500 thousand spatial grid points for a total of 6 billion bytes of storage.

The distributed system can be understood by looking at the sequence of operations involved in visualizing the hurricane data set. We would seek to produce real-time animations from this data set at about five frames per second. During each 0.2 second frame time, the system would execute the following steps:

a. The Stardent will send to the CRAY the user's controls for selecting which combination of physical variables to view, for selecting which time step to view (which 8 minute step out of a week), for selecting iso-levels for contouring variables, and for selecting the geographic extents of the region to view.

b. The CRAY will read the grids for the selected time step and physical variables from the disk. If we limit the number of simultaneous variables to three, this would be five frames per second by three variables by 500 thousand bytes per grid giving 7.5 million bytes per second of disk bandwidth. This transfer rate is achievable with supercomputer disk systems.

c. The CRAY will generate subgrids of about 50 thousand points each, by subsectoring and possible resolution blow-down, according to the user's selection of geographic extents.

d. The CRAY will generate polygonal contour surfaces from the subgrids for each selected physical variable, according to the user's selection of iso-levels. The surfaces will contain very roughly 50 thousand triangles.

This is a heavy computational load and will require polygon-finding algorithms adapted to exploit the parallel and vector facilities of the CRAY.

e. The CRAY will transmit the triangles to the Stardent. Fifty thousand triangles require 3.6 million bytes of storage, so the overall data rate would be 5 frames per second by 3.6 million bytes of triangles by 8 bits per byte giving 144 million bits per second. The current Stardent renderer requires that the triangles be compressed into polytriangle strips, with one vertex per triangle. This alters the storage to 1.2 million bytes per 50 thousand triangles and the data rate to 48 million bits per second. However, polytriangle strips present some problems for processing and image quality, so a solution without polytriangle strips is preferable.

f. The Stardent will render the triangles according to the user's selections for 3-D pan, zoom and rotation, surface color and transparency, and light source placement. This calls for the Stardent to render 5 times 50 thousand or 250 thousand triangles per second.

Note that the images generated will include a variety of visual elements such as wind trajectories and maps. They are much easier to handle than contour surfaces, so we have left them out of the above discussion. Also note that the distributed system could be applied to interactively visualize other weather and climate simulations of similar size.

5. INTERACTIVE MODEL DEVELOPMENT

We are also interested in using distributed algorithms to help scientists interactively develop thunderstorm simulations, a task requiring numerous trial and error adjustments to initial atmospheric conditions and to model parameters. This is done through an iterative cycle of short simulation runs, inspecting the model output data and comparing them to known storm behavior, and adjusting the initial conditions and model parameters.

Interacting with a running model is tricky because of man-machine coupling mismatches. Large simulations are much too slow for interaction, and we would concentrate on simulations with about 100 thousand spatial grid points. Even this size of simulation would progress too slowly to be directly visually interesting, so the visualization must be asynchronous with the model. The system will maintain an accumulating model output data set on the CRAY, and let the user move around in the time steps, rather than being in lock step with the model. To control cumulative error, the model needs to calculate many more time steps than the user wants to see. The storm model will calculate time steps for every two seconds of storm time but only store a time step in the data set for every minute of storm time. The user will visualize from the accumulating data set, using the same distributed software system that was described for visualizing the hurricane above. When the user wants to change the simulation, the system will enter a new operating mode allowing the initial conditions and model parameters to be changed and the model to be restarted at an earlier time step. This iteration will continue until the simulation behaves to the satisfaction of the scientist.

6. ACKNOWLEDGEMENTS

We wish to thank Dave Kamins and Jeff Vroom of Stardent for their help with this work. We also want to thank Verner Sumoi, Francis Bretherton, Robert Fox, James Dodge, Greg Wilson and Daniel Soderman for their support and encouragement. This work was supported by NASA/MSFC (NAS8-33799 and NAS8-36292).

7. REFERENCES

Hibbard, W., 1988: A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, California, American Meteorology Society, 57-61.

Tripoli, G. J., 1989: The University of Wisconsin Regional Atmospheric Modeling System: Dynamical Framework and Simulation of Geostrophic Adjustment. In preparation.