VIPAR: Visualization in Parallel

Jo Leng, Kevin Roy, Dr J.M.Brooke
CSAR Applications Support

VIPAR was originally a 2 year EPSRC research project started in April 1996 and concluded in December 1997. It was carried out at the Manchester Visualization Centre, University of Manchester and had industrial partners AVS Inc. and Meiko Ltd. VIPAR started as proof of concept research but now there is interest in making it into a stable tool that can speed up the visualization pipeline. A preliminary study of VIPAR and the work required was conducted as a UKHEC case study. A report will be available from the UKHEC web site shortly (http://www.ukhec.ac.uk).

The project's original aim was to remove performance bottlenecks from within the visualization pipeline. At that time several other projects had implemented solutions that were tied to specific parallel support libraries, hardware or visualization systems. These all had poor portability, VIPAR was developed to have better portability.

The Structure of VIPAR and It's Libraries

Visualization systems like AVS/Express, Iris Explorer, IBM Data Explorer and Khoros are described as Application Builders or Modular Visualization Environments (MVE). These tools provide access to large visualization libraries via a highly developed GUI which is easy for new or non-technical personnel to operate. The user drags and drops modules from the systems libraries and connects them to produce a network which in turn produces an associated application and visualization.

The VIPAR libraries are a suite of routines that provide an interface between the visualization system and the message passing system. The three libraries are, VPRvsi - a visualization library to provide an interface to the visualization system, VPRidd - an intermediary library that performs the necessary calculations and provides the interface between the two other libraries, and VPRdd - the lower level message passing and parallel process control library. These libraries are designed to sit on top of one another and provide increasing levels of complexity.

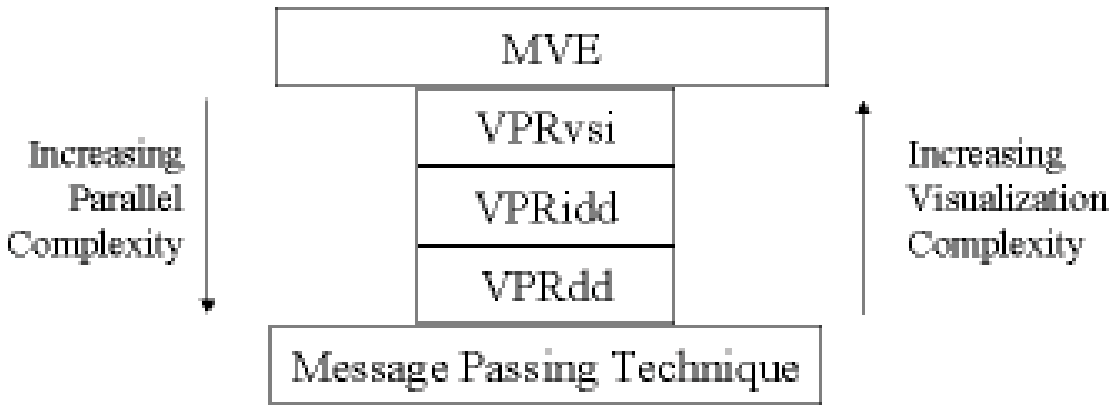


Fig 1: Shows how the VIPAR libraries are layered to achieve portability between MPI libraries and the visualization system.

VIPAR is also a construction kit for producing parallel visualization modules quickly and easily. However this tool, the DDTool, was only ever produced as a prototype.

Current Interest In VIPAR

Since the end of the VIPAR project there have been several changes that have created interest in re-opening the project:

* Development of hardware rendering systems has meant that the major visualization bottlenecks have moved away from the rendering components into other parts of the visualization pipeline. It makes sense to parallelize just the bottlenecks, i.e. the CPU intensive components.

* Now there are more and larger MPP systems allowing users to increase the size of data sets.

* More application areas are using visualization.

* Computational steering/monitoring is becoming more important.

Technical Aspects of Reviving VIPAR

The main technical problem of reviving VIPAR came from the fact that VIPAR originally started as a research project, it was not a software development project. It proved its hypothesis by proof of concept, there was no tested VIPAR distribution produced.

The Parallel Libraries

When the original VIPAR was designed it was assumed that MPI libraries would move to the MPI-2 standard and this would have spawning included, parallelisation within VIPAR currently relies on spawning. If this had happened the complexity of VIPAR could be reduced, however, it is still only LAM MPI that supports spawning. LAM MPI is freeware and not fully implemented/tested on all platforms.

Now VIPAR is being re-evaluated a more robust solution is being considered. If a socket was put into VIPAR it could be used to call codes on the local or remote machine and these codes could use the most efficient parallel library whether it is MPI or Open MP. This new solution would also reduce the complexity of VIPAR and so increase its usability.

The Visualization System

The original VIPAR modules were designed for academic testing and did not have the sort of robustness required for routine work. Much effort has been put into transforming VIPAR's structure into a sturdy project structure (the International AVS Centre's project structure), making the linking between AVS/Express and the VIPAR libraries non-specific (connected via environment variables rather than absolute paths) and re-implementing components so that 2 or more VIPAR modules could be used in one session. At the same time the user interface has been re-developed to be more robust and hopefully idiot proof.

The Current Status of VIPAR

VIPAR has currently been ported so that it will run the isosurfacing example with the 32 bit version of AVS/Express and LAM-MPI on Fermat (a SGI Origin). The user interface, error handling and structure have all been improved to make the it more robust to use and install.

The isosurfacing module produces a surface of a given value within a scalar volume of data. The surface is in effect a 3D contour. You would use an isosurface if you wished to see the graphical depiction of a particular data value within a volume of data. The figures below show an isosurface and its associated network in AVS/Express. When the parallel isosurface module, PMiso is expanded you can see that it is made up of a distributor, a harness and a user interface.

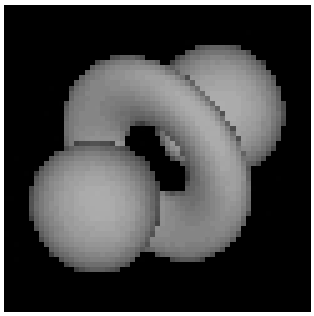
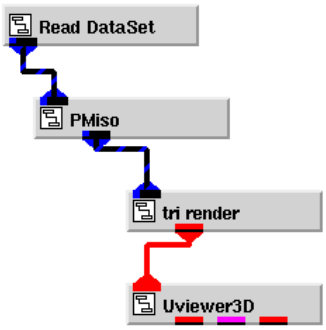


Fig 2: Isosurface of the electron potential of a hydrogen atom.



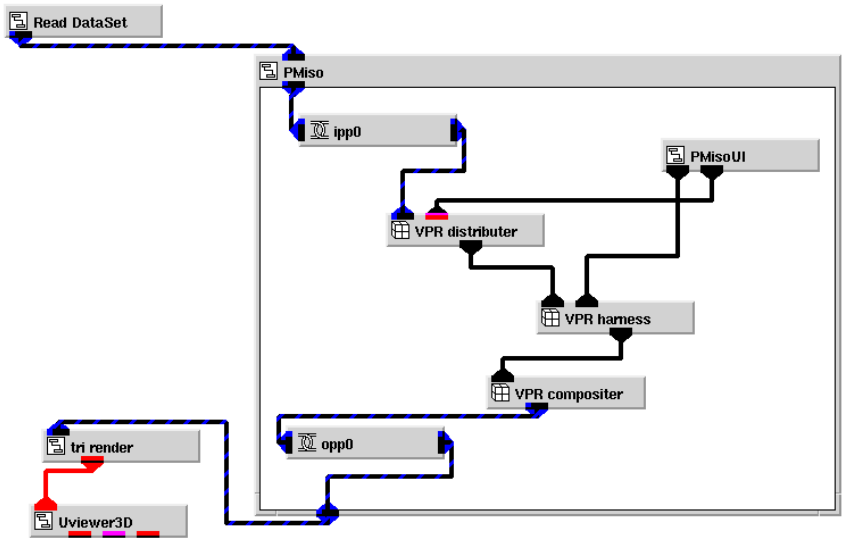


Fig 3: The network containing the VIPAR isosurfacer. The image on the left has the VIPAR module expanded to show the internal structure.

In the future we aim to take VIPAR out of research and to make it into a stable product that can communicate through a portal to other machines and parallel libraries. Not only would this improve the performance of VIPAR but would enable computational steering and reuse of computational functions within the visualization system.

Acknowledgements

The original research project was developed by S. Larkin, A. Grant and W.T. Hewitt at Manchester Visualisation Centre, Manchester Computing, University of Manchester.

People Focus

Jon McClaren

I started work for CSAR on 5th March 2001, working as a Software Engineer on the EUROGRID project, reporting to John Brooke. I work on developing GRID technologies; specifically, I will be constructing a GRID Resource Broker, with the help of John Brooke, and people from FECIT. I am currently completing a test installation of UNICORE which can run jobs on our Cray T3E.

Previously, I did a MPhil and PhD here at the University of Manchester, in the Department of Computer Science, where I was supervised by John Gurd. My MPhil compared different parallelisation paradigms, including directive-based parallelisation and MPI. My Ph.D was in parallel computing - I developed a novel method for automatic parallelisation, based on a hypothesis about data structures in scientific programs. The PhD was only recently submitted, and I'm still waiting for the Viva. I did my BSc over in York, and inbetween my two educational stints, I worked for a couple of years as a Software Engineer with a company in Altrincham. At this company, I was in a development team of around 20 people, working on the company's core product, a LIMS (Laboratory Information Management System). At this time, I worked mainly on UNIX systems (with some NT and VMS work too), writing in C; I also became very familiar with the ORACLE database system.

During my PhD, I became a film addict - usually I go to the cinema twice a week, or even more. Other interests include listening to various forms of music, including upcoming Manchester bands like "Elbow" and "I am Kloot", as well as classical music (especially when its at the Bridgewater Hall). Now that the PhD is done, and I have more time, I'm getting back into cycling and swimming.

Mike Daw

I've been working with computers for the last 5 years. My last job was with Logica - a company that was described by its CEO as a "British Microsoft". I think I'm still bound by a confidentiality agreement, so I won't comment on this.

My other IT jobs were for eccentric little software houses populated by an eclectic, interesting and sometimes talented bunch of people who have been my guides and mentors.