A thin HDF5 interface for parallel in-situ visualization within ParaView

John Biddiscombe[1], Jérôme Soumagne[2]

[1]CSCS - Swiss National Supercomputing Centre, biddisco@cscs.ch
[2]CSCS - Swiss National Supercomputing Centre, soumagne@cscs.ch

Abstract

As simulation codes become more powerful and more interactive, it is desirable to monitor the simulation *in-situ*, performing not only visualization, but analysis of the incoming data as it is generated - so that the simulation may be stopped, or modified, thereby conserving CPU resources. This talk describes an implementation of an in-situ framework which has been integrated into the ParaView visualization package alongside existing analysis modules.

The architecture is intended to address three principal objectives: Require little or no modification to the simulation code in order to allow a live visualization. Allow the simulation to be run on one parallel machine whilst the visualization is run on a separate (or the same) parallel machine. Provide good performance to ensure that massive simulations may be handled as easily as small test cases.

The interface developed is built around the HDF5 file IO library used commonly in HPC applications and a thin Xdmf layer [1]. The HDF5 API allows the derivation of custom *virtual file drivers* (VFDs) which may be instantiated at run-time on a per file basis to control how data is written to the file system. We have made use of this facility to create a specialized MPI based VFD which allows the simulation to write data in parallel to a file, but which is actually redirected over the network to a visualization cluster where the data is stored in a Distributed Shared Memory (DSM) buffer - in effect a (remote) virtual file system. The ParaView application acts as a server/host for this DSM and can read the file contents directly using the HDF5 API as if reading from disk. The transfer of data between simulation and visualization machines may be done using either an MPI based communicator shared between the applications, or using a socket based communication. The management of both ends of the network transfer is transparently handled by the DSM VFD layer, meaning that an application using HDF5 can make use of in-situ visualization without any code changes. It is only necessary to re-link the application against a modified version of the HDF library which contains our driver. However, the complexity of HDF5 is such that many different data layouts are possible and it is not feasible to read all HDF5 files using a single common standard reader. To mitigate this problem, we support several readers/formats within our ParaView plugin, including the H5Part particle format and the Xdmf format for other arbitrary HDF5 data. The user may provide a light XML template which is used to generate Xdmf description files automatically. Once data has been loaded, all existing ParaView filters and display methods are available and the user may setup an analysis workflow which is updated automatically as new data is received.

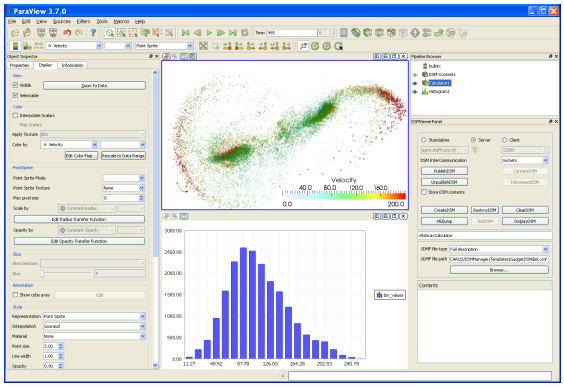

Figure 1: In-Situ Visualization of Gadget2 results in ParaView

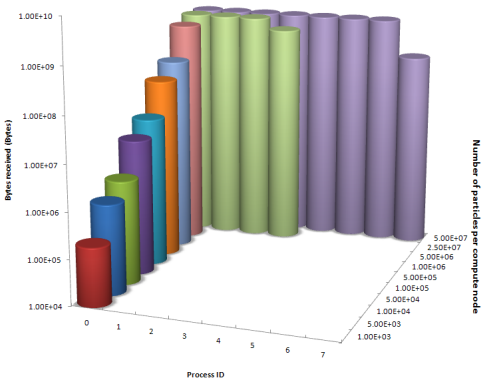

Figure 2: Distribution of data to parallel DSM nodes for fixed (10GB) DSM size

References

[1] Jerry A. Clarke and Raju R. Namburu. Simplifying Partitioning Complexities by Using a Common Data Hub. Technical report, U.S. Army Research Laboratory, 2002. ARL-TR-2799.