Distributed computational fluid dynamics

Karl Jenkins* Xiaobo Yang, Mark Hayes† Stewart Cant*

August 15, 2003

Abstract

Computational fluid dynamics simulations of relevance to jet-engine design, for instance, are extremely computationally demanding and the use of large-scale distributed computing will allow the solution of problems that cannot be tackled using current resources. It is often appropriate to leave the large datasets generated by CFD codes local to the compute resource in use at the time. This naturally leads to a distributed database of results that will need to be federated as a coherent resource for the engineering community. We describe the use of Globus and Condor within Cambridge for sharing compute resources, progress on defining XML standards for the annotation of CFD datasets and a distributed database framework for them.

1 Introduction

Computational fluid dynamics (CFD) is now widely used in both industry and academia to solve turbulent and reacting flows, such as those found in the combuster of a jet engine. Industry requires these CFD techniques to offer rapid turnaround as well as high accuracy, and this demands high-quality physical modelling of unresolved small-scale processes in the turbulence and combustion. Results from industrial scale CFD are most often in the form of large and complex datasets. Thus, remote access to this information is an integral part of the CFD development and design process.

Investigation of industrial problems implies that the CFD simulations need to be more demanding, both in terms of the physics to be simulated and the levels of spatial and temporal resolution required. The requirement for increased computer power is being met at a local level by clusters of powerful PC-type workstations and at university and national level by very large massively-parallel supercomputers. Therefore, CFD offers a major opportunity for the development and application of Grid technology in engineering and forms the motivation for the present study.

A difficulty that arises in these practical turbulent combustion processes is a strong coupling between turbulence, chemical kinetics and heat release. These interactions are generally three dimensional and time dependent, and are not easily accessible to experimental investigation.

*Cambridge University Engineering Department, kwj20@eng.cam.ac.uk,rsc10@eng.cam.ac.uk

†Cambridge eScience Centre, xy216@cam.ac.uk, mah1002@cam.ac.uk

Direct numerical simulation (DNS) is now a widely accepted technique for simulating flows which require resolution down to the smallest length scale. A parallel combustion DNS code called SENGA has been developed in Cambridge to study turbulent combustion. SENGA solves the governing equations for a fully compressible reacting flow using high order explicit finite difference methods and the parallel operation is implemented through the message passing interface (MPI).

Therefore, to explore the potential of the Grid for CFD applications, a mini-Grid system has been set up between the Cambridge University Engineering Department and the Cambridge eScience Centre. This system comprises two dedicated PC clusters and dedicated data storage machines. The link is made with a virtual private network (VPN), although this has not been fully tested yet. The VPN provides a route around the departmental firewalls. The clusters run Globus and Condor for remote job submission, file transfer and batch queue management. We have also developed a web based portal for remote job submission.

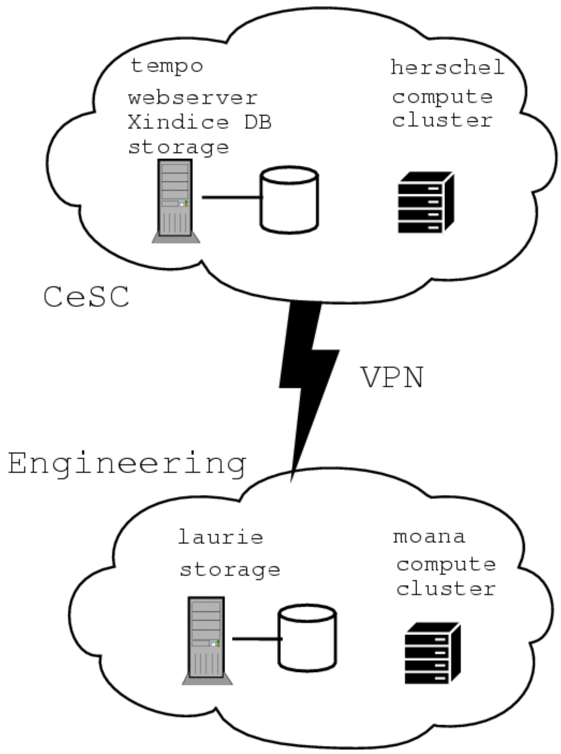

Figure 1 - Cambridge CFD mini-Grid

2 Web portal development & use of XML

The Globus Toolkit [1] is an open source, reference software base for building grid infrastructure and applications. Its use in this project allows access to appropriate local and remote compute resources for the purpose of running SENGA and retrieval of the output datasets. As Globus is currently based on a rather low level set of command line tools, a user friendly front end has been developed (a "portal") to guide the user through running the code. With web browsers being ubiquitous nowadays, a web interface is a natural choice for portal development.

The portal code developed for the CeSC project "Electromagnetic scattering from aircraft"[2] has been re-used here.

The first stage in submitting a job through the CFD portal is to specify a set of input parameters for the SENGA run. These parameters are saved as an XML file suitable for validation against a custom written schema for SENGA[3]. The Xerces-C++ [4] toolset is used for validation. If successful, the input parameters are also written to a plain text file suitable for input to the SENGA code itself. The job is then submitted to a remote compute cluster, either at CeSC or the Engineering Department.

Once the numerical simulation has finished, all output data is transferred back to one of several machines which act as file servers. The physical location of the output files is then recorded along with the input parameters in XML format. This information is stored in a native XML database (we are currently evaluating Apache Xindice [5]) for later query and retrieval. For example, it would be of interest to ask for the results of a run generated by a particular user on a particular date under certain initial conditions.

Multiple instances of this database framework may be installed by collaborators at remote sites (both academic and industrial), populated with their own datasets and federated using technology such as OGSA-DAI.[6]. This community resource will allow a wide range of tasks from interactive visualisation of remote datasets through to the collection of statistical data that will be of direct use in future models.

References

[1] http://www.globus.org/

[2] Visualisation & Grid applications of electromagnetic scattering from aircraft, M. Spivack et al, Proceedings of the 2nd UK eScience All Hands Meeting, 2003

[3] http://www.escience.cam.ac.uk/projects/cfd/senga.xsd

[4] http://xml.apache.org/xerces-c/

[5] http://xml.apache.org/xindice/

[6] http://www.ogsa-dai.org.uk/