CLRC

Technical Report
DL-TR-99-002

Survey of Computational Steering, Meta-computing and Network Information Tools

RJ Allan

July 1999

COUNCIL FOR THE CENTRAL LABORATORY OF THE RESEARCH COUNCILS

Enquiries about copyright, reproduction and requests for additional copies of this report should be addressed to:

The Central Laboratory of the Research Councils
Chadwick Library
Daresbury Laboratory
Daresbury
Warrington
Cheshire
WA4 4AD
Tel: 01925 603397 Fax: 01925 603195
E-mail library@dl.ac.uk

ISSN 1362-041X

Neither the Council nor The Laboratory accept any responsibility for loss or damage arising from the use of information contained in any of their reports or in any communication about their tests or investigations.

Parallel Application Software on High Performance Computers. Survey of Computational Steering, Meta-computing and Network Information Tools. *

R.J. Allan

Computational Science and Engineering Department,
CLRC Daresbury Laboratory,
Daresbury, Warrington WA4 4AD, UK

Email: r.j.allan@dl.ac.uk

This report is available from http://www.cse.clrc.ac.uk/Activity/HPCI

July 2, 1999

Abstract

A software environment of unprecidented quality and functionality is emerging in which coupled computing resources are accessed via client-server and Web-based tools. This is driven by a combination of the computer industry and the loose collection of worldwide "freeware" programmers. Geoffrey Fox has referred to it as the "Distributed Commodity Computing and Information System" – DcciS.

In this report we examine a number of tools and projects for science and engineering applications on wide-area network-based systems. This includes computational steering and meta-computing techniques.

Keywords: computational steering, meta-computing, network solvers, programming tools, network of workstations (NOWs), cluster computing.

Neither the Council nor the Laboratory accept any responsibility for loss or damage arising from the use of information contained in any of their reports or in any communication about their tests or investigations.

[*] Edition 1: June 1999

Contents

1 Introduction

Computational scientists are greedy for computer resources. Single very large computers, even if they are of DMM architecture, are limited – more often by cost than by technology. There are however many more modest systems who's total power far outweigh the few massive supercomputers available. It is this fact which has sparked an interest in harnessing distributed resources to create "computational power grids". At the departmental level such grids are built from workstations or PC "commodity" clusters which would otherwise be infrequently used but can be harnessed to obviate the need to purchase mid-range servers. At the national level they may be built by collaborating computer centres or university research groups. Activities at the international level involve government laboratories and large national centres.

Distributed software tools, and especially those which facilitate very complex "coupled" applications to be constructed and used are likely to be of growing interest over the coming few years. We have therefore carried out a survey of the current research and tools which are being constructed in this area. We do not however attempt to provide an introduction to the distributed-computing techniques which are both complex and diverse. There are numerous discussions in the computer-science literature which should be consulted for background information (see e.g. [27, 21]).

Distributed computing systems however offer more than just a large cpu resource. A software environment of unprecidented quality and functionality is emerging along with the use of the internet for leisure purposes. This is driven by a combination of the computer industry and the loose collection of worldwide "freeware" programmers. Fox has referred to this as the "Distributed Commodity Computing and Information System" – DcciS.

The simplest tools provide an "object broker" mechanism to access computational resources, for instance a numerical library on a remote platform. More sophisticated systems provide fault-tolerance and checkpointing for meta-computing experiments [15]. An example of software of this type is NetSolve. We do not discuss the broker software here, but the computer industry has emerging standards such as CORBA, Java JWORB, COM etc. and many references can be found via the Web. Two "middle layer" tools are GLOBUS (NCSA Alliance in the US) and Legion (US-NPACI Collaboration). These form the basis of many other tools and grids such as GUSTO.

Building systems that alter program behaviour during execution based on meta-computing techniques and user-specified criteria (computational steering) has recently become a research topic of particular interest in the US and Japan where there are several alliances of computing centres separated by large distances. In Europe, Germany has taken a lead because of the regional computing centres. In the UK the JREI-funded centres may be (but are not yet) a source of similar resources.

It may be thought that a discussion of computational steering does not fit well into a survey of distributed computing tools. Nevertheless the techniques are related in requiring methods to alter running processes, either to move them for the purpose of load balancing or to alter their execution path.

Steering and meta-computing tools require powerful visualisation facilities to run in a distributed computing environment with a distributed infrastructure (run-time system). Building such an infrastructure requires devising strategies for co-ordinating execution across machines (concurrency control mechanisms), mechanisms for fast data transfer between machines and mechanisms for user manipulation of remote execution.

A project at the Georgia Institute of Technology has developed the Falcon run-time monitoring system and the Progress and Magellan computation steering tools which use it to develop and control large-scale applications.

SCIRun (pronounced "ski-run") is a simulation-steering tool designed for shared-memory multiprocessors and now ported to a distributed environment at the University of Utah, Salt Lake City with funding from NCSA.

Other projects, such as NetSolve, are part of the NetLib activity of Dongarra et al.

1.1 Criteria for Inclusion

Packages covered here are those which can be run on either true parallel machines, workstation and PC clusters or on shared-memory systems. Very often machines of several types will be connected accross a wide-area network (perhaps via ATM). Whilst we have in the main restricted our attention to packages which are intended for use with FORTRAN77, Fortran 90, C and C++, since these are the most widely used languages for scientific applications, JAVA is playing a large role in this area, at least for the construction of network-enabled tools.

The main criterion for inclusion is that a package should be of use in a scientific or engineering application. Some of the entries cover packages which are already in existence and available. However, many packages are under construction, or proposed software projects, and are included if they are thought to be of sufficient interest.

1.2 Individual Entries

The different fields in the entry for each package should be self explanatory. Names and addresses given are simply somebody who can be contacted about the package; they are not meant to represent the entire cast responsible for the software. For full lists of the organisations and people involved the actual documentation (or Web page) should be consulted.

1.3 Intended Audience and Feedback

This survey is particularly geared towards users of the UK national academic computing facilities and for this reason also has a slight "UK slant", although most of the work has been done in the USA. However, the information contained here should also be useful to a wider audience.

It is our intention to keep this report as up-to-date as possible. To this end, we would be very keen to hear about any packages which are of interest in parallel scientific computing and are not currently included. Corrections and comments are also welcomed.

The author can be contacted by email via r.j.allan@dl.ac.uk.

2 Activities Worldwide

A number of consortia of research groups worldwide are developing large-scale distributed computing systems and meta-computing systems. These are also in many cases the focus of the software development described in the rest of this report. Some of the projects are first described.

2.1 Berkeley NOW and NOWII Projects

Network of Workstations (NOW) is a research system which started around 1995 and was designed as a dedicated high-performance platform. It consists of Sun UltraSparc systems linked by Myrinet. Key software includes GLUnix (middleware for queuing, gang scheduling and other resource management), xFS (scaleable parallel file system) and Active Messages (low-latency communications layer based on Thinking Machines CM-5 software and achieving 10 μs latency and 40 MB/s bandwidth).

NOWII extends the original project to provide multi-user access. A software framework WebOS is being constructed which will provide incrementally scaleable geographically aware Web service.

In the NOW and NOWII projects active messages are used for fast communication. An asynchronous communication mechanism is provided which uses a user-defined message handler invoked via an interrupt mechanism. This is very similar to the mechanism provided by Intel in their NX/2 operating system. This is implemented as GAM (Generic Active Messages).

For further information se Web URL
htmllinkhttp://now.cs.berkeley.edu. See also Hwang and Xu [21] for an introduction to the project and comparison with other systems.

2.2 Illinois HPVM Project

The goal of this project, which started around 1997, is to develop shared controllable high-performance components for distributed systems. This includes predictable communication, managementy of heterogeneity, stable performance models and adaptive resource management. Virtual reality is supported using high-speed networking and an ability to manipulate large data sets. A variety of compute and networking components are being evaluated.

Software includes Illinois Fast Messages with APIs to MPI, SHMEM and Global Arrays, Dynamic co-scheduling resource management, FM-QoS heterogeneous communication layer and front-end administration tools using Java.

Further information is available from Web URL

2.3 Real-World Computing Partnership

A project is under way to build distributed systems with shared resources. For further information contact:

K. Kubota et al.
Real-World Computing Partnership
Tsukuba, Ibaraki 305-0032, Japan

or see Web URL `http://www.rwcp.or.jp/lab/mpperf`.

2.4 GUSTO Consortium

GLOBUS distributed and meta-computing concepts are being tested on a global scale by participants of the Globus Ubiquitous Supercomputing Testbed Organization (GUSTO). This is an agreement between PACI sites to develop a meta-computing testbed. PACI is the NSF Partnership in Advanced Computational Infrastructure which includes most of the large computing centres in the USA.

GUSTO is based at Argonne National Laboratory and the Unversity of Southern California and started in 1997. It currently spans over twenty institutions and includes some of the largest computers in the world. Both dedicated and commodity internet services are used.

GUSTO is further described in the GLOBUS Web pages at URL
`http://www.globus.org`.

2.5 NASA Information Power Grid

The NAS IPG project is designed to implement seamless access to resources between NASA sites and a few NPACI sites. This followed from a number of workshops and reviews in the period Autumn 1997. Guals of the project are to provide access to all resources for a single large simulation and to include virtual reality and access to large-scale data stores. A number of middleware implementations and demonstrator applications are being developed in phase II of the project starting in 3Q99 and continuing until 3Q04.

Further information available the Web:
`http://www.nas.nasa.gov/Groups/Tools/IPG`.

There is also an "Information Power Grid Hotlist" from the NASA Web site which includes information on distributed computing, meta-computing and Java `http://www.nas.nasa.gov/NAS/Tools`.

2.6 ASCI Problem Solving Environment (PSE)

A major activity driven from Los Alamos National Laboratory is the Problem Solving Environment for the Accelerated Strategic Computing Initiative (ASCI). It also includes Lawrence Livermore National Laboratory and Sandia National Laboratory. Components of PSE relevant to network computing are:

- High Performance Computing Support: High Performance Computing Support's (HPCS) role is to provide a supporting infrastructure between platforms and applications for effective high-end application execution and tera-scale data management.
  o Archival storage,
  o Scientific data management,
  o High speed interconnect,
  o Scalable I/O,
  o Distributed resource management
  o Platform and service integration;

- Tri-Lab Networking: Designing and implementing this wide/local area network architecture that enables uniform, transparent, and efficient distributed classified and unclassified computing among the three defense programs laboratories continues to be a formidable technical and administrative task that involves every aspect of networking.
  o Tri-lab connectivity,
  o New secure service and encryption upgrades,
  o Network modeling;

- Distributed Computing Environment: The purpose of the Distributed Computing Environment team is to provide a common set of key core services throughout the ASCI community, common both inter-organisationally (within a single lab) and between the ASCI computing environments at each of the three laboratories.
  o Production DCE core services,
  o Tri-lab distributed services/support,
  o DCE secure web pilot
  o Tri-lab DFS deployment,
  o DFS/HPSS integration testing,
  o Expanded desktop deployment,
  o Distributed objects,
  o Assessment study of PKI and DCE,
  o ASCI application support.

The debugging and visualisation activities will be described in a separate report [1].

2.7 JAERI/STA

The CCSE was established within the Japanese Atomic Energy Research Institute (JAERI) in 1995. It is playing a leading role in the research and development of computational science and engineering in Japan. This is continuing the work started in the Science and Technology Agency (STA) and will continue to satisfy their requirements.

Principal strands of the research and development at CCSE are:

- development of parallel basic software;

- development of parallel algorithms;

- development of parallel processing tools;
- studies of numerical simulations on complex phenomena by particle and continuum methods;
- new computer architectures.

These feed into applications of special interest to the STA laboratories and Japanese Universities and software is available on JAERI and STA computers. Fortran 90 and MPI is used and the software is portable across many platforms including: Intel Paragon, Fujitsu VPP, Hitachi SR2201, Fujitsu AP3000, IBM SP, NEC SX4, Cray T90 etc.

A specific deliverable relevant to this report is the Metacomputing environment STAMPI and its accociated tools. Further information is available from the Web site at `http://www.jaeri.go.jp/english/index.cgicomp/comp.html`.

2.8 US Defense Modelling and Simulation Office

The Defense Modeling and Simulation Office (DMSO) High-Level Architecrure (HLA) project offers standards for an interoperability framework enforcing re-usability and shareability of objects and components based on new technology standards. The aim is to couple disjoint simulations on HPC systems and commodity clusters.

There are a number of parallel and distributed simulation tasks at Metron Incorporated that are sponsored by the High Performance Computing Modernization Office (HPCMO) through its Common HPC Software Support Initiative (CHSSI) project. These include: (1) modifying NSS to execute with high-performance on massively parallel machines, (2) extending the IMPORT simulation language and SPEEDES modeling framework to provide easier-to-use programming interfaces, and (3) developing a High-Level Architecture (HLA) Run-Time Infrastructure (RTI) for supporting interoperability between different simulations on high-performance computers. These efforts are being coordinated by the Naval Research Laboratory (NRL), Space and Naval Warfare Systems Command (SPAWAR), and the DMSO.

Another project sponsored by DMSO at Syracuse University is WebHLA.

The DMSO Web page on High level Architecure is at `http://hla.dmso.mil`. The Metron Web site is at `http://www.ca.metsci.com`.

2.9 NSCP

The National Scalable Cluster Project is developing a prototype meta-computing system including three university clusters in Chicago, Maryland and Pennsylvania. Goals are to develop software and demonstrate scalable clustered computing enabling data transfer between geographically remote sites.

vBNS (very Broad Network System) is used to construct a fast network. It uses ATM (Asynchronous Transfer Mode) protocols to achieve transfer speeds, sufficient to link nodes in local and wide area computing clusters, with the power to transfer Terabytes of data within minutes.

Other activities include data maning, data warehousing and medical supercomputing.

Contact: Prof. R. Hollebeek,
University of Pennsylvania, Department of Physics and Astronomy,
209 So. 33rd St., Philadelphia, PA, 19104, USA hollebeek@nscp.upenn.edu. Web URL is
http://nscp.upenn.edu.

3 Network-based Solvers and Information Systems

Computational "power grids" have recently become a hot topic. They do not necessarily give higher computing power for a given application than the largest parallel supercomputers, but they do enable maximal exploitation of existing resources. This is particularly important in the middle range of computing where many computers and clusters are linked over a wide-area network providing a very flexible environment and making use of systems which otherwise would only be lightly loaded. This however raises serious problems of security and authentification, and demands a level of collaboration rarely seen in facility management. Indeed this may become a major success of the technology.

A book on the subject [14], based on experiences with the GLOBUS project, appeared in July 1998 emphasising its growing importance. It contains a large amount of background material and descriptions of past and current projects by 30 expert authors.

There is also an "Information Power Grid Hotlist" from the NASA Web site which includes information on distributed computing, meta-computing and Java http://www.nas.nasa.gov/NAS/Tools.

Power grids provide more than just computational resources. They can also provide access to distributed information sources and instruments (e.g. on telescopes or synchrotron sources). They are fundamental in funding solutions to challenging problems such as smart instruments, collaborative engineering or data mining.

The first computational power grid was GUSTO, a project connecting 27 computer centres across the USA led by Ian Foster (ANL) and Carl Kesselman (University of Southern California). This is described in the GLOBUS entry (Section 4) and in [14].

NetSolve is a large research project driven from the University of Tennessee, Knoxville and Oak Ridge National Laboratory by Jack Dongarra et al. Other network software includes NEOS and NetLink. The NetLink project has similar goals fo NetSolve but via the NetLink access agent. The NEOS network package, is specifically designed for optimisation problems. Several other projects are in the pipeline, but being collaborations of major computing centres are on the national scale and mainly originating in the USA.

3.1 IPG

Name: IPG – Information Power Grid

Description: IPG is envisaged as a unified collection of geographically dispersed supercomputers, storage devices, scientific instruments, workstations and advanced user interfaces. The project has been proposed by a team from NASA research centres, NCSA and NPACI. Applications are likely to include aeronautics and other areas of interest to NASA such as space sciences and earth sciences.

Systems:

Contact: Bill Johnston, project manager,
Lawrence Berkeley Laboratory, USA

Email:

URL/FTP: see `http://science.nas.nasa.gov/Groups/Tools/IPG`

Comments:

References:

3.2 NEOS

Name: NEOS – Network-Enabled Optimization System

Description: The Optimization Technology Center (OTC, see `http://www.mcs.anl.gov/home/otc`) is a joint project between the US Argonne National Laboratory and North-Western University which started in 1994. The mission of the centre is to widen the community awareness of optimisation techniques and to promote the use of such techniques. NEOS is a WWW interface to the software and computational resources at the OTC and is provided as three levels: a server; a guide (decision tree); and a set of tools. NEOS currently solves usual and stochastic problems and linear network optimisation problems. The NEOS guide also contains information about software packages and case studies.

Systems:

Contact:

Email:

URL/FTP: `http://www-unix.mcs.anl.gov/neos/Server`
`http://www-unix.mcs.anl.gov/otc/Guide`
`http://www-unix.mcs.anl.gov/otc/Tools`

Comments:

References: [12]

3.3 NetLink

Name: NetLink

Description: has the objective to find a data distribution architecture that, in a distributed manner, can help to centralise library maintenance and tuning. Uses an “access agent” component for a variety of software and target hardware and to provide authentification via secure domains. Uses a sophisticated cache mechanism for object searches.

Systems:

Contact: I. Holmqvist and E. Lindström,
Umea University, S-90187 Umea, Sweden

Email: `dpiht@cs.umu.se`

URL/FTP: `http://www.hpc2n.umu.se`

Comments: prototype only. Uses resources at HPC2N.

References: [20]

3.4 NetSolve

Name: NetSolve

Description: uses a client-server-agent software architecture to harness loosely-coupled systems on a network. NetSolve is intended to provide transparent access to a whole variety of software libraries, highly tuned for the target architecture. This improves maintainability of software and avoids the end user having to download and compile it.

NetSolve is implemented as a three-tiered system:

1. a client – specifies the problem. May be a C or Fortran program linked to the NetSolve library, Mathematica sessions, or Java applets calling the NetSolve library. A Java GUI is also provided;
2. agents – C programs running as daemons act as resource brokers;
3. servers – registered with agents and can perform certain services (e.g. have particular applications installed). Provide optimal computation environment for their particular architectures.

At the API level NetSolve looks like a high-level library with a single function call `netsolve`. Character strings are introduced to specify the required action. A non-blocking version is also available, but the user has to take care of resource usage and determinism.

NetSolve currently has an interface to ScaLAPACK and related components.

Systems: uses Condor [25] for its distributed computing management

Contact: J. Dongarra and H. Casanova,
University of Tennessee, Knoxville, TN 37996, USA

Email: `dongarra@cs.utk.edu` or `casanova@cs.utk.edu`

URL/FTP: available from NetLib `http://www.netlib.org`

Comments: currently a prototype

References: [10]

3.5 Ninf

Name: Ninf – Network-based Information Library for Global World-wide Computing Infrastructure

Description: Ninf is an ongoing global network-wide computing infrastructure project which allows users to access computational resources including hardware, software and scientific data distributed across a wide area network with an easy-to-use interface. Ninf is intended not only to exploit high performance in network parallel computing, but also to provide high-quality numerical computation services and accesse to scientific databases published by other researchers. Computational resources are shared as Ninf remote libraries executable at a remote Ninf server. Users can build an application by calling the libraries with the Ninf Remote Procedure Call, which is designed to provide a programming interface similar to conventional function calls in existing languages, and is tailored for scientific computation. In order to facilitate location transparency and network-wide parallelism, the Ninf meta-server maintains global resource information regarding computational server and databases, allocating and scheduling coarse-grained computation to achieve good global load balancing. Ninf also interfaces with existing network services, such as the Web, for easy accessibility. Ninf IDLs have so far been defined for LAPACK, LibSci and other numerical libraries and databases. There is a major project to develop a CFD modelling network.

Systems: Cray J90, SUN UltraSparc and SparcStation 20, NOW clusters

Contact: Ninf Administration Group,
Electrotechnical Laboratory,
Umezono, Tsukuba 305, Japan

Email: `ninf@etl.go.jp`

URL/FTP: `http://ninf.etl.go.jp`

Comments: The Ninf Global Computing Network is accessed via URL `http://pdplab.trc.rwcp.or.jp`. Ninf is an on-going research project. Note ETL also maintains a useful mirror site for numerical algorithms and high-performance computing at `http://phase.etl.go.jp`

References: [31]

3.6 PSUE

Name: PSUE – Parallel Simulation User Environment

Description: The Parallel Simulation User Environment (PSUE) provides the capability for rapid solution of highly diverse computational field simulation problems through the use of unstructured grid technology. The environment will reduce problem turn around time by linking all stages of the analysis into one common environment.
The PSUE includes the following capability:
o geometry builder;
o geometry repair;
o unstructured grid generation;
o grid quality analysis;
o remote/parallel platforms;
o post-processing and data analysis;
o help facilities;
o application integration.

Systems: The PSUE has been developed using X, OSF Motif and OpenGL library routines, which are available across most UNIX platforms. These routines give a consistent, modular feel to the graphical interface.

Contact: Prof. N. Weatherill
Dept. Civil Engineering
University of Wales, Swansea
Singleton Park, Swansea SA2 8PP, UK
and
I.C. Risk
British Aerospace (Operations) Ltd.
Sowerby Research Centre
Filton, Bristol BS12 7QW, UK

Email: `n.p.weatherill@swansea.ac.uk` or `ian.risk@src.bae.co.uk`

URL/FTP: `http://www.telecall.co.uk/~srcbae/exploit.htm`

Comments: part of the EU Caesar (FP3 number 8328) and Julius (FP4 number 25050) projects

References:

The geometry builder is capable of importing CAD data which can be modified and simple geometrical entities created using point, line and surface creation. A geometry repair facility overcomes topological inconsistencies correcting surface overlaps and gaps. Each of these tools prepares the data for the unstructured grid generation module which uses a Delaunay triangulation algorithm to efficiently generate 2D planar/3D surface triangles and 3D volume tetrahedra. Grid point density is controlled using boundary point distribution and point, line and triangular sources either imported or created using the geometry builder. Grid cosmetics are incorporated to improve grid quality which can be examined using histogram and visual techniques. Numerical libraries are available to create element/side/face based data. The ability to utilise remote and parallel platforms allows the use of the most suitable machine size/type for specific operations, particularly, grand challenge problems. The incorporation of links to visualisation packages AVS and ENSIGHT allow easy access to the wealth of post-processing facilities available in commercial software.

Application integration allows the user the flexibility to incorporate their own, commercial or public domain software into the environment. This can be performed at many levels of operation through user defined script files. Recompilation of the PSUE is not required, providing a fast and efficient method of consolidating all the user's software together into a single package. Once integrated data may be sent to and from applications via a data transfer interface which may use file, pipe or socket transfer.

3.7 UNICORE

German resource-sharing project for BMBF (Bunderministerium für Bildung und Forschung) supercomputer centres. No further information found at present.

3.8 WebHLA

Name: WebHLA – Web-based High-Level Architecture

Description: DoD project to develop distributed computing services via commodity systems. An interactive programming and training environment for high-end computing. Uses a three-tier approach implemented in JWORB middleware over the GLOBUS metacomputing or NT cluster back end:
1. front end – WebFLOW enables visual collaborations and visual authoring tools connect meta-computing application;
2. distributed objects – WebFLOW servers acting as proxies to computer systems;
3. back end – WebFLOW clients.
This follows the philosphy of the *Pragmatic Object Web* of Fox et al.

Systems:

Contact: G.C. Fox and W. Furmanski
Syracuse University, Syracuse, NY, USA

Email: `gcf@npac.syr.edu` or `furm@npac.syr.edu`

URL/FTP: `http://www.npac.syr.edu`

Comments: currently a prototype, but an early application was demonstrated at Supercomputing'98.

References: [15, 27]

4 Distributed Application Management Tools

This section describes some of the "middle-ware" tools which can be used to develop applications using distributed resources. They are used by many of the other tools in this report, except the ones which have their own client-server components.

Issues for middleware software are: security; authentification; system faliure recover; checkpointing; system availablity; load balancing; job management and process migration. An overview is provided by Hwang and Xu [21].

Agents, Software Agents, Intelligent Mobile Agents (IMAs) and Softbots are terms used to describe the concept of mobile computing or mobile code. See Bradshaw [6]. An example of agent technology is D'Agents from Dartmouth College, USA.

4.1 ASCI Distributed Systems

Name: ASCI Distributed Systems project

Description: The Distributed Systems portion of the ASCI Problem Solving Environment consists of the networking, Distributed Computing Environment (DCE), and Distributed Resource Management (DRM) activities. These three research and development activities are very interdependent and are fundamental to the creation of the basic infrastructure for the ASCI environment.

For the past two years, the DCE and networking projects in ASCI have been tightly coupled. For example, the DCE goal of providing a common cross-cell authentication capability that will enable a user's single authentication to be honored on all ASCI computing platforms and servers is key to removing major networking barriers and achieving the necessary network throughput performance.

The DRM activities are closely tied to the DCE project in that:
(1) DRM depends on DCE for authentication,
(2) DRM will offer access to DFS from batch jobs,
(3) work is under way to manage the lifetimes of DCE credentials for use by batch jobs, and
(4) strong consideration is being given to replacing the current socket-based communication in DPCS at LLNL with DCE RPCs.

In the future, networking and visualisation resources will need to be managed and scheduled. Therefore, the DRM project will require an even greater awareness of networking and visualisation issues in order to begin to include these elements in the resource management strategy.

In terms of correspondence to other ASCI programs, the Distributed Systems activities are closely aligned with most of those PSE research projects that directly relate to the Distance Computing portion of DisCom2, and to a lesser extent to the Distributed Computing portion. Similarly, the Distributed Systems activities relate well to NEWS visualisation corridor efforts. By administratively grouping networking, Distributed Computing Environment and Distributed Resource Mangement. Coordination between these three closely related and interdependent activities is being increasde.

Systems: ASCI tri-lab platforms

Contact: Bob Tomlinson, LANL or
Barry Howard, LLNL or
Doug Brown, SNL

Email: `bob@lanl.gov` or `bhoward@llnl.gov` or `cdbrown@sandia.gov`

URL/FTP: `http://www.llnl.gov/asci/pse/ds/dist_sys.html`

Comments:

References:

The Tri-Lab distributed computing team (a consortium of computer scientists from LLNL, LANL and SNL) is collaborating to coordinate:

o implementations of the DCE and DFS technologies from the The Open Group;
o security plans and MOUs for cross-cell trust between sites;
o with other PSE components and platforms for a well integrated solution;
o with other collaboratory efforts, including the ESnet DCE Working Group;
o "early on" with users to establish testbeds and evaluate technologies.

4.2 Codine

Name: Codine v4.2

Description: Codine is a Resource-Management System targeted to optimise utilisation of all software and hardware resources in a heterogeneous networked environment. The easy-to-use graphical user interface provides a single-system image of all enterprise-wide resources for the user and also simplifies administration and configuration tasks. The tools originated as public-domain projects in the USA but ar now marketed by Genias Software GmbH in Germany.
Codine contains components for:
o resource license management;
o heterogeneous distributed computing;
o SMP computing support;
o job queue management;
o fault tolerance;
o checkpointing and migration.

X11 and Motif GUI interfaces are provided for programming and system management.

Systems: DEC Unix, SUN SunOS and Solaris, Parsytec, HP, SGI, IBM AIX

Contact: Genias Software GmbH,
Erzgebirgstrasse 2,
93073 Neutraubling, Germany

Email: `mbox@genias.de`

URL/FTP: `http://www.genias.de/products/codine`

Comments: Full documentation and technical description if available from the Genias Web pages.

References:

4.3 Condor

Name: Condor v6.1.5

Description: The goal of the Condor project is to develop, implement, deploy, and evaluate mechanisms and policies that support High Throughput Computing (HTC) on large collections of distributively-owned computing resources. Guided by both the technological and sociological challenges of such a computing environment, the Condor Team has been building software tools that enable scientists and engineers to increase their computing throughput.
Condor manages processes in a pool of workstations. Provides transparent checkpointing and restart facilities so that computations can be moved from over-loaded or failed machines to lightly-loaded ones. Current limitations (not unique to Condor!) include:
o only migrates processes between machines of the same architectures;
o only migrates processes within its own server;
o only works with serial (single-process) programs;
o system calls always executed on "host" machine.

Systems:

Contact: M. Livny
University of Wisconcin, USA

Email: `http://www.cs.wisc.edu/condor`

URL/FTP:

Comments: used as a middle layer in NetSolve, see separate entry

References: [25, 8]

4.4 D'Agents

Name: D'Agents – Dartmouth Agents

Description: a mobile agent is a program that can migrate from machine to machine in a heterogeneous network. The program chooses when and where to migrate. It can suspend its execution at an arbitrary point, transport to another machine and resume execution on the new machine. In the picture below, an agent carrying a mail message migrates first to a router and then to the recipient's mailbox. The agent can perform arbitrarily complex processing at each machine in order to ensure that the message reaches the intended recipient.
Mobile agents have several advantages over the traditional client/server model:
o Efficiency: Mobile agents consume fewer network resources since they move the computation to the data rather than the data to the computation.
o Fault tolerance: Mobile agents do not require a continuous connection between machines.
o Convenient paradigm: Mobile agents hide the communication channels but not the location of the computation.
o Customisation: Mobile agents allow clients and servers to extend each other's functionality by programming each other.
There are alternative techniques which have many of these same advantages such as queued RPC, proxy servers, etc. The problem with these alternative techniques is that each one is only suitable for certain applications. A mobile-agent system on the other hand is a single, unified framework in which a wide range of distributed applications can be implemented easily and efficiently.

Systems:

Contact: Prof. G. Cybenko
Thayer School of Engineering, Dartmouth College, USA

Email: `rgray@cs.dartmouth.edu`. There is also an un-moderated majordomo mailing list for users.

URL/FTP: `http://www.cs.dartmouth.edu/~agent` and `http://www.actcomm.dartmouth.edu`

Comments: source code and documentation is available from the Web page

References: [7]

A mobile-agent system called D'Agents is under development at Dartmouth College. The ultimate goal of D'Agents is to support applications that require the retrieval, organization and presentation of distributed information in arbitrary networks. Some of the research areas are:

o Security mechanisms
o Support for mobile and partially connected computers
o Navigation, network sensing and resource discovery tools
o Automatic indexing, retrieval and clustering techniques for text and other documents D'Agents is used in several information-retrieval and workflow applications.

Other notable mobile agent systems include:

- Telescript and Odyssey from General Magic
- ARA (Agents for Remote Access)

- TACOMA
- IBM Aglets

4.5 GLOBUS

Name: GLOBUS v1.0.0 (29/10/98)

Description: The Globus project is developing basic software infrastructure for computations that integrate geographically distributed computational and information resources.
GLOBUS is a joint project of Argonne National Laboratory and the University of Southern California's Information Sciences Institute. Project team includes groups at Argonne, USC/ISI, and the Aerospace Corporation, with significant contributions also being made by other partners.
Core components include:
o GRAM – globus resource allocation and process manager;
o Nexus – heterogeneous communication infrastructure, supports unicast and multicast;
o MDS – meta-computing directory information services, structure and state information;
o GSI – authentification and related security services;
o GASS and GEM – global access to secondary storage and executable management;

GLOBUS concepts are being tested on a global scale by participants in the GUSTO Consortium of NPACI sites.
GLOBUS is also the first software on the TransPAC network which links the Asia Pacific Advanced Network (APAN) and vNBS academic research networks, see URL `http://www.transpac.org`. Scientific applications will be ported as the system is build throughout 1999.

Systems: Ian Foster and Carl Kesselman
Argonne National Laboratory, USA University of Southern California, USA

Contact:

Email:

URL/FTP: `http://www.globus.org`

Comments: a useful tutorial is provided by Kesselman at URL
`http://www.npaci.edu/Training/NPACI_Institute98/Presentations/kesselman`. Globus is featured in the "Grids" book [14].

References: [13, 14, 8]

4.6 Hector

Name: Hector

Contact: J.S. Steinman
Metron Incorporated, 512 Via de la Valle, Site 301
Solana Beach, CA 92075, USA

Email: steinman@ca.metsci.com

URL/FTP: http://www.ca.metsci.com/speedes

Comments:

References: [4]

4.10 WOS

Name: WOS – Web Operating System

Description: an incremental approach to demand-driven services provision on heterogeneous and dynamic resources. Communication replaces the notion of servers. Different versions of services and WOS may be running on a given network. Services are provided by "education engines" and "warehouses" connected by a discoverl/ location protocop (WOSRP). THis can be implemented using CML, CBL or CORBA.

Systems:

Contact:

Email:

URL/FTP:

Comments:

References: [24]

5 Computational Steering

Program steering has been defined as the capacity to control the execution of long-running, resource-intensive programs. This may include modifying program state, managing data output, starting and stalling program execution, altering resource allocations etc. Dynamic steering requires the user to monitor program or system state and have the ability to make changes. This could be through subroutine calls added as "instrumentation" or by interacting with the code's data structures. An extensive survey of research in this area was carried out in November 1994 by Gu et al. [18]. however not many of the projects led to practical tools. In the intervening four years more progress has been made, and we describe just a few of the current projects here.

5.1 CUMULVS

Name: CUMULVS

Description: allows the scientist to modify a fixed set of parameters while using AVS to visualise the computational model. Implemented as a user-programmed AVS module. Has been used to distribute CFD applications. Supports collaborative visualisation and simulation allowing several viewers to "plug in" and steer. A checkpointing facility allows cross-platform migration and heterogeneous restarts, so may be of interest in meta-computing.

Systems: a middle later between PVM and AVS

Contact: Oak Ridge National Laboratory,
USA

Email: available from NetLib
`http://www.netlib.org`

URL/FTP:

Comments:

References: [16]

5.2 FALCON

Name: FALCON

Description: This is a monitoring system with low monitoring latency and perturbation. Steerable applications are developed through source-code modifications using Progress or Magellan (see below) and steering is assisted by the run-time system. FALCON captures many of the same things that a debugger would, and modifying variables enables you to steer your program while it is executing. Provides hooks to visualisation systems such as Iris Explorer. Can also monitor applications at the thread level, but a Cthreads package is required (which is available).
Uses DataExchange and PBIO for event transport in a heterogeneous environment and event filtering and processing.
Falcon, Progress and Magellan form part of a larger project to create "distributed laboratories" which is described on the Web pages.

Systems:

Contact: Karsten Schwan,
Georgia Institute of Technology, USA

Email:

URL/FTP: `http://www.cc.gatech.edu/systems/projects/FALCON`

Comments: Not yet available for public release

References: [19]

5.3 Progress

Name: Progress

Description: steerable applications are developed through source-code modifications and steering is assisted by a run-time system. Progress assists scientists to develop steerable applications by instrumenting their code with library calls and using "steerable objects" which can be altered at run time. The latter include sensors, actuators, probes, function hooks, complex actions and synchronisation points – in fact many of the same concepts found in VR applications. Has been used for MD simulations.

Systems:

Contact: Jeffrey Vetter,
Georgia Institute of Technology, USA

Email:

URL/FTP: `http://www.cc.gatech.edu/systems/projects/steering`

Comments: uses the FALCON run-time system for on-line monitoring (see separate entry)

References: [33]

5.4 Magellan

Name: Magellan

Description: derived from the Progress system (see above). Extends the client and server steering models. Uses a specification language ASCL to provide commands for monitoring and steering using the same objects as Progress. The application code must still be instrumented. Has been used for MD simulations.

Systems:

Contact: Jeffrey Vetter,
Georgia Institute of Technology, USA

Email: `http://www.cc.gatech.edu/systems/projects/steering`

URL/FTP:

Comments: uses the FALCON run-time system for on-line monitoring (see separate entry)

References: [34]

5.5 SciRun

Name: SciRun – Scientific Computing and Imaging

Description: A scientific problem-solving environment (PSE) which provides the ability to interactively guide or steer a running computation. The entire process of computation modelling, simulation and visualisation is built and executed within the PSE. SciRun was designed initially for multi-threaded shared-memory multiprocessors using C++ classes. A distributed-memory version is being produced and threading is now used to hide latency and perform other tasks.
Applications may be composed from new and existing components which are C++ classes describing geometries, meshes, fields, surfaces etc. Current applications are based on 3D tetrahedral unstructured meshes and SciRun defines the format of the data structure with which it interacts. A hexahedral mesh interface is being developed. The Diffpack, SparseLib++ and PETSc libraries are currently included. Components are linked in a dataflow network familiar to AVS Express users. All components support steering which is implemented in three distinct ways in the system:
o direct lightweight parameter changes – affect a running module;
o cancellation – when a parameter change cancels and re-starts a module;
o feedback loops – changes to parameters require other modules to be re-run.
Meta-computing is supported. SciRun aims to address the problems of interaction and integration of scientific simulation and visualisation in a distributed computing environment.

Systems: uses AVS Express on an SGI for control and visualisation

Contact: S.G. Parker, D.M. Weinstein and C.R. Johnson
University of Utah, Salt Lake City, UT 84112, USA

Email: `crj@cs.utah.edu`

URL/FTP: `http://www.cs.utah.edu/~sci`

Comments: funding from the NCSA PACI Alliance and venture capital. May become a commercial product in time.

References: [26, 3]

5.6 VASE

Name: VASE

Description: system presents an abstraction for a steerable program and offers tools that create and manage collections of steerable codes. VASE annotates existing Fortran code to create a high-level model of the application. Users therefore do not have to work at the source code level. Software developers must however annotate the existing code. Once this has been done CASE coordinates the execution of codes in a distributed environment under and SPMD execution model. A powerful C-like scripting language provides flexible support for data selection and steering during execution.

Systems:

Contact:

Email:

URL/FTP: none found

Comments:

References: [22]

6 Meta-computing Tools

Whilst a number of the tools and collections of software described in the previous sections are of use in a meta-computing environment [11] only a couple of tools seem to have been specifically designed for this purpose.

6.1 CARMI

See [29]. No further information has been found so far.

6.2 GLOBUS

Name: GLOBUS v1.0.0 (29/10/98)

Description: we provided a separate entry on the GLOBUS distributed middleware in Section 4. Part of the GLOBUS project also focusses on meta-computing using facilities at the GUSTO Consortium sites. High-level services include:
o MPICH-G and PAWS – grid-enabled MPI communications libraries. MPICH-G uses Nexus for heterogeneous communication (see previous entry);
o CC++ and HPC++ – parallel languages;
o grid-enabled libraries to provide uniform programming environment;
o remote-access and visualisation;
o DUROC and Nimrod – resource brokers;
o graphical status monitors.
The meta-computing directory service (MDS) maintains lists of resource objects in a distributed directory. Information can be updated from the Globus system, other information providers and tools and from applications. Information is provided dynamically to tools and applications. A lightweight directory access protocol has been developed. MDS tools include object class browser, explorer, various APIs and search tools and translators from GLOBUS object definition language.

Systems: Ian Foster and Carl Kesselman
Argonne National Laboratory, USA University of Southern California, USA

Contact:

Email:

URL/FTP: http://www.globus.org

Comments: a useful tutorial is provided by Kesselman at URL http://www.npaci.edu/Training/NPACI_Institute98/Presentations/kesselman. Also featured in the "Grids" book [14]

References: [13, 14, 8]

6.3 STA

Name: STA – Seamless Thinking Aid

Description: from the Centre for Promotion of Computational Science and Engineering (CSSE), part of JAERI, Japan. It is a Web-enabled Java-based environment which includes a number of tools for assisting parallel programming and using computers connected by networks.
The goal is to allow larger calculations and to couple applications with different memory or architectural requirements.
Editors, parallelising compilers, debuggers and performance tuning tools are able to exchange data in a seamless way. The kpx performance monitor is described in [1].
In order to use a heterogeneous computing resource a modified version of MPI called STAMPI is provided. This is able to dynamically allocate child processes to computers, provides a varying number of message routers to optimise communications, uses VSCM or CCMI for internal or external communication respectively and uses the MPI-2 standard programming interface.
STAMPI has been used to link together e.g. vector and RISC-based systems (NEC and IBM) in coupled 3D CFD and structural mechanics calculations for aircraft modelling.

Systems:

Contact: Hideo Kaburaki,
CCSE, JAERI, Tokai-muar, Naka-gun,
Ibaraki 319-11, Japan

Email: kaburaki@sugar.tokai.jaeri.go.jp

URL/FTP: http://www.jaeri.go.jp/english/index.cgicomp/comp.html

Comments:

References:

7 Acknowledgements

The preparation of this report was funded by EPSRC partly through a grant GR/K82635 to the CLRC HPCI Centre at Daresbury Laboratory and partly through its Service Level Agreement with the CSE

Department and the Distributed Computing Support Programme DisCo.

References

[1] R.J. Allan, R.R. Ward and M.C. Goodman *Survey of Parallel Performance Tools and Debuggers* (CLRC Daresbury Laboratory, 1999) draft

[2] P. Arbenz, W. Gander and M. Oettli *The Remote Computational System* Lecture Notes in Computational Science 1067 pp662-7 (Springer, 1996)

[3] E. Arge, A.M. Bruarset, H.P. Langtangen (eds.) *Modern Software Tools for Scientific Computing* (Birkhäuser, 1997) ISBN 0-8176-3974-8 and 3-7643-3974-8

[4] J.S. Steinman, G. Berliner, G.E. Blank, J.S. Brutocao, J. Burckhardt, M. Peckham, S. Shuper, K. Stadsklev, T. Tran, R.V. Iwaarden, L. Yu *The SPEEDES-based run-time Infrastructure for higl-level Architecture on high-performance Computers* in "High Performance Computing 1999" A. Tentner (ed.) (Society for Computer Simulation International, 1999) pp 255-66 ISBN 1-56555-166-4

J. Wallace, L. Petersen, G. Leonard, J. Celano, J. Brutocao and J. Caldwell *Import v2.0 beta 1: a Tool for large-scale, complex System Simulation* in "High Performance Computing 1999" A. Tentner (ed.) (Society for Computer Simulation International, 1999) pp 267-72 ISBN 1-56555-166-4

[5] D. Bernholdt, G.C. Fox, W. Furmanski, B. Natarajan, H.T. Ozdemir, Z.O. Ozdemir and T. Pulikal *WebHPC - an interactive Programming and Training Environment for High Performance Modelling and Simulation* in Proc. DoD HPC'98 Users' Group Conf. (Rice University, Houston, TX, June 1998) available from URL
`http://www.npac.syr.edu/iwt98/pm/documents`

[6] J.M. Bradshaw *Software Agents* (AAAI Press, the MIT Press, Menlo Park, CA, 1997)

[7] B. Brewington, R. Gray, K. Moizumi, D. Kotz, G. Cybenko and D. Rus *Mobile agents in distributed information retrieval* In Intelligent Information Agents, chapter 12. M. Klusch (ed.) (Springer-Verlag, 1999) To appear as ISBN 3-540-65112-8.

W. Caripe, G. Cybenko, K. Moizumi and R. Gray *Network awareness and mobile agent systems* IEEE Communications Magazine 36 (1998) 44-9

D. Kotz and R.S. Gray *Mobile agents and the future of the Internet* ACM Operating Systems Review 33 (1999) 7-13

[8] R. Buyya (ed.) *High Performance Cluster Computing: Architecures and Systems, Volume 1* (Prentice Hall, NJ, USA, 1999)

R. Buyya (ed.) *High Performance Cluster Computing: Programming and Applications, Volume 2* (Prentice Hall, NJ, USA, 1999)

For more information, please see:
`http://www.dgs.monash.edu.au/~rajkumar/cluster`
`http://www.phptr.com/ptrbooks/ptr_0130137847.html`
`http://www.phptr.com/ptrbooks/ptr_0130137855.html`

[9] G. Fox, W. Furmanski, M. Chen, C. Rebbi and J. Cowie *WebWork: Integrated Programming Environments Tools for National and Grand Challenges* NPAC Technical Report SCCS-0715 (Syracuse University, USA, 1995)

[10] H. Casanova and J. Dongarra *NetSolve: A Network-enabled Server for solving Computational Science Problems* Int. J. Supercomputer Applications and High-performance Computing 11 (1997) 212-23

J.S. Plank and H. Casanova and M. Beck and J. Dongarra *Deploying Fault-tolerant and Task Migration with NetSolve* in "Applied Parallel Computing" Proc. 4th International Workshop PARA'98. B. Kagström, J. Dongarra, E. Elmroth and J. Waśniewski (eds.) LNCS 1541 pp418-32 (Springer, 1998) ISBN 3-540-65414-3

H. Casanova and J. Dongarra *NetSolve's Network-enabled Server: Examples and Applications* IEEE Computational Science and Engineering (1998)

[11] C. Catlett and L. Smarr *MetaComputing* Comm. ACM 35 (1992) 44-52

[12] J. Czyzyk, M. Mesnier and J. Moré *NEOS: The Network Enabled Optimization System* Technical Report MCS-P615-1096 (Argonne National Laboratory, 1996)

[13] I. Foster and K. Kesselman *GLOBUS: a Metacomputing Infrastructure Toolkit* Int. J. Supercomputing Applications (1997)

[14] I. Foster and C. Kesselman (eds.) *The Grid: Blueprint for a new Computing Infrastructure* (Morgan Kaufmann Publishers, 1998) ISBN 1-55860-475-8. Abstracts of chapters and ordering information from URL
`http://www.mkp.com/grids`

[15] G.C. Fox and W. Furmanski *Petaops and Exaops: Supercomputing on the Web* IEEE Internet Computing 1 (1997) 38-46. Also available via the Web at URL
`htt://www.npac.syr.edu/users/gcfpetastuff/petaweb`

G.C. Fox, W. Furmanski, T. Haupt, E. Akarsu and H. Ozdemir *HPcc as High-Performance Commodity Computing on top of Integrated Java, CORBA, COM and Web Standards* in "Euro-Par'98: Parallel Processing" D. Pritchard and J. Reeve (eds.) LCNS 1470 pp55-74 (Springer 1998) ISBN 3-540-64952-2

G. Fox and W. Furmanski *Java for Parallel Computing and as a general Language for Scientific and Engineering Simulation and Modelling* Concurrency: Practice and Experience 9 (1997) 415-26

G. Fox et al. *Using WebHLA to integrate HPC FMS Modules with Web/ Commodity based Distributed Objects Technologies of CORBA, Java, COM and XML* in "High Performance Computing 1999" A. Tentner (ed.) (Society for Computer Simulation International, 1999) pp 273-8 ISBN 1-56555-166-4

[16] G.A. Geist, J.A. Kohl and P.M. Papadopoulos *CUMULVS: Providing Fault Tolerance, Visualisation and Steering of Parallel Applications* SIAM (August 1996)

[17] A. Grimshaw, W. Wulf, J. French, A. Weaver and P. Jr. Reynolds *A Synopsis of the Legion Project* Technical Report CS-94-20 (University of Virginia, 1994)

A.S. Grimshaw, W.A. Wulf, J.C. French, A.C. Weaver and P.F. Reynolds *Legion: the next Logical Step toward a Nationalwide Virtual Computer* Technical Report CS-94-21 (University of Virginia, 1994)

[18] W. Gu, J. Vetter and K. Schwann *An annotated Bibliography of Interactive Program Steering* SIGPLAN Notices 29 (1994) 140-8 and Technical Report GIT-CC-94-15 (Georgia Institute of Technology, 1994) available from Web URL
`file://ftp.cc.gatech.edu/pub/tech_reports/1994/GIT-CC-94-15.ps.Z`

[19] W. Gu, G. Eisenhauer, E. Kramer, K. Schwan, J. Stasko and J. Vetter *FALCON: On-line Monitoring and Steering of Large-scale Parallel Programs* in Proc. 5th Symposium of the Frontiers of Massively Parallel Computing (February 1995) 422-29 and Technical Report GIT-CC-94-21 (Georgia Tech 1994) from Web URL
`file://ftp.cc.gatech.edu/pub/tech_reports/1994/GIT-CC-94-21.ps.Z`

W. Gu, G. Eisenhauer and K.Schwan *FALCON: On-line Monitoring and Steering of Parallel Programs* Concurrency: Practice and Experience (1998)

[20] E. Holmqvist and E. Lindström, *NetLink: A Modern Data Distribution Approach Applied to Transparent Access of High Performance Software Libraries* in "Applied Parallel Computing" Proc. 4th International Workshop PARA'98." B. Kagström, J. Dongarra, E. Elmroth and J. Waśniewski (eds.) LNCS 1541 pp248-54 (Springer, 1998) ISBN 3-540-65414-3

[21] K. Hwang and Z. Xu *Scalable Parallel Computing: Technology, Architecture, Programming* (WBC/McGraw Hill, 1998) ISBN 0-07-031798-4

[22] D.J. Jablonowski, J.D. Bruner, B. Bliss and R.B. Haber *VASE: The Visualisation and Application Steering Environment* in Proc. Supercomputing'93 pp560-69 (IEEE Computer Society Press, 1993)

[23] B. Kagström, J. Dongarra, E. Elmroth and J. Waśniewski (eds.) *Applied Parallel Computing* Prod. 4th International Workshop PARA'98. Lecture Notes in Computer Science 1541 (Springer 1998) ISBN 3-540-65414-3

[24] P.G. Kropf *Overview of the WOS Project* in "High Performance Computing 1999" A. Tentner (ed.) (Society for Computer Simulation International, 1999) pp350-5 ISBN 1-56555-166-4

I. Banicescu and H. Unger *Running Scientific Computations in a Web Operating System Environment* in "High Performance Computing 1999" *ibid* pp356-62

S.A.Hopper, A.R.Mikler, P. Tarau, F.Chen and H. Unger *Mobile Agent-based File System for the WOS: an Overview* in "High Performance Computing 1999" *ibid* pp363-8

H. Unger *a New Security Mechanism for the use in large Distributed Systems* in "High Performance Computing 1999" *ibid* pp369-74

[25] M. Litzkow and M. Livny *Experiences with the Condor Distributed Batch System* in Proc. IEEE Workshop on Experimental Distributed Systems. (University of Wisconsin, Madison, 1990)

the Condor Distributed Processing System Dr. Dobbs' Journal (1995)

J. Pruyne and M. Livny, *Interfacing Condor and PVM to harness the cycles of workstation clusters* to appear in "Future Generations of Computer Systems" P. Sloot (ed.)

D.H.J. Epema, M. Livny, R. van Dantzig, X. Evers and J. Pruyne *A Worldwide Flock of Condors: Load Sharing among Workstation Clusters* to appear in Future Generations of Computer Systems, P. Sloot (ed.)

[26] M. Miller, C.D. Hansen and C.R. Johnson *Simulated Steering with SCIRun in a Distributed Environment* in "Applied Parallel Computing" Proc. 4th International Workshop PARA'98. B. Kagström, J. Dongarra, E. Elmroth and J. Waśniewski (eds.) LNCS 1541 pp366-76 (Springer, 1998) ISBN 3-540-65414-3

S.G. Parker, D.M. Weinstein and C.R. Johnson *The SCIRun Computation Steering Software System* in "Modern Software Tools in Scientific Computing" E. Arge, A.M. Bruaset and H.P. Langtangen (eds.) pp1-44 (Birkhäuser, 1997) ISBN 0-8176-3974-8 or ISBN 3-7643-3974-8

[27] R. Orfali and D. Harkey *Client/Server Programming with Java and CORBA* (Wiley, 1997) ISBN 0-471-16351-1

[28] D. Pritchard and J. Reeve (eds.) *Euro-Par'98: Parallel Processing* Lecture Notes in Computer Science 1470 (Springer 1998) ISBN 3-540-64952-2

[29] J. Pruyne and M. Livny *Parallel Processing on Dynamic Resources with CARMI* in "Job Scheduling Strategies for Parallel Processing" D.G. Feitelson and L. Rudolph (eds.) Lecture Notes in Computer Science 949 (Springer-Verlag, 1995)

[30] S.H. Russ, B. Flachs, J. Robinson and B. Heckel *Hector: Automated Task Allocation for MPI* in "10th International Parallel Processing Symposium" (IEEE Press, Honolulu, HI, Apr. 1996)

S.H. Russ, B. Meyers, M. gleeson, J. Robinson, L. Rajogopalan, C. Tan and B. Heckel *User Transparent Run-time Performance Optimisation* in "2nd Int. Workshop on Embedded HPC and Applications" at 11th IEEE Int. Parallel Processing Symposium (1997)

I. Banicescu and H. Unger *Running Scientific Computations in a Web Operating System Environment op cit*

S.H. Russ, J. Robinson, and M. Gleeson *Dynamic Communication Mechanism Switching in Hector* Technical Report MSSU-EIRS-ERC-97-8 (Mississippi State University, Dec. 1997)

S.H. Russ *Using Hector in an Architecture for Rapid Distributed Fault Tolerance* Technical Report MSSU-EIRS-ERC-97-17 (Mississippi State University, Dec. 1997)

J. Robinson, S.H. Russ, B. Flachs and B. Heckel *A Task Migration Implementation of the Message-Passing Interface* in Proc. Fifth High Performance Distributed Computing Conference (Syracuse, NY, Aug. 1996)

S.H. Russ, B.K. Flachs, J.A. Robinson and B. Heckel *Hector: Automated Task Allocation for MPI* Technical Report MSSU-EIRS-ERC-95-6 (Mississippi State University, Sept. 1995)

[31] S. Sekiguchi, M. Sato, H. Nakada, S. Matsuoka and U. Nagashima *Ninf: Network based Information Library for Globally High-performance Computing* in Proc. POOMA (Santa Fe, 1996)

[32] L.H. Turcotte *A Survey of Software Environments for Exploiting Networked Computing* Technical Report MSU-EIRS-ERC-93-2 (Mississippi State University, 1993)

[33] J. Vetter and K. Schwan *PROGRESS: A Toolkit for Interactive Program Steering* in Proc. 24th Ann. Conf. on Parallel Processing (1995)

[34] J. Vetter and K. Schwan *High-performance Computational Steering of Physical Simulations* in Proc. 11th Int. Parallel Processing Symposium (Geneva, Switzerland, April 1997)

High-level Architecture and run-time infrastructures (DoD Modelling and Simulation Office, DMS) via Web URL
`http://www.dmso.mil/hla`