Hybrid Computational Infrastructure Supporting eResearch

Geoffrey Fox, Indiana University May 24 2010

Introduction

Important developments -- the data deluge, Cloud computing, multicore architectures and growing importance of lightweight clients (tablets and smartphones) -- are changing the Cyberinfrastructure (eInfrastructure) supporting eResearch. Many of the detailed features of Grids seem unlikely to survive with clouds replacing many aspects of compute grids. On the other hand, supercomputers and clusters supporting traditional parallel computing will in near term at least remain unchanged. Those aspects of Grids supporting data will still be needed as data is intrinsically distributed as it is gathered by a multitude of sensors and instruments. However the utility computing aspects of Grids including high throughput computing seem likely to move to clouds which offer excellent support for loosely coupled jobs not requiring the low latency and localization of (MPI-based) parallel jobs. As described below new computing paradigms typified by MapReduce appear attractive as they offer ease of use with little performance loss. As noted below, the importance of clouds and the new software ideas is not just their technical merit but also that they are supported by commercial software and so it appears more likely that cloud-based eInfrastructure will be easier to sustain than Grids where lack of commercial interest implies reliance on academic software.

Cloud Computing

Cloud computing[1] is at the peak of the Gartner technology hype curve[2] but there are good reasons to believe that as it matures that it will not disappear into their trough of disillusionment but rather move into the plateau of productivity as have for example service oriented architectures. Clouds are driven by large commercial markets where IDC estimates that clouds will represent 14% of IT expenditure in 2012 and there is rapidly growing interest from government and industry. There are several reasons why clouds should be important for large scale scientific computing

1) Clouds are the largest scale computer centers constructed and so they have the capacity to be important to large scale science problems as well as those at small scale.
2) Clouds exploit the economies of this scale and so can be expected to be a cost effective approach to computing. Their architecture explicitly addresses the important fault tolerance issue.
3) Clouds are commercially supported and so one can expect reasonably robust software without the sustainability difficulties seen from the academic software systems critical to much current Cyberinfrastructure. As clouds evolve from "Infrastructure as a Service" to "Platform as a Service", there are a growing number of cloud computing tools. These include fault tolerant file systems and new storage models, distributed table data structures, a variety of databases, queues, notification, monitoring, web interfaces (portals), content delivery networks, scheduling and high level approaches to scheduling of multiple related jobs.
4) There are 3 major vendors of clouds (Amazon, Google, Microsoft) and many other infrastructure and software cloud technology vendors including Eucalyptus Systems that spun off UC Santa Barbara HPC research. This competition should ensure that clouds should develop in a healthy innovative fashion. Further attention is already being given to cloud standards [3]
5) There are many Cloud research, conferences and other activities with research cloud infrastructure efforts including Nimbus[4], OpenNebula[5], Sector/Sphere[6] and Eucalyptus[7].
6) There are a growing number of academic and science cloud systems supporting users through NSF Programs for Google/IBM and Microsoft Azure systems. In NSF OCI, FutureGrid[8] will offer a Cloud testbed and Magellan[9] is a major DoE experimental cloud system. The EU framework 7 project VENUS-C[10] is just starting.
7) Clouds offer "on-demand" and interactive computing that is more attractive than batch systems to many users.

Listening to some of the talks at the recent Cloud Futures workshop[11], one might imagine that all scientific computing could be performed on clouds. This is not true but rather the situation is somewhere in the middle with

some important classes of scientific computing being suitable for clouds but others not. The problems with using clouds are well documented and include

8) The centralized computing model for clouds runs counter to the concept of "bringing the computing to the data" and bringing the "data to a commercial cloud facility" may be slow and expensive.
9) There are many security, legal and privacy issues[12] that often mimic those Internet which are especially problematic in areas such health informatics and where proprietary information could be exposed.
10) The virtualized networking currently used in the virtual machines in today's commercial clouds and jitter from complex operating system functions increases synchronization/communication costs. This is especially serious in large scale parallel computing and leads to significant overheads in many MPI applications [13, 14]. Indeed the usual (and attractive) fault tolerance model for clouds runs counter to the tight synchronization needed in most MPI applications.

Some of these issues can be addressed with customized (private) clouds and enhanced bandwidth from TeraGrid to commercial cloud networks. For example, there could be growing interest in "HPC as a Service" as exemplified by Penguin Computing on Demand. However it seems likely that clouds will not supplant traditional approaches for very large scale parallel (MPI) jobs in the near future. It is natural to consider a hybrid model with jobs running on either classic HPC systems or clouds or in fact both as a given workflow (as in example below) could well have individual jobs suitable for different parts of this hybrid system. Commercial clouds support "massively parallel" applications but only those that are loosely coupled and so insensitive to higher synchronization costs. Let us focus on "massively parallel" or "many task" cloud applications as these most interestingly "compete" with possible Supercomputer implementations. In this case, the programming model MapReduce[15] describes problems suitable for clouds. This is offered on Amazon clouds and is expected soon on other commercial clouds while it can be implemented on any cluster using the open source Hadoop[16] software for Linux or the Microsoft Dryad system[17] for Windows clusters. One can compare MPI, MapReduce (with or without virtual machines) and different native cloud implementations and find comparable (with a range of 30%) performance on applications suitable for these paradigms [18]. MapReduce and its extensions offer the most user friendly environment.

One can describe the difference between MPI and MapReduce as follows. In MapReduce multiple map processes are formed -- typically by a domain(data) decomposition familiar from MPI -- these run asynchronously typically writing results to a file system that is consumed by a set of reduce tasks that merge parallel results in some fashion. This programming model implies straightforward and efficient fault tolerance by re-running failed map or reduce tasks. MPI addresses a more complicated problem architecture with iterative compute--communicate stages with synchronization at the communication phase. This synchronization means for example that all processes wait if one is delayed or failed. This inefficiency is not present in MapReduce where resources are released when individual map or reduce tasks complete. MPI of course supports general (built in and user defined) reductions so MPI could be used for applications of the MapReduce style. However the latter offers greater fault tolerance and user friendly higher level environment largely stemming from the coarse grain functional programming model implemented as side-effect free tasks. Over simplifying, MPI supports multiple Map-Reduce stages but MapReduce just one. Correspondingly clouds support application that have the loose coupling supported by MapReduce while classic HPC supports more tightly coupled applications. Research into extensions of MapReduce attempt to bridge these differences [19].

MapReduce covers many high throughput computing applications including "parameter searches". Many data analysis applications including information retrieval fit the MapReduce paradigm. In LHC or similar accelerator data, maps consists of Monte Carlo generation or analysis of events while reduction is construction of histograms by merging those from different maps. In the SAR data analysis of ice sheet observations, maps consist of independent Matlab invocations on different data samples. Life Sciences have many natural candidates for MapReduce including sequence assembly and the use of BLAST and similar programs. On the other hand partial differential equation solvers, particle dynamics and linear algebra require the full MPI model for high performance parallel implementation.

Grand Challenge Implications of MapReduce and Clouds

MapReduce and Clouds can be used for some of the applications that are most rapidly growing in importance. Their support seems essential if one is to support large scale data intensive applications. More generally a more careful analysis of clouds versus traditional environments is needed to quantify the simplistic analysis given above.

There is a clear algorithm challenge to design more loosely coupled algorithms that are compatible with the map followed by reduce model of MapReduce or more generally with the structure of clouds. This could lead to generalizations of MapReduce which are still compatible with the cloud virtualization and fault tolerance features.

There are many software challenges including MapReduce itself; its extensions (both in functionality and higher level abstractions); and improved workflow systems supporting MapReduce and the linking of clients, clouds and MPI engines. We have noted research challenges in security and there is also active work in the preparation, management and deployment of program images (appliances) to be loaded into virtual machines. The intrinsic conflict between virtualization and the issues around locality or affinity (between nodes in MPI or between computation and data) needs more research.

On the infrastructure side, we have already discussed the importance of high quality networking between MPI and cloud systems. Another critical area is file systems where clouds and MapReduce use new approaches that are not clearly compatible with traditional TeraGrid approaches. Support of novel data structures such as Big Table across clouds and MPI clusters is probably important. Obviously governments and the computational science community need to decide on the balance between use of commercial clouds as well as "private" science clouds mimicking Magellan and providing the large scale production facilities for codes prototyped on FutureGrid.

References

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia Above the Clouds: A Berkeley View of Cloud Computing http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf

[2] Press Release Gartner's 2009 Hype Cycle Special Report Evaluates Maturity of 1,650 Technologies http://www.gartner.com/it/page.jsp?id=1124212

[3] Cloud Computing Forum & Workshop NIST Information Technology Laboratory Washington DC May 20 2010 http://www.nist.gov/itl/cloud.cfm

[4] Nimbus Cloud Computing for Science http://www.nimbusproject.org/

[5] OpenNebula Open Source Toolkit for Cloud Computing http://www.opennebula.org/

[6] Sector and Sphere Data Intensive Cloud Computing Platform http://sector.sourceforge.net/doc.html

[7] Eucalyptus Open Source Cloud Software http://open.eucalyptus.com/

[8] FutureGrid Grid Testbed http://www.futuregrid.org

[9] Magellan Cloud for Science http://magellan.alcf.anl.gov/ , http://www.nersc.gov/nusers/systems/magellan/

[10] European Framework 7 project starting June 1 2010 VENUS-C Virtual multidisciplinary EnviroNments USing Cloud infrastructure.

[11] Recordings of Presentations Cloud Futures 2010 Redmond WA, April 8-9 2010 http://research.microsoft.com/en-us/events/cloudfutures2010/videos.aspx

[12] Lockheed Martin Cyber Security Alliance April 2010 Cloud Computing Whitepaper http://www.lockheedmartin.com/data/assets/isgs/documents/CloudComputingWhitePaper.pdf

[13] Edward Walker, Benchmarking Amazon EC2 for High Performance Scientific Computing, USENIX ;login, vol. 33(5), Oct 2008 http://www.usenix.org/publications/login/2008-10/openpdfs/walker.pdf

[14] Jaliya Ekanayake, Xiaohong Qiu, Thilina Gunarathne, Scott Beason, Geoffrey Fox High Performance Parallel Computing with Clouds and Cloud Technologies to appear as a book chapter to Cloud Computing and Software

Services: Theory and Techniques, CRC Press (Taylor and Francis), ISBN-10: 1439803153. http://grids.ucs.indiana.edu/ptliupages/publications/cloud_handbook_final-with-diagrams.pdf

[15] Dean, J. and S. Ghemawat. 2008. MapReduce: simplified data processing on large clusters. Commun. ACM 51(1): 107-113.

[16] Open source MapReduce Apache Hadoop, http://hadoop.apache.org/core/

[17] Jaliya Ekanayake, Thilina Gunarathne, Judy Qiu, Geoffrey Fox, Scott Beason, Jong Youl Choi, Yang Ruan, Seung-Hee Bae, Hui Li Applicability of DryadLINQ to Scientific Applications Technical Report January 30 2010 http://grids.ucs.indiana.edu/ptliupages/publications/DryadReport.pdf

[18] Thilina Gunarathne, Tak-Lon Wu, Judy Qiu, and Geoffrey Fox, Cloud Computing Paradigms for Pleasingly Parallel Biomedical Applications, Proceedings of Emerging Computational Methods for the Life Sciences Workshop of ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.

[19] Jaliya Ekanayake, Hui Li, Bingjing Zhang, Thilina Gunarathne, Seung-Hee Bae, Judy Qiu, Geoffrey Fox Twister: A Runtime for Iterative MapReduce, Proceedings of the First International Workshop on MapReduce and its Applications at ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.