Integrating Clouds and Cyberinfrastructure: Research Challenges

Manish Parashar, Rutgers University, parashar@rutgers.edu

Geoffrey Fox, Indiana University, gcf@indiana.edu

Other contributors…

Aggressive Cloud computing technology development has resulted in many different classes of Cloud services that provide attractive solutions for many different types of business applications. It is expected that Cloud services will join more traditional research cyber infrastructure components, such as high-performance computing system, clusters and Grids in supporting scientific exploration and discovery. It is clear from current research that there are real benefits in using Clouds and Cloud computing abstractions as part of a hybrid cyber infrastructure to support CDS&E, for example, to simplify the deployment of applications and the management of their execution, improve their efficiency, effectiveness and/or productivity, and provide more attractive cost/performance ratios. Furthermore, Clouds and Cloud computing abstractions can support new classes of algorithms and enable new applications formulations, which can potentially revolutionize CDS&E research and education. However, before CDS&E can fully realize the potential benefits of a hybrid cyber infrastructure that integrates Cloud services, several research issues remain as outlined in the previous sections and summarized below.

The discussion below highlights key research issues. The discussion is based on two reports. The first is by Gannon and Fox[[1]](#footnote-1), in which they reviewed and classified applications suitable for Clouds. The second is by Parashar et al.[[2]](#footnote-2), in which they explored how a hybrid HPC/Grid + Cloud cyber infrastructure can be effectively used to support real-world science and engineering applications, presented illustrative scenarios, and discussed limitations and research challenges.

**1. Algorithms and Application Formulations for Clouds**

A key attribute of Clouds is on-demand access to elastic resources, i.e., applications programmatically access more or less resources as they evolve, to meet changing needs. Such a capability can have a significant impact on how algorithms are developed and applications are formulated. For example, the execution of an application no longer has to constraint itself to a fixed set of resources that are available at runtime and can grow or shrink its resource set based on the demands of the science – the science can drive the scale and type of resource involved, based on for example, the levels of refinement required to resolve a solution feature, or the number of ensembles that need to be run to quantify the uncertainty in a solution, or the type of online analytics services that need to be dynamically composed into the application workflow. Understanding how CDS&E applications can effectively utilize Clouds and Cloud abstractions as part of a hybrid cyberinfrastructure, to enable new practices and levels of scientific insights remains a research challenge.

Research is also needed to explore the meaningful science, engineering and business application scenarios that can take advantage of such hybrid infrastructure. For example, a meaningful HPC plus Cloud use case may consist of simulations with online data analytics/visualization. In such a scenario, exposing the ability to modify goals/configurations based on data analytics feedback to the user will be critical to ensuring impact on the science. For example, in data-intensive computations the use of feature tracking might allow the scientist to adjust application parameters based on the analysis of meaningful features using a public Cloud, where the analysis can be performed in a timely manner due to shorter resource provisioning times compared to a high-end HPC system. We believe that such meaningful scenarios will exist in all areas of CDS&E.

There are important classes of applications that need special attention and have special research challenges. Examples are:

* Biomedical and bioinformatics applications where cloud architecture brings special challenges in the area of privacy (see later). Furthermore, Clouds have been attractive platforms for these applications as they are emerging big data areas and there is less history in using HPC platforms.
* Sensor webs are another emerging area where elastic nature of Clouds is well suited for the often bursty nature of sensor data.
* Big data applications based on new MapReduce or Iterative MapReduce environments are attractive on Clouds and result in broad research areas include addressing both programming and storage challenges. Latter include SQL and NOSQL models and the reconciliation of distributed data and centralized cloud computing

**2. Programming Systems and Abstractions**

One of the keys research challenges is developing appropriate programming abstractions and language extensions that can enable CDS&E applications to simply and effectively take advantage of the elastic access to resources and services during application formulation. Furthermore, it may be necessary to define constraints (for example, budgets, data privacy, performance, etc.) to regulate the elasticity, and the programming abstractions my provide support for expressing these constraints so that they can be enforced during execution. Similarly, such annotations can also define possible adaptations, which could then be used to increase performance, manageability and overall robustness of the application. For example, dynamically increase the assigned resources in order to increase the resolution of a simulation under certain convergence constraints, modify convergence goals to avoid failure or guarantee completion time. The Cloud service models can also lead to interesting services specialized to CDS&E that provide entire applications or applications kernels as a service (i.e., SaaS). Furthermore, and arguably more interestingly, it can also export specialized platforms for science as a services, which encapsulate elasticity and abstract of the underlying hybrid cyber infrastructure. In return, the scientists are only required to provide, core kernels, meaningful parameters, and basic configurations.

**3. Middleware stacks, management policies, and economic models**

Middleware services will need to support the new CDS&E applications formulations and services enabled. A key research aspect will be the autonomic management and optimization (multiple objectives including performance, energy, cost, reliability, etc.) of application execution through cross-layer application/infrastructure adaptations. It will be essential for the middleware services to be able to adapt to the application’s behavior as well as system configuration, which can change at run time, using the notion of elasticity at the application and workflow levels. Furthermore, appropriate services are necessary to be able to provision different types of resources on demand. For example, if we target HPC as a Cloud and HPC plus Cloud approaches on the NSF funded cyber-infrastructure such as XSEDE, Open Science Grid and FutureGrid along with commercial Clouds such as Amazon EC2 or Microsoft Azure, autonomic provisioning and scheduling techniques, including Cloud bursting will be necessary to support hybrid usage modes. Finally, monitoring, on-line data analytic for proactive application/resource management and adaptation techniques will be essential as the scale and complexity of both the applications and hybrid infrastructure grows.

There are many particular research areas related the two sections above. These include:

* Scheduling models optimized for MapReduce and for other Cloud usage modes such as scalable sensor webs (aka Sensor Grids or Clouds) where one has Clouds controlling and supporting a distributed Grid of sensors.
* Optimizing the run time features and performance for MapReduce and Iterative MapReduce. This includes new reduction primitives, polymorphic implementation on different systems with for example, exploitation of high performance networks as in classic MPI research.
* Support of federation of clouds and cloud bursting (typically the linkage of private and public Clouds) and on-demand cloud federation.
* New storage models such as data parallel HDFS and Hbase (Bigtable).
* NOSQL table structures such as Cassandra and commercial approaches such as Amazon SimpleDB and Azure Table.
* Economic models for an ecosystem with multiple cloud systems and CI.
* Research on Cloud software stacks. There is research at all levels of the software stack with two rather different emphasis areas. Research on systems that provide basic virtual machine provisioning, deployment and management. This includes Eucalyptus, Nimbus, OpenStack and OpenNebula with virtual networking as a distinct activity. At the other end are integration of capabilities to provide rich Platform-as-a-Service as offered by major commercial systems. Concepts such as appliances provide novel ways of delivering these capabilities.
* Clouds tend to achieve scalability by allowing faults. Research is needed on both, how to expose faults to users as well as services to build fault tolerant applications. Most research in HPC tends to be on forbidding faults; however Clouds highlight a different philosophy with resilient applications running on faulty systems.
* Green IT is naturally synergistic with Clouds and related research includes examining the impact of Cloud features on power use, including the cost of powering idle machines supporting elastic clouds as well as a application aware approaches to power management.

**4. End-to-end Challenges and Best-Practices in Building a Cloud Application**

Most attempts to directly port a conventional HPC application to a cloud platform fail. The challenge is to think differently and rewrite the application to support the new computational and programming models.  In the case of moving applications to current public/private clouds, the following practices can lead to success:

* Build the application as a service: Because you are deploying one or more full virtual machines and because clouds are designed to host web services, you want your application to support multiple users or, at least, a sequence of multiple executions.   If you are not using the application, scale down the number of servers and scale up with demand.  Attempting to deploy 100 VMs to run a program that executes for 10 minutes is a waste of resources because the deployment may take more than 10 minutes.  To minimize start up time one needs to have services running continuously ready to process the incoming demand.
* Build on existing cloud deployments: The cloud is ideal for large map reduce computations so use an existing map reduce deployment such as Hadoop or a similar service.
* Use PaaS if possible:  For platform-as-a-service clouds like Azure use the tools that are provided such as queues, web and worker roles and blob, table and SQL storage.  Note open source PaaS is weaker than commercial.
* Design for failure: Applications that are services that run forever will experience failures.   The cloud has mechanisms that automatically recover lost resources, but the application needs to be designed to be fault tolerant. In particular, environments like MapReduce (Hadoop, Daytona, Twister4Azure) will automatically recover many explicit failures and adopt scheduling strategies that recover performance "failures" from for example delayed tasks. One expects an increasing number of such Platform features to be offered by clouds and users will still need to program in a fashion that allows task failures but be rewarded by environments that transparently cope with these failures.
* Use as a Service where possible: Capabilities such as SQLaaS (database as a service or a database appliance) provide a friendlier approach than the traditional non-cloud approach exemplified by installing MySQL on the local disk. We anticipate many prepackaged aaS capabilities such as Workflow as a Service for eScience will be developed and simplify the development of sophisticated applications.
* Moving Data is a challenge: The general rule is that one should move computation to the data, but if the only computational resource available is the cloud, you are stuck if the data is not also there. Moving a petabyte from a laboratory to the cloud over the Internet will take time.   The idea situation is when you can gradually stream the data to the cloud over time as it is being created.  But if it exists in one place the best method of moving it is to physically ship the disk drives to the data center.   This service is available from some cloud providers.

**5. Security policies and mechanisms**

Clouds tend to emphasis the need for quality security mechanisms due to the sharing of storage and computing. One research area investigates hybrid architectures with algorithms broken into two; a low cost but non privacy preserving part running on an intrinsically secure private clouds, and a time consuming but privacy preserving part executing on a public cloud. Genomic data (human) and other health records are demanding here. The concept of differential privacy and health data anonymization is an active research topic. In addition to basic security for computing and storage there is research on privacy preserving search with the elegant but time consuming concept of Homomorphic Encryption, which allows encrypted data to be searched by encrypted queries. Some key research areas were highlighted at the recent NSF Workshop on Security for Cloud Computing[[3]](#footnote-3) include:

*Adversary models for cloud computing*:

* Identification of new security threats.
* Identification of possible sources of attacks – due to the different roles (user, providers, etc.) in cloud computing.
* Understanding how attacks in cloud computing are organized.
* Pricing and categorizing the security level.
* Relying on delegated technology and underlying cloud technologies.

*Delegation and authorization in cloud computing*:

* Delegation and authorization – such as attribute-based encryption for access control, secure comparison for complex policy enforcement, and encryption delegation for fine-grained temporal context.
* Mobile device access.
* Computation over encrypted data – when utilizing homomorphic encryption and homomorphic signatures.
* End-to-end cloud life cycle: restricted delegation, secure service composition, multiple credential types, and fine-grained access control.
* Restriction delegation (authorization based on capabilities not based on typical identification).

*End-to-end security in cloud computing*:

* Verification of work on clouds on behalf of clients.
* Kind of checks required at client and cloud sides.
* Trust between client and cloud.
* Security as a service within a cloud.
* Policy-based security applied to the end-to-end problem.
* Dealing with end-to-end privacy.
* Data ownership in the cloud.

*New problems in security for cloud computing*:

* Considerations of legal service level agreements (SLAs) and stronger privacy policies for content providers who collect large amount of personal data.
* Attestation of mechanisms in clouds – e.g., trusted launch of VM and VM migration.
* Attestation of actions and proof-attestation of provider security mechanisms.
* Execution of algorithms on encrypted data.
* Cloud forensics – e.g., secure and correlate temporal and spatial evidence, use of log-based event for reconstruction.
* Reactive stability challenges, cross-layer robustness, pervasive virtualization, secure migration of data, storage, dependencies between services, placement, and management vulnerabilities.

**6. Deployment/Transition to practice**

**Standards:** There are many important standard activities, from those specifying the basic virtual machine structure to higher-level standards defining the PaaS environment, for example, queue and table structures. Although there is some support for these standards – such as OCCI (from OGF) in OpenNebula and OpenStack – this area is still under development. NIST and IEEE are playing leadership roles.

**Procurement and Account Management Tooling:** One of the major obstacles today is the rudimentary state of account management and procurement options for obtaining and using cloud services. Substantial progress needs to be made in order enable procurement of services through capped purchase orders, or subcontracts; administration of sub-accounts and delegation of resources and authorities; reporting and alarming/limiting; and several others. This is an area in definite need of development activities.

**Documentation:** Documentation on how to use and the details of the system internals are both woefully inadequate. Development work is needed in partnership with the cloud providers, to improve the general documentation and particularly “HOW-TO” use cases for research usage.

**Technical Training:**The development and/or modification of codes adapted to the cloud environment require a set of skills currently in very short supply. These skills need to be taught, particularly to new practitioners in research, but also to mid-career practitioners in order to extract full benefit of this new technology.

1. G. Fox, D. Gannon, “Cloud Programming Paradigms for Technical Computing Applications,” Technical Report, <http://grids.ucs.indiana.edu/ptliupages/publications/Cloud%20Programming%20Paradigms.pdf>, 2012, [↑](#footnote-ref-1)
2. M. Parashar, M. AbdelBaky and I. Rodero, “Cloud Paradigms and Practices for CDS&E,” Technical Report, 2012, <http://cometcloud.org>, 2012. [↑](#footnote-ref-2)
3. NSF Workshop on Security for Cloud Computing http://illinois.edu/blog/view/695/66281?count=1&amp;ACTION=DIALOG [↑](#footnote-ref-3)