RE: NSF PI meeting on *The Science of Cloud Computing*

Name: Chandra Krintz
Affiliation: Computer Science Dept., Univ. of California, Santa Barbara
Email: ckrintz@cs.ucsb.edu January 17, 2011

Primary Topics: 1, 4
Secondary Topics: 3, 7, 9, 10

Summary of Current Research Related to Cloud Computing

Chandra Krintz and her students are the progenitors of AppScale, an open-source cloud platform that implements the APIs of Google App Engine (GAE) and other extant and new cloud services, including those for MapReduce, MPI, and data analytics. AppScale enables researchers and developers to implement, test, deploy, and scale their web applications and services using their own cluster resources (as opposed to executing (and storing sensitive code/data) on Google's resources, and potentially paying for this service). AppScale implements multiple language runtimes (Java, Python, Ruby, R, and X10) for application frontends, a wide range of open source database technologies (key-value or relational), which it configures, deploys, and scales automatically.

AppScale has an international user community of over 1000 members, is used as part of graduate and undergraduate classes at UCSB, and provides the foundation for Chandra and her group to investigate the next generation of cloud programming and runtime support. In particular, Chandra and her students are investigating (i) techniques for reducing the overhead of virtualization in cluster and cloud settings; (ii) monitoring and profiling cloud systems (support for different languages, components, and resources) for use in scaling, load balancing, feedback to developers, and cost estimation; (iii) cloud platform and service support for compute-intensive applications; (iv) and the technology necessary for interoperation and coordination between the cloud application and platform layers and heterogeneous cloud fabrics – the initial steps toward enabling hybrid cloud computing – which include Amazon EC2, Eucalyptus, Microsoft Azure, and Google App Engine.

Important Future Research Directions in Cloud Computing

Hybrid Cloud Support Cloud fabrics offer a wide range of similar software and services (storage, compute, data/task management, scaling) – yet, each has their own APIs and tools. This lack of portability leads to "lock-in" – the inability to easily move from one cloud system to another – and impedes adoption and wide spread use of cloud technologies. We propose to investigate new technologies that enable applications and services to execute without modification over multiple cloud platforms and infrastructures to facilitate, among other things, disaster recovery, cost competition, and transparent and coordinated use of private and public cloud systems.

Cloud-Aware Programming Language and Runtime Support To enable applications developers to customize, control, and debug cloud applications and thereby pursue a wider range of application domains for cloud execution, we propose to investigate new cloud-aware language extensions and runtime support. This includes mechanisms (i) for expressing the data requirements (persistence, sharing, replication, consistency), the task/data relationships, and the deployment/update/lifecycle expectations inherent in cloud applications to cloud fabrics, (ii) for supporting compute- and data-intensive applications (iii) for accessing cloud applications/services via mobile devices, and (iv) for enabling integrated application debugging.

Novel Uses of Cloud Systems: Software Stewardship We also plan to investigate using the cloud platform to address a critical impediment to collaborative science: *code maintenance* – the configuration, sharing/extending, correlated code/data provenance, release/version tracking, deploying, etc., of scientific software (research artifacts). The lack of code maintenance results in reinvention, inadequate comparison, inability to reproduce experiments, limited reuse, and loss of contributions due to the transient nature of academic software development (students gradate, post-docs move on). We propose software stewardship (a unifying workflow integration service) to ease and automate collaborative code maintenance through new cloud platform support, tools, APIs, and the use of virtual machine instances (snapshots of the environment, code, data, and software stack for a particular application) as scientific instruments.