Cloud Futures 2010: Advancing Research with Cloud Computing

April 8–9, 2010 | Redmond, WA

Microsoft® Research

Table of Contents

Interactive and Fault-tolerant Data Analytics in the Cloud

Magdalena Balazinska

University of Washington, USA
magda@cs.washington.edu

The ability to analyze massive-scale datasets has become a critical requirement for businesses and scientists today. Cloud-computing platforms can facilitate this task by offering powerful parallel data processing systems such as Dryad or MapReduce as a service with the option to scale these systems in an elastic fashion depending on a user's data analysis needs. MapReduce-type parallel processing systems, however, provide only batch processing, where a user must wait for an entire query to complete before seeing an answer. This strategy complicates exploratory data analytics, where a user wants to ask a variety of ad-hoc queries. One challenge associated with more interactive types of data analysis, as possible in Dryad, is fault-tolerance: if a set of interconnected operators execute in a pipelined fashion and one crashes, the whole set may need to be restarted. In this talk, we present our approach for addressing this challenge. We describe our newly developed techniques for efficient, fault-tolerant, yet pipelined parallel query processing.

Several strategies are in fact possible for making a pipelined query-plan fault-tolerant and the appropriate choice depends on several factors that include the failure model and the properties of the query plan. To address this second challenge, we developed, FTOpt, a cost-based fault-tolerance optimizer for parallel data processing systems. FTOpt selects the appropriate fault-tolerance strategy and resource allocation for each operator in a pipelined query plan so as to minimize the expected time to complete the query.

Experimental results demonstrate that different fault-tolerance strategies lead to the best performance in different settings and that our optimizer is able to correctly identify the winning strategy.

Biography:

Magdalena Balazinska *is an assistant Professor in the department of Computer Science and Engineering at the University of Washington.*

Magdalena's research interests are broadly in the fields of databases and distributed systems. Her current research focuses on scientific and sensor data management, cloud computing, and distributed stream processing. Magdalena holds a PhD from the Massachusetts Institute of Technology (2006). She is a Microsoft Research New Faculty Fellow (2007), received an NSF CAREER Award (NSF IIS-0845397), a Rogel Faculty Support Award (2006), and a Microsoft Research Graduate Fellowship (2003-2005).

Cloud Computing for Chemical Property Prediction

Paul Watson

Newcastle University, UK
paul.watson@newcastle.ac.uk

This talk will describe our collaboration with chemists to use Cloud Computing to accelerate the prediction of chemical properties based on their structure. The Chemists have unique software - the Discovery Bus - which automatically builds Quantitative Structure-Activity Relationship (QSAR) models. These can be used in the design of better, safer drugs (they have been working on new anti-cancer therapies), as well more environmentally benign products, at the same time reducing animal experimentation. New chemical activity data sets only become available irregularly, but when they do, the Discovery Bus requires large computational resources to generate new and improved models. This is because it explores a variety of model-building techniques before picking out the ones that produce the best results. For example, the Chemists have recently acquired some new data which will allow them to build more and better models, but unfortunately this would take over 5 years on their current infrastructure.

This is potentially an ideal Cloud application as large compute resources are required when new data is available, but this only occurs irregularly. Over the past 15 months, in the "Junior" project, we have designed and built a scalable, Azure cloud-based infrastructure in which the different model-building techniques are explored in parallel. As a result, the Discovery Bus has been accelerated to the extent that the processing of large data sets, which would previously have taken years, can now be completed in a few weeks.

This talk will describe the design of the Azure cloud-based infrastructure for accelerating QSAR modelling. It will cover how we addressed the practical challenges we faced, and summarize the performance results. Finally it will argue that the computational pattern we have exploited has general applicability for cloud computing.

Biography:

Paul Watson is Professor of Computer Science at Newcastle University, UK and Director of the North East Regional e-Science Centre. He also leads the RCUK "Social Inclusion through the Digital Economy" Hub.

He graduated in 1983 with a BSc in Computer Engineering from Manchester University, followed by a PhD in 1986. In the late 1980s, as a Lecturer at Manchester University, he was a designer of the Alvey Flagship and Esprit EDS parallel systems. In 1990 he moved to ICL to work as a system designer of a product based on these research prototypes. The result - the ICL Goldrush MegaServer - was released as a product in 1994. He moved to Newcastle University in 1995 where his main research interests are in data-intensive computing and e-Science.

In total, he has over seventy refereed publications, and three patents. He is a Chartered Engineer and Fellow of the British Computer Society.

Looking Inside the Virtualization Layer for Performance, Security, and Software Fault Tolerance

Sorav Bansal

IIT - Delhi, India
sbansal@cse.iitd.ernet.in

Cloud computing environments rely heavily on the virtualization layer to provide resource consolidation, dynamic load balancing, hardware fault tolerance, mobility, and homogeneity. While the current generation of virtualization engines are extremely good at what they do, we believe that a virtualization engine could do more. We are interested in looking at the virtualization layer as a platform for improving performance of the software stack running above it, enhancing software security, and providing better fault tolerance.

At IIT Delhi, we are developing a virtualization layer for x86 from grounds-up to better address these issues. While developing our virtualization layer, we made many design decisions to make it easier to add dynamic optimization rules, allow the addition of configurable low-overhead security checks, and have mechanisms to allow software to tolerate certain classes of bugs. Our virtualization engine runs directly on hardware (bare-metal hypervisor) and runs an unmodified guest operating system above it. We employ binary translation to virtualize the guest operating system, and do not rely on hardware support for virtualization available on recent processors (Intel VT, AMD-V).

We chose to implement a binary translator, instead of a trap-and-emulate style hypervisor to have greater control and visibility of the guest software, with minimal performance loss. We shall discuss our experiences in developing our binary translation based hypervisor, and present some early performance results. We describe, how we intend to use our virtualization layer for improving performance, security, and software fault tolerance of applications running on the cloud.

For Full PDF version of the abstract, see:
http://www.cse.iitd.ernet.in/~sbansal/pubs/cloudfutures2010_abstract.pdf

Biography:

Sorav Bansal is an Assistant Professor in the Computer Science Department at IIT Delhi. His research interests are virtualization, programming language support for concurrency, and testing and verification of concurrent programs. Sorav finished B.Tech. (CS) from IIT Delhi in 2001, and Ph.D from Stanford University in 2008.

Amazon Web Services (AWS)

Deepak Singh

Amazon

In an era where high throughput instruments and sensors are increasingly providing us faster access to new kinds of data, it is becoming very important to have timely access to resources which allow scientists to collaborate and share data while maintaining the ability to process vast quantities of data or run large scale simulations when required. Built on Amazon's vast global computing infrastructure, Amazon Web Services (AWS) provides scientists with a number of highly scalable, highly available infrastructure services that can be used to perform a variety of tasks. The ability to scale storage and analytics resources on-demand has made AWS a platform for a number of scientific challenges including high energy physics, next generation sequencing, and galaxy mapping. A number of scientists are also making a number of algorithms and applications available as Amazon Machine Images, or as applications that can be deployed to Amazon Elastic MapReduce. In this talk, we will discuss the suite of Amazon Web Services relevant to the scientific community, go over some example use cases, and the advantages that cloud computing offers for the scientific community. We will also discuss how we can leverage new paradigms and trends in distributed computing infrastructure and utility models that allow us to manage and analyze data at scale.

Biography:

Deepak Singh *is a business development manager at Amazon Web Services (AWS) where he is works with customers interested in carrying out large scale computing, scientific research, and data analytics on Amazon EC2, which provides resizable compute capacity in the Amazon cloud. Prior to AWS, Deepak was a strategist at Rosetta Biosoftware, a business unit at a subsidiary of Merck & Co. Deepak came to Rosetta Biosoftware from Accelrys where he was product manager for the life science modeling and simulation portfolio and subsequently Director of the Accelrys NanoBiology Initiative, an effort to investigate multiscale modeling technologies to model biological applications of nanosystems. He started his career as a scientific programmer at GeneFormatics, where was responsible for the maintenance and development of algorithms for protein structure and function prediction. Deepak has a Ph.D. in Physical Chemistry from Syracuse University.*

Towards Predictable Cloud Computing

Andreas Polze

Hasso-Plattner-Institute for Software Engineering at
University of Potsdam, Germany
andreas.polze@hpi.uni-potsdam.de

Service-Oriented Computing and Cloud Computing are built upon the abstraction of a service that appears to the outside user and programmer as the unit of granularity. Typically, services are assumed to be executed in a purely sequential fashion.

With the advent of multicore and manycore systems, the hunt for parallelism in applications has become the main theme for programmers and system architects. In order to achieve predictable behavior with multithreaded applications one has to consider inter-dependencies among the threads operating on behalf of a single user. With little monitoring and resource management infrastructure in place, this optimization goal is difficult to deal with in today's application servers and cloud computing models.

We have developed NTrace for monitoring and tracing within the Windows kernel. Based upon tools such as NTrace, we envision an architecture, where the local run-time system, the local fabric, can be used to identify potential performance and resource bottlenecks during testing of cloud applications under development. Under the assumption, that there exists a correlation between a cloud application's behavior on the local fabric and in the cloud, one may use automatically generated monitoring interfaces to oversee the application when deployed in the cloud.

Cloud Computing comes in three flavors: software-as-a-service, platform-as-a-service, and infrastructure-as-a-service. While the former two are well known through Salesforce.com, Google's AppEngine and Microsoft's Windows Azure cloud efforts, we are focusing on infrastructure-as-a service for providing experimentation environments at universities. Based on our experience in providing virtualized laboratories, we will outline a Cloud based infrastructure for conduction operating system experiments and programming tasks based on our Windows Research Kernel courseware.

Our talk will conclude with an overview on the work done at the operating systems and middleware group at Hasso Plattner Institute at University Potsdam, Germany.

Biography:

Prof. Dr. Andreas Polze *is the Operating Systems and Middleware Professor at the Hasso Plattner Institute for Software Engineering at University Potsdam, Germany. He is also the head of the Ph.D. school on "Service-Oriented Systems Engineering" at HPI. Andreas received a doctoral degree from Freie University Berlin, Germany, in 1994 and a habilitation degree from Humboldt University Berlin in 2001, both in Computer Science.*

At HPI, his current research focuses on architectures of operating systems, on component-based middleware, as well as on predictable distributed and cloud computing. Andreas Polze was visiting scientist with the Dynamic Systems Unit at Software Engineering Institute, at Carnegie Mellon University, Pittsburgh, USA, were he worked on real-time computing on standard middleware (CORBA) and with the Real-Time Systems Laboratory at University of Illinois, Urbana-Champaign. Andreas has acted as work component leader and member of scientific board in the 6th framework European Integration project "Adaptive Services Grid". Work in ASG has strong links to the Web Services community and industrial standardization efforts.

His current research interests include Predictable Service Computing, Adaptive System Configuration, and End-to-End Service Availability for standard middleware platforms. He is member of the GI and the IEEE. He currently is member of the program committees of ISORC (Intl. Symp. On Object-Oriented Real-Time Computing) and WORDS (Workshop on Real-Time Dependable Systems). Andreas Polze has (co-) authored more than 60 papers in scientific journals and conference proceedings. He has contributed to five books.

Together with Mark Russinovich and David Solomon, Andreas Polze is one of the co-authors of the Windows Curriculum Resource Kit (CRK), the top-download at the Microsoft faculty resource center. Current projects are centered around the Windows Research Kernel (WRK). Andreas Polze has been funded through the Rotor-I and Rotor -II projects. He received a Phoenix Direct Funding award in 2007 for his research on Phoenix for Real-time Robotics and Process Control.

The Cloud for University City Campus

Danilo Montesi

University of Bologna, Italy
danilo.montesi@gmail.com

This abstract concerns the development of a cloud enabling the creation of a University City Campus in Bologna. Bologna has specific features that make it a perfect location for this experimental setting.

First, the University of Bologna is one of the most attended universities in Italy, with about 85 thousand students enrolled. They are supported by a number of structures: 23 Faculties, 70 Departments and 1 Research Institute, 21 Research Centers, 11 Service Centers, 8 Special Structures, 67 Libraries and 16 Museum Structures.

Second, University buildings are spread all around the city, mainly in the city center with additional branches at strategic locations such as the S.Orsola University Hospital and the faculty of Engineering.

Third, a large and increasing part of the Bologna city center is already served by a free wireless network. This network, called Iperbole wireless (http://www.comune.bologna.it/wireless/), exploits many networks provided by other important institutions, like the University wireless network Almawifi which covers many University buildings, in addition to other dedicated hot spots.

Given this availability of people, structures and technological infrastructures, we envision the creation of a "Connected City Campus" based on services available in the cloud. In particular, at the beginning students will be able to use some virtualized laboratories which will enrich and extend existing physical structures. As an example, the Computing Labs of the Computer Science Department already provide access to Windows SQL Server and other services, that will be available in the cloud, concurrently accessible by a large number of students and without physical limitations.

The objective is to turn the entire city center of Bologna into a large "Campus in the Cloud", to provide an improved service to its students and to be used as a testing environment for new cloud technologies.

Biography:

Danilo Montesi obtained a Ph.D in Computer Science from the University of Pisa in 1993 and worked at the Politecnico di Milano. In October 1993 he became an ERCIM Fellow and moved to the University of Lisboa. He worked at the Rutherford Appleton Laboratory, under the same fellowship and at the Department of Computer Systems and Telematics, University of Trondheim. In October 1994 he was awarded a HCM fellowship to work at Imperial College. Under a Senior NATO fellowship he worked at Purdue University in 1996. He visited BT Labs with a Senior Fellowship in 1997. From 1996 to 2000 he was an assistant professor at the University of Milano and taught at the University of East Anglia. In 2000 he became an associate professor at the University of Bologna. In 2002 he became full professor at the University of Camerino. Since 2005 he is full professor of Database and Information Systems at the University of Bologna. His principal interests are in the area of database and information systems. He is now vice dean of the department and teaching director.

Cloud Computing Projects in Engineering

Harold Castro

Universidad de los Andes, Columbia
hcastro@uniandes.edu.co

José Tiberio Hernández

University of Los Andes, Columbia
jhernand@uniandes.edu.co

The development of e-Science is characterized by the need of big scale computational structures. These needs have been met with paradigms such as distributed computing grids; nevertheless the operation of this infrastructure has traditionally required that the final users are involved in installation, administration and maintenance processes, tasks that require non trivial IT knowledge and efforts.

In opposition, the cloud computing paradigm is disruptive easing the individualization and delivery of computing infrastructures (processing, storage and networks), software and applications, such as, high usability services hiding all the complexities associated to the basic infrastructure and technologies from the final user, while providing mechanisms to be accessed on demand via Internet. In that sense, the cloud computing paradigm represents an attractive approach to facilitate the creation of personalized development, testing and production environments that can be displayed, assigned and accessed on demand, thus providing services that can adapt dynamically and easily to the emerging needs in research development. On the other hand the versatility implicit in personalized execution environments, evidences the fact that cloud computing can be used to improve the development of academic activities supported by virtual laboratories, virtual learning environments and e-learning projects, increasing the degree of student participation, promoting interdisciplinary collaboration, development of practical skills and traditional learning methodology dynamics thus satisfying 21st century student expectations.

In the presentation the process to exploit the paradigm of grid computing at the Universidad de los Andes is presented, and how this process eventually led to the cloud infrastructures that are currently being developed. In the first place, and with the purpose of joining international projects, a Campus-grid initiative was launched that would offer core services needed to interconnect disperse resources in the campus and these to international grid initiatives already established or under development. Presently, Uniandes is part of project CMS of the LHC and is coordinating member for Colombia of the project EELA. In this process, different research groups at the University started to explore the benefits of having these type of infrastructures. Besides the High Energy physics group, groups like IMAGINE (Visual Computing) and COMIT (Information technology and communications) developed collaboration projects with researchers in biology, optimization, chemical engineering, civil engineering among others, proving the predominant role of grid computing in current research.

Harold Castro Barrera *graduated in Computer Science at Universidad de los Andes in 1989, he got a D.E.A (MSc) from the Institut National Polytehcnique de Grenoble (INPG) (http://www.inpg.fr) and since 1995 he holds a Ph.D. in Computer Science from INPG also. Since 2005 he is associate professor at the Computing and Systems Department (http://sistemas.uniandes.edu.co) at Universidad de los Andes. Besides his teaching and advising activities at graduate and undergraduate level, he is the director of the COMIT (Communications and Information Technology) research group which main research focus are distributed systems. He personally leads grid activities not only at the university level but at a national scope also: he coordinates the national JRU for grid activities. His Interest areas are: Distributed systems, grid and cloud computing and Mobile computing.*

José Tiberio Hernández Peñaloza *has a PhD in Informatics at l'Ecole Nationale Supérieure de Techniques Avancées, an MSc in Computers and Systems Engineering, as well as being Computers and Systems Engineer of the Universidad de los Andes. He has worked as Dean and Vice-Dean of the Engineering School of the Universidad de los Andes and currently works as Associate Professor in the same institution. His Interest areas are: Visual Computing, HPC in collaboratives environments, Innovation in Engineering.*

Proposing a Curriculum/Course for Cloud Computing from a Software Engineering Perspective

Werner Kurschl

Upper Austria University of Applied Sciences, Austria
werner.kurschl@fh-hagenberg.at

Since cloud computing made its way into highly scalable business applications it's gaining more and more attention from researchers, software engineers, and IT media. Thoroughly reading through journals, books, and IT magazines reveals some misunderstandings, because not everyone agrees on what it is, what the building blocks are, how a typical architecture should look like, and how a cloud computing application is built. To overcome this unsatisfying situation especially for the software engineering domain, we propose a cloud computing curriculum. This curriculum will define key terms, discuss key software technologies and cloud software architecture issues. It will also show implementations of comparable sample applications based on Microsoft Windows Azure and Google App Engine as well as architectural design issues for large and complex cloud applications. Design guidelines as well as open issues will conclude the curriculum.

Biography:

Werner Kurschl *is a professor for Software Engineering at the Upper Austria University of Applied Sciences since 2001, where he is teaching bachelor and master courses: i) software programming skills (C/C++, ii) C#, Java), iii) web architectures and frameworks, iv) as well as mobile and ubiquitous systems (e.g. wireless sensor networks, natural speech processing, etc).*

He is also the head of the research group SMART at the Upper Austria University of Applied Sciences. This group works with industry and academia on topics like distributed systems, cloud computing, wireless sensor networks, software architectures, and model-driven software development.

He holds a PhD in Computer Sciences and he is a member of the Microsoft Innovation Award jury (2009, 2010) in Austria.

Operating System Research Extended into the Cloud

Alexander Schmidt

Hasso-Plattner-Institute
alexander.schmidt@hpi.uni-potsdam.de

Cloud Computing comes in three flavors: software-as-a-service, platform-as-a-service, and infrastructure-as-a-service. While the former two are well known through salesforce.com, and Google's and Microsoft's recent cloud efforts, the latter service, as provided by amazon.com, is especially interesting with respect to providing computer science experimentation environments at universities. Our experience has shown that students tend to solve their assignments on their own laptops rather than on faculty provided machine. Based on this observation, we will outline a Cloud-based infrastructure, the SkyLab, for conducting operating system experiments and programming assignments, based on our Windows Research Kernel (WRK) courseware, online in the cloud. Finally, the talk will be concluded with an overview on the work done at the Operating Systems and Middleware group at Hasso Plattner Institute at University Potsdam in Germany.

Biography:

Dipl.-Inf. Alexander Schmidt *studied computer science at the Chemnitz University of Technology where he graduated and received his diploma. In 2006 Alexander Schmidt joined the Operating Systems and Middleware group at Hasso-Plattner-Institut (HPI) as a Ph.D. student. His main research focus is in the area of application monitoring and especially in the operating system context.*

At HPI, Alexander is involved in teaching operating systems courses as well as the Windows Research Kernel project. He contributes to the Windows Monitoring Kernel, an efficient event-logging infrastructure for monitoring arbitrary applications based on Windows systems as well as the NTrace tool for dynamically instrumenting applications at function boundaries. As part of his thesis, he created the KStruct OS kernel inspection framework, which focuses on consistently accessing shared data structures while the OS is running.

Alexander has been a Summer intern with Microsoft in Redmond in 2008 and 2009.

Consumer Decision Support for Resource Procurement in IaaS Markets

Kurt Vanmechelen

University of Antwerp, Belgium
kurt.vanmechelen@ua.ac.be

The uptake of the cloud computing paradigm and the ensuing development of the Infrastructure as a Service (IaaS) market, has given rise to exciting new possibilities for on-demand IT resource provisioning and brings with it a potential to transform the manner in which consumers approach IT infrastructure needs. The change from a fixed-capacity infrastructure model based on capital expenditures to a virtually unlimited capacity model based on operational expenditures, leads to new challenges related to cost-optimization and user decision support. As cloud IaaS markets mature, consumers are confronted with an increasing number of providers, products, and pricing schemes. Quickly, the complexity of this new market environment is increasing to a level that makes the problem of cost-optimized resource allocation under application-level QoS constraints intractable without decision support systems. The presence of resources internal to the consumer organization in hybrid clouds further complicates these matters, especially in terms of cost accounting.

In this talk, we present the HICCAM (Hybrid Cloud Construction And Management) research project that aims to tackle these issues. We report on the design of a management framework for optimizing resource acquisition in IaaS markets and show to what extent our initial optimization problems can be solved by linear programming and heuristics tailored to the problem domain. In addition, we draw from our experience in the field of Grid Economics to comment on the IaaS market structure and trading mechanisms, and present an outlook for further developments in this area. Finally, we highlight the integration of cloud-related courses in the curriculum of the Master in Computer Science at the University of Antwerp.

Biography:

Dr. Kurt Vanmechelen *is a postdoctoral researcher and teaching assistant within the Computational Modeling and Programming (CoMP) group of the Department of Mathematics and Computer Science at the University of Antwerp, Belgium. His main research interests are centered on Grid and Cloud Computing systems in general, and resource management and allocation within such systems in particular. He obtained a PhD in Computer Science from the University of Antwerp in 2009, in the field of market-based resource management for Grid systems.*

Using Cloud Computing to Support Virtual Congress

Juan G. Lalinde-Pulido

Linea de Investigación y Desarrollo en Informática Educativa, Columbia
jlalinde@eafit.edu.co

Every year, the Colombian Ministry of Education holds several meetings to capacitate teachers. In February 2009, the Ministry offered the first Colombian Virtual Congress, centered about collaborative projects, and supported by EAFIT University with its technological platform. The purpose of this congress was to test if this kind of events could be hold in a completely virtual environment. Been a proof of concept, it was promoted word of mouth, so the initial expectation was to have about 300 attendees, whom must register in order to have access to the platform. The congress was conceived as a virtual community with support for video conferences, forums, virtual classrooms, sharing experiences and wiki. Three days before starting the congress there were 4000 users registered, so we were forced to make some changes in the supporting platform in order to assure its stability under that demand. At the end, there were more than 8000 attendees and the congress was a complete success, with an average of 2000 visits a day.

The main problems that we had to solve in order to scale the platform were related with the fact that users are active in the congress and they can publish information. Some restrictions aroused due to technical design of the platform used, like storing published documents on a file system instead of a database, but some limitation aroused from the dynamic of the interactions. This restriction was so strong that dynamics like chat had to be removed from the video lectures.

In this talk we will present the technological structure of the congress, the solutions we used on the run for scaling, how the architecture is been modified in order to be deployed on a cloud and the economical aspects that made cloud computing the right platform for this type of events.

Biography:

Juan G. Lalinde-Pulido *has a BS in Computer Science from EAFIT University, a BS in Mathematics from Universidad Nacional de Colombia and a PhD in Telecommunications from Universidad Politécnica de Valencia, Spain, working with Distributed Real Time Systems. He joined EAFIT University in 1995. From 1998 to 2005 he was the head of the Computer Science Department and the director of the Computer Science Program. In 2003 he was one of the coordinators for the design of the ECAES test for Computer Science. His experience with High Availability Systems dates back to 1993 when he developed network management systems for PTT companies. Since 2005 he is an advisor to Colciencias, the government department responsible for science and technology in Colombia. He is GSEC Gold Certified from the SANS Institute since 2002. In 2009 he joined the Linea de Investigación y Desarrollo en Informática Educativa to coordinate the engineering lab.*

Building Application for the Cloud on Google App Engine

Christian Schalk, Google

This session will provide a general overview of Google App Engine technology and how it serves as a Platform as a Service. We will then provide more detailed technical information on how to build Python and Java apps (with an emphasis on Java) and deploy them to the cloud.

Biography:

Chris is a Developer Advocate at Google who engages both the App Engine and Enterprise OpenSocial communities. He has also worked on Google Friend Connect, Google AJAX APIs, Maps, Gears and Google Web Toolkit. Prior to Google, Chris was a Principal Product Manager at Oracle in the development tools group as well as co-author of "JavaServer Faces: The Complete Reference". In his spare time, he plays trumpet in local Bay Area symphonies.

Early Progress on SEJITS in Manycore and Cloud Environments

Armando Fox

University of California, Berkeley, USA
fox@cs.berkeley.edu

Cloud computing bears the exciting promise of democratizing massively parallel programming with large datasets, but the necessary programmer tools are still primitive compared to languages such as Python and Ruby, which are expressive enough to support high-level abstractions matched to the application domains of productivity programmers. We propose to facilitate cloud programming by allowing programmers to write in such a language, but at runtime we intercept entry to selected blocks of code embodying patterns of computation that a human expert has specified how to map to the computing platform. This specification consists of source code templates and instructions for instantiating them based on runtime arguments.

We call this idea Selective, Embedded, Just-in-Time Specialization, or SEJITS. The approach is selective because unlike optimizing compilers, we don't need to be able to JIT-specialize the whole language, or automatically figure out the "best" way to map a computation onto available hardware resources; if no human-provided specializer is available, we can always fall back to executing in the host language.

It is embedded because the necessary introspection and metaprogramming facilities are built into modern scripting languages, eliminating the need to modify the language interpreter and thereby making it easier for specializer authors to make their work available to application writers.

A key insight of SEJITS is that it decentralizes the work of producing specializers for different computation patterns on different computing platforms, and allows specialization decisions to be made on a function-by-function level within a single application rather than committing to a single framework (Hive, map/reduce, etc.) which then constrains how the entire application must be written.

Given that SEJITS arose in the context of programming manycore using highly-productive languages, we believe there is promise that the same Python source code could run without modification on a variety of architectures (e.g. a cloud of GPGPUs), selectively identifying patterns in the application for which a specializer for the underlying platform is available.
We will discuss early progress on SEJITS in manycore and cloud environments, as well as a cloud computing substrate called Nexus that provides the machinery by which a single cloud can support multiple frameworks simultaneously such as MPI, map/reduce, etc.

Biography:

Armando Fox *(fox@cs.berkeley.edu) is an Adjunct Associate Professor at UC Berkeley and a co-founder of the Berkeley RAD Lab. Prior to that he was an Assistant Professor of Computer Science at Stanford, and received his PhD, MS and BS degrees at Berkeley, Illinois at MIT respectively. His current research interests include applied statistical machine learning and cloud computing; he is a co-author of the recently released position paper "Above the Clouds: A Berkeley View of Cloud Computing" and has frequently spoken on this topic. He has published several papers in collaboration with top machine learning researchers on the application of machine learning to diagnosing, characterizing and identifying operational problems in datacenter-scale and cloud computing installations. His 2003 collaboration with David Patterson on Recovery-Oriented Computing earned him the distinction of being included in the "Scientific American 50" top researchers. In previous lives he helped design the Intel Pentium Pro microprocessor and founded a company to commercialize his UC Berkeley dissertation research on mobile computing.*

Consistency Options for Replicated Storage in the Cloud

Ken Birman

Cornell University, USA
ken@cs.cornell.edu

Existing cloud computing platforms are all over the spectrum on the issue of consistency for data replicated by cloud-hosted services. The most common approach accepts Brewer's CAP conjecture as a kind of theorem, abandoning consistency guarantees under the belief that these are incompatible with high performance and availability, especially in systems that need to be accessible during partitioning failures. Indeed, many cloud owners accord CAP the status of a kind of folk theorem, asserting that strong forms of consistency lead to poor scalability, disruptions such as performance oscillations, and can even cause cloud communication hardware to fail, for example by abusing mechanisms such as IPMC in a greedy effort to maximize update speeds for replicated data. But when we abandon consistency in favor of ad-hoc properties such as eventual (but not guaranteed) convergence, we also end up with systems that are poorly suited to handling sensitive applications, such as ones that need secure replication of keys within a group, or that would play critical roles such as in an air traffic control system. On the other hand, most researchers believe in CAP and would conjecture that strong consistency models just can't scale adequately.

In recent work, I've been developing Isis2, a platform that aims to create an easily used data replication option for programmers working with C# on .NET and other typed, managed languages. The skill level expected from users is similar to that needed when using C# with a GUI builder platform such as Visual Studio.

As the name suggests, Isis2 will run as a modern group communication library, in which replication occurs at the granularity of individual objects and can be extremely dynamic: the object is given a name, like a file name (indeed, the file might be where the object persists its state when inactive, if it has anything that needs to be saved), and applications bind to objects rather casually, at runtime, just as one can write programs that open large numbers of files. While a group is active, the replicated data lives in the current members of the current view; "clients" of a group, and members, can multicast to it, and any replica can read its local state. Locking is available for applications that need to avoid concurrency problems. And the core protocol that interests me the most maps update operations directly to IP multicast: the cost of an asynchronous replicated update is that of a single UDP send, with other overheads hidden in the background. Obviously, one can't beat this performance target, but does this level of performance preclude consistency guarantees?

I'll argue that scalability and performance are feasible even with strong consistency, namely a new model that combines elements of virtual synchrony with elements of state machine replication (Paxos) [1]. Isis2 poses many other challenges: no matter what consistency model is used, the system will need to scale in the numbers of groups, numbers of members, and number of applications using the system within the cloud as a whole. I won't have time to focus on those topics, but will summarize some of my reasons for believing that these challenges can be overcome.

Biography:

Ken Birman *is the N. Rama Rao Professor of Computer Science at Cornell University. His work focuses on data center consistency, reliability and scalability, and he has a long history of developing algorithms for fault-tolerance, security, and reliable multicast in such settings. The virtual synchrony model he developed was widely adopted as a consistency policy for replicated data, and the Isis Toolkit he created became popular as a tool for building data centers; it played a foundational role in the New York and Swiss Stock Exchanges, the French Air Traffic Control System and the US Navy AEGIS Warship, among other high profile applications. The author of several books and more than 200 journal and conference papers, Dr. Birman was Editor in Chief of ACM Transactions on Computer Systems from 1993-1998, is a Fellow of the ACM, and was the recipient of the 2009 IEEE Kanai Award for Distributed Systems and the 2009 IEEE TPDS Outstanding Achievement Award.*

Elasticity through Modularity

Jan Rellermeyer

ETH Zurich, Switzerland
rellermeyer@inf.ethz.ch

Cloud computing has the potential to significantly affect and improve the way how data is processed and research conducted. However, in order to leverage the scalability of the cloud fabric, applications have to become equally elastic and flexible. This is so far an open challenge in software engineering. In practice, cloud computing faces many of the challenges and difficulties of distributed and parallel software. While the service interface hides the actual application from the user, the application developer still needs to come to terms with distributed software required to run on dynamic clusters and operate under a wide range of configurations.

In this talk, we outline our vision of a model and a runtime platform for the development, deployment, and management of applications in the cloud. Our basic idea is to turn the notion of software modules into a first class entity used for management and distribution so that the modules can be autonomously adapted, monitored, and re-deployed by an underlying intelligent runtime system in order to increase the reliability, scalability, and availability of the services provided. The runtime implements the interface of a single system, hiding the complexity of the underlying fabric and using the space between the virtual deployment (as seen by the application) and the physical distributed deployment (as seen by the cloud fabric) for runtime optimization. We present a working prototype implementing the proposed concept and leveraging the OSGi standard for dynamic modules in Java (.net is future work). Through a symbolic code analysis at load-time, the system identifies the shared state of modules and uses this information to dynamically provision, migrate, and replicate the modules—adapting and rewriting the code when needed to achieve elasticity. We envision such a runtime to significantly ease the process of writing scalable and composable cloud services for collaborative research.

Biography:

Jan S. Rellermeyer *received his MSc CS in Distributed Systems from ETH Zürich, Switzerland, in 2006. Currently, Jan is finishing his PhD in the Systems Group at ETH, which is lead by Prof. Gustavo Alonso, Prof. Donald Kossmann, Prof. Timothy Roscoe, and Prof. Nesime Tatbul. His research interests are cloud computing, modular software architecture, distributed systems, and language runtime systems.*

Jan is the invited researcher of the OSGi Alliance and author and contributor to several successful open source projects. He is also a regular speaker at both developer and scientific conferences.

Jan's work on the "Virtual OSGi Framework" is supported by the Microsoft Innovation Cluster for Embedded Software (ICES).

Programming a Service Cloud

Rosa M. Badia

Barcelona Supercomputing Center–CSIC
rosa.m.badia@bsc.es

Currently, programmers face the challenge of adapting their applications to the new computing paradigms (i.e., distributed computing environments, clouds, Software as a Service (SaaS), underlying hardware platforms, etc). With the objective of simplifying the programmers challenges, COMP Superscalar (COMPSs) aims to provide a straightforward programming model and an execution framework to help on abstracting application from the actual execution environment. Besides other features, COMPSs is able to decompose the applications in tasks, organizes the tasks in a task graph that contains the data dependences between tasks, parallelizes the execution of the applications at task level, performs automatic data transfers when required, etc,without having to add any special call to the original Java application.

COMPSs has been developed by BSC during the last two years. Currently there is a stable version that runs in distributed platforms and clusters. This version is characterized by Java as programming language and ProActive GCM and JavaGAT as underlying middleware, and Javassist as mechanism for interception. In the context of the proposal, BSC plans to evolve the current system (programming model + runtime environment) to a Service Oriented Architecture programming framework (Service Supercalar, ServiceSs). Applications are enabled to run in a cloud infrastructure or in a federation of clouds.

The core elements of the applications will be computational tasks executed by workers in the cloud or web services offered by one or several service providers (service provider side). Each application (applications side) will be composed by these core elements and offered as a service (Software as a Service).

The ServiceSs framework will provide:

- A programming model for SOA, based on standard sequential programming languages that are familiar to the developers, that requires a minimum effort for efficiently executing applications in the cloud.
- A runtime able to orchestrate the execution of the core elements (tasks and web services). At execution time, the ServicesSs runtime detects the dependencies between the different core elements of an application and uses this information to generate a workflow. The workflow is dynamically scheduled,mapping each core element to a cloud resource or to a service instance offered either by the internal or external service providers, taking into account terms of business, trust or performance parameters to meet the SLA terms.
- A set of tools for deployment of the core elements and composite applications.

The ServiceSs characteristics mentioned above such as the infrastructure unaware programming model and the interception mechanisms make this framework suitable for moving current applications to cloud computing paradigm because reduce the changes required by the original application to the minimum.

Biography:

Rosa M. Badia *has a PhD on Computer Science (1994) from the Technical University of Catalonia (UPC). She is a Scientific Researcher from the Consejo Superior de Investigaciones Científicas (CSIC) and manager of the Grid Computing and Cluster research group at the Barcelona Supercomputing Center (BSC). She was involved in teaching and research activities at the UPC from 1989 to 2008, where she was an Associated Professor since year 1997. From 1999 to 2005 she was involved in research and development activities at the European Center of Parallelism of Barcelona (CEPBA). Her current research interest are: performance prediction and modelling of MPI programs and programming models for complex platforms (from multicore to the Grid/Cloud). She has published more than 90 papers in international conferences and journals in the topics of her research. She has participated in several European projects and currently she is currently participating in projects NUBA (at Spanish level), TERAFLUX and OGF-Europe and it is a member of HiPEAC2 NoE.*

Enabling Scalable Genomics Research Across Desktop and the Cloud

Yogesh Simmhan
Microsoft Research, Redmond
yoges@microsoft.com

Scientific research is faced with the challenge of marshaling and analyzing large volumes of data from a variety of instruments and simulations, the resource capacity for which is beyond the reach of most scientists. Cloud computing has the potential to advance data intensive sciences by providing shared, pay as you go access to large scale storage and compute nodes co-located at global datacenters that offer economies of scale. While cloud computing democratizes on demand resource access for the broader eScience community, the promise of scalability should not be at the cost of increased application complexity and rewrite overhead. Also, the ability to work with familiar desktop tools is as important to scientists as the need to off load resource heavy tasks to the cloud.

Computational biology has traditionally been at the forefront of adopting technological advances, be it with open data sharing, web service access to tools or in silico experiments, and is likewise poised to make use of cloud computing opportunities as can be seen through recent literature.

In this talk, we describe two applications from genomics: genome wide association study (GWAS) and human genome phasing that run on the Windows Azure cloud using platform tools we have developed. The GWAS application, which typically runs on an HPC cluster, uses our generic worker pattern to migrate and run on Azure at scale with comparable performance. The phasing application is composed as a workflow in the Trident Workbench that seamlessly orchestrates across activities present on both desktop and the cloud. In particular, data movement between the divide is handled transparently and efficiently.

These two applications realize the potential of the cloud and demonstrate how tools such as these can bridge the gap between the cloud as it stands and a true platform for eScience.

Biography:

Yogesh Simmhan *is a postdoc researcher working in the eScience group at Microsoft Research. His research is in enabling large-scale, data- and compute- intensive applications to effectively leverage current and emerging distributed platforms, such as clouds, clusters and grids, with particular focus on workflows and provenance, scientific data management, and distributed scheduling. Yogesh has a Ph.D. from Indiana University advised by Profs. Beth Plale and Dennis Gannon.*

Implementing a DICOM Image Archival Solution in Microsoft Windows Azure Cloud – An Information Technology Senior Project

Chia-Chi Teng

Brigham Young University, USA
ccteng@byu.edu

Growing long-term cost of managing an onsite medical imaging archive has been a subject which the health care industry struggles with. Based on the current trend, it is estimated that over 1 billion diagnostic imaging procedures will be performed in the United States during year 2014, which will generate about 100 Petabytes of data. The high volume of medical images is leading to scalability and maintenance issues with health providers' onsite picture archiving and communication system (PACS) and network.

Cloud computing promises lower cost, high scalability, availability and disaster recoverability which can be a natural solution to the rising cost of long-term medical image archive. While researchers and policy makers are still actively studying the security, privacy, and liability issues involving sensitive medical information in the cloud, various technology vendors such as IBM and Amazon have started to provide solutions for early adopters. A group of senior Brigham Young University Information Technology (IT) students are inspired to implement a digital imaging and communications in medicine (DICOM) archive service in the cloud based on Microsoft Windows Azure platform and tools as their senior capstone project.

The system includes 1) a DICOM server running as a worker role which handles standard DICOM store/query/retrieve requests; 2) a DICOM image indexer, also as a worker role, that parses the metadata (tags) and store the information a SQL Azure database; 3) a web role UI implemented in Silverlight that allows users to search and view archived DICOM images based on any and combination of DICOM tags. The DICOM server and indexer are implemented in C# based on the open source project DICOM# (http://dicom-cs.sourceforge.net/). Valuable lessons were learned by the students as they researched and developed on the Azure platform, also by the faculty as they integrate this new technology in to their IT curriculum.

Biography:

Chia-Chi Teng *is an associate professor of the Information Technology program at Brigham Young University since January 2008. He received his BS in Electrical Engineering from National Tsing Hua University in Taiwan in 1985 and his PhD in Electrical Engineering from University of Washington in Seattle in 2007. Prior to his academic career, he was a Software Design Engineer for nearly 17 years at Microsoft in Redmond, from 1989 to 2005. He participated in the design and development in products such as Windows 95, Internet Explorer, Microsoft Network (MSN), Media Center, and Windows Vista, etc. He was also a co-inventor of 12 patents during his tenure at Microsoft. His current research interests include biomedical image analysis and management, biomedical and health informatics, and online health education systems. In his research and teaching, he emphatically promotes Microsoft technologies among the students and encourages them to learn through various projects.*

Rapid Diagnosis of Acute Heart Disease by Cloud-based High Performance Computing for Computer Vision

Oleksii Morozov

University Hospital of Basel, Switzerland
AMorozov@uhbs.ch

To advance in the treatment of acute heart disease, novel approaches to objective and quantitative analysis of cardiac imaging data are needed. Leading edge analysis of 4D cardiac datasets requires solving of very large mathematical problems: We have extensive experience in autonomous quantitative heart analysis, where a typical equation to solve may have up to dozens of millions unknowns - in emergency situations, results should be available very quickly.

Currently, the clinical applicability of these potentially groundbreaking new diagnostic technologies is severely limited by the requirement of high performance computing infrastructure at multiple points of care, e.g. the large number of hospital emergency rooms in a country. However, for a single point of care, this high computing performance may only be needed for a little time per day. Therefore, a strategy where computation is not performed locally but in a cloud is highly desirable from a quality of care standpoint, as well as a healthcare economic view.

In this presentation, we will show how large scale collaborative research and application in this area can be enabled by a cloud based service. As an example we will describe handling and analyzing large datasets representing full motion data in normal and abnormal hearts in acute heart disease. We will also show how the related task of gathering and analyzing globally representative cardiac reference datasets could profit from a distributed system like a cloud. Furthermore, we will estimate cost for performing such tasks using local resources compared to a cloud offering, e.g. Microsoft Azure, for a specific problem in acute cardiovascular care.

Biography:

Oleksii Morozov *was born in Veseloye, Zaporozhye rgn., Ukraine, on March 23, 1981. He received the M.Sc. degree in radio-electronics from National Aerospace University "KhAI", Kharkov, Ukraine in 2004. He is currently pursuing the Ph.D. degree in the Physics in Medicine group at the University Hospital of Basel, Switzerland, where he is working on biomedical multidimensional signal processing.*

Microsoft Azure for Research

Roger Barga and Jared Jackson

Microsoft Research

Cloud computing offers a potential mechanism to increase the efficiency of current research, ensure continuity of critical data and enable new kinds of research not now feasible. In this model, researchers focus on the higher levels of the software stack – applications and innovation, not low-level infrastructure. The cloud service providers deliver economies of scale and capabilities driven by a large market base and energy efficient infrastructure. In this tutorial we will cover the basics of the Windows Azure cloud computing platform and its associated application programming model. Throughout our tutorial we will use examples from actual research applications and services built on Windows Azure to illustrate how cloud computing can be used an intellectual amplifier for research.

Biography:

Roger Barga *is currently an architect in the Cloud Computing Futures (CCF) group in Microsoft Research (MSR), where he leads a technical engagements team that works with researchers interested in carrying out large scale computing, scientific research, and data analytics on the Windows Azure cloud. Prior to joining CCF, Roger led the Advanced Research Tools and Services (ARTS) team in MSR which built innovative services and tools for data intensive research. Roger joined Microsoft in 1997 as a Researcher in the Database Group of Microsoft Research, where he participated in both systems research and product development efforts in database, workflow and stream processing systems. Contact him at barga@microsoft.com.*

Jared Jackson *is a Lead Research Software Development Engineer in the Extreme Computing Group of Microsoft Research. He is currently helping develop technologies that bring cloud computing to researchers of divergent backgrounds. Jared has been with Microsoft Research for over four years and has focused on projects involving computer science education, scientific workflows, and bioinformatics. Jared has a M.S. in Computer Science from Stanford University.*

Leveraging Domain-Specific Software Architectures for Classifying Cloud Service Abstractions

T.S. Mohan
1Ecom Research Labs, Infosys Technologies
Bangalore, India
subramanian_mohan@infosys.com

Cloud computing service abstractions have been broadly classified at three levels—as relevant to IT managers (as Infrastructure as a Service, IAAS), developers (Platform as a Service, PAAS), or end users (Software as a Service, SAAS). However, scientists and business enterprises have found it difficult to choose the right cloud services for leveraging the cloud's disruptive economies of scale: elastic computational power or storage. Reference architectures for the cloud have been particularly lacking, as have specifications of the key interactions between various cloud services and deployment configurations. This is likely due to the organic emergence of the cloud. Ten years ago, the notions behind cloud conflicted with best practices of exotic and expensive hardware and software, as opposed to low-cost, commodity alternatives. But, as Google, Yahoo, and other web innovators have shown, those days are over; clouds can scale economically using commodity hardware and specialized cloud software. The time is ripe for an explicit software engineering focus when considering the cloud.

While clouds are new, several best practices and specific lessons from the world of Grid Computing have a great potential for adoption. The notion of Domain Specific Software Architectures (DSSA) when applied to grids enables the articulation of service component interactions both at configuration time and at runtime. Using a cloud DSSA, we uncover several interesting parallels between grids and clouds. Specifically we look into key examples from Amazon web services (representative of IAAS), Microsoft Azure and Google App Engine (representative of PAAS), and Gmail or Hotmail email service (representative of SAAS) to illustrate the power of applying DSSA. Given the plethora of cloud computing service offerings, we argue that DSSA could represent a plausible model for comparative analysis, helping to illustrate tradeoffs in various cloud service offerings.

Biography:

T.S. Mohan *is a Principal Researcher in Ecom Research Labs, E&R Unit of Infosys Technologies in Bangalore. His areas of research interests include High Performance Computing – Grid and Cloud Computing as well as large-scale software architectures and systems.*

Achieving Energy Efficient Computing for Future Large Scale Applications

Tajana Simunic Rosing
University of California, San Diego, USA
tajana@ucsd.edu

In a future where data will not only come from classical computing systems as it does today, but also from millions of sensors and mobile devices, the need for energy-efficient large-scale data computation will explode. An excellent example of such a system is CitiSense project underway at UCSD. Individuals using CitiSense infrastructure will get feedback on how the environment and their actions affect their long term health. Public health agencies, clinicians, educators, insurance companies will be able to analyze interaction between health, individual choice and environment for the first time at large scale. The amount of data and computation required for applications such as CitiSense will push data center capabilities to limits and will lead to unreasonable energy costs.

The goal of SEE Lab is to design systems that are dramatically better at delivering the results needed with minimum energy spent in the process. We have developed optimal power management strategies for stationary workloads that have been implemented both in HW and SW. Run-time adaptation has been done via an online learning algorithm that selects among a set of policies. We generalize the algorithm to include thermal management since we found that minimizing the power consumption does not necessarily reduce the overall energy costs. To reduce the performance costs typically associated with state of the art thermal management techniques, we developed a new set of proactive management policies. The experimental results using real datacenter workloads on an actual multicore system show that our proactive technique is able to dramatically reduce the adverse effects of temperature by over 60%. Most recently we have shown that symbiotic scheduling of workloads in virtualized environments can lead to average 15% energy savings with 20% performance benefit in high utilization scenarios.

Biography:

Tajana Šimunic Rosing *is currently an Assistant Professor in Computer Science Department at UCSD. Her research interests are energy efficient computing, embedded and wireless systems. Prior to this she was a full time researcher at HP Labs while working part-time at Stanford University. At Stanford she has been involved with leading research of a number of graduate students and has taught graduate level classes. She finished her PhD in 2001 at Stanford University, concurrently with finishing her Masters in Engineering Management. Her PhD topic was Dynamic Management of Power Consumption. Prior to pursuing the PhD, she worked as a Senior Design Engineer at Altera Corporation. She obtained the MS in EE from University of Arizona where her thesis topic was high-speed interconnect and driver-receiver circuit design. She has served at a number of Technical Paper Committees, and is currently an Associate Editor of IEEE Transactions on Mobile Computing.*

Early Observations on the Performance of Windows Azure

Marty Humphrey
University of Virginia, USA
humphrey@cs.virginia.edu

While Cloud computing APIs, mechanisms and abstractions are rapidly maturing, perhaps the most significant open issue is performance -- the concern is not necessarily whether I can write my application to run in the cloud, but rather how the application will perform if I do so. For example, we are currently working with the National Cancer Institute to design services for Windows Azure for the cancer Biomedical Informatics Grid (caBIG), and it is extremely challenging to determine the lowest-cost, scalable software architecture that is able to predictably meet the performance requirements (from both the client perspective and the service perspective). Few, if any, Cloud providers offer quantitative performance guarantees and instead typically define computing capabilities in abstract terms (e.g., Amazon's "EC2 Compute Units"). Regardless of the potential advantages, Cloud infrastructures will ultimately fail if applications cannot predictably meet behavioral requirements.

In this talk, we present the results of performance experiments we have conducted on Windows Azure since October 2009. To make a comprehensive assessment, we measure the performance of the major APIs, both during the CTP phase and the commercial release. First, to assess the ability to dynamically scale, we measure how long it takes to acquire and release computing instances (i.e., virtual machine startup/shutdown costs). Second, we measure the performance of all three Azure storage services: blob, queue and table. Third, we evaluate the performance of the database service, SQL Azure Database, specifically the influence of client location and scalability in terms of the number of clients. Fourth, as a comparison to the loosely coupled communication model provided by Azure storage services, we study the inter-role TCP/IP communication performance in Azure. We conclude the talk with our overall assessment and provide recommendations for potential users of Windows Azure based on these early observations.

Biography:

***Marty Humphrey** is an Associate Professor in the Department of Computer Science at the University of Virginia. He received a B.S. and M.S. degree in Electrical Engineering from Clarkson University in 1986 and 1989, respectively. He received his Ph.D. degree in computer science from the University of Massachusetts in 1996. From 1996-1998, he was an Assistant Professor of Computer Science and Engineering at the University of Colorado at Denver. From 1998-2002, he was a Research Assistant Professor at UVa. From 2002-2008, he was an Assistant Professor at UVa. His research projects have been supported in part by the National Science Foundation, Department of Energy, National Cancer Institute, IBM, and Microsoft.*

Management and Contextualization of Scientific Virtual Appliances

Germán Moltó

Universidad Politécnica de Valencia, Spain
gmolto@dsic.upv.es

The advent of Cloud computing technologies introduces new challenges for the execution of scientific applications, which typically require specific hardware and software configurations. Currently, Virtual Machine Managers (VMMs) such as Eucalyptus or OpenNebula pave the way to manage VMs in elastic Cloud infrastructures, providing moderate support for VM contextualization (e.g. IP allocation, hot-plugging disks at boot time, etc.). However, the process of turning a VM into a Virtual Appliance (VA) that encapsulates the entire hardware and software configuration for an application to run successfully is still fairly manual. Scientific applications typically rely on numerical libraries, databases, web services, system packages, etc. whose installation and configuration can sometimes be automatically performed. Therefore, we envision semi-automatic procedures to contextualize VMs for scientific applications, where users and/or developers would provide a declarative description of their applications and its dependences. Then, contextualization software would resolve the dependences and provide the appropriate environment to install each requisite. However, installing complex software might not be automated so easily. This would lead to a set of Pre-Contextualized VMs (PCVMs) where the user (or the VM provider) would have installed this kind of software. These PCVMs could be stored in VM providers (such as Amazon S3) and a VM Repository Service would store metadata information about each VM. This would allow PCVMs sharing among different scientists, thus enhancing collaboration. These PCVMs would certainly be finally configured at boot time to produce a self-contained VA that allows the execution of the application.

We are currently working towards this vision and have developed Proof-of-Concept tools that automatically deploy scientific applications that rely on different build systems (ant, make), different software packages (Tomcat, Globus Toolkit WS Core), and different configuration files (XML, INI files). These tools, although in their early stages, ease the deployment of applications onto base virtual machines.

Biography:

Germán Moltó *is associate professor in the Department of Information Systems and Computation at the Universidad Politécnica de Valencia (UPV), and researcher in the Grid and High Performance Computing group (GRyCAP). He received the B.Sc. and Ph.D. degrees in computer science from that university in 2002 and 2007. His research interests include Grid and Cloud Computing, Service-Oriented Architectures and its usage in different scientific fields, such as biomedical applications and protein design. He is the Principal Investigator (PI) of a project that aims at combining Grid and Cloud computing for the execution of scientific applications. He has coauthored more than 50 papers in international and national conferences as well as over 10 papers in high-impact journals. He holds a title of Specialist in Universitary Pedagogy. In 2009 he received an Excellence in Teaching award from the Computer Science School. He currently teaches data structures and algorithms to second-year undergraduate students*

SKYINFO – Innovative Technology Convergence Cloud Application

Matjaz B. Juric

University of Maribor, Slovenia
matjaz.juric@uni-mb.si

Today, every individual is overwhelmed with large amount of data, generated on every single moment of his life. Most of this data carries no significant meaning and it only makes it harder for individual to find information, he is looking for. Extracting relevant and valuable information presents a though challenge. We have developed SKYINFO to answer these challenges. The goal of SKYINFO is to provide an intelligent and pervasive information service, which offers timely accurate, localized and personalized information. On one hand users report about events in their proximity, and on the other they receive information about messages from other users, based on their location and expressed interest. Possible use cases include reporting of accidents and other traffic information, localized advertising, warning about approaching weather conditions, reporting lost and found objects, etc.

Key contribution of SKYINFO is simple access to relevant information in real time, using smart mobile devices. User is notified only about events that took place near his location, that suit areas of his interest and are valid inside a specified time frame. Main SKYINFO innovation is linking relevant information and events with location and user interest, which leads to quickest access to desired information. Selection of information that is presented to each user is based on current user's location, his expressed interests and time frame, within which information is valid. SKYINFO uses an innovative approach for notifying about occurred events. User can send a photo with included geo-tag and SKYINFO automatically sorts it into corresponding category (traffic, event, etc.) and extracts location. Third innovation is an algorithm for calculating message and user trust, which helps us identify and eliminate false messages.

SKYINFO uses several Microsoft technologies, including Azure, Silverlight, .NET framework. It also provides interoperability with other platforms, such as Java, using web services. This project has been developed in cooperation with the major mobile services provider in our country.

The project has been developed by an expert group at the Cloud Computing Centre at the University of Maribor Science Park in close cooperation with the Microsoft Innovation Centre. It has been established in 2009 as the first Cloud Computing Centre in this part of Europe.

Biography:

Matjaz B. Juric, Ph.D.*, is a professor at the University of Maribor, and the head of Cloud Computing Centre. He has more than 15 years of work experience. He has authored/coauthored Business Process Driven SOA using BPMN and BPEL, Business Process Execution Language for Web Services (English and French editions), BPEL Cookbook: Best Practices for SOA-based integration and composite applications development (award for best SOA book in 2007 by SOA World Journal), SOA Approach to Integration, Professional J2EE EAI, Professional EJB, J2EE Design Patterns Applied, and .NET Serialization Handbook. He has published chapters in More Java Gems (Cambridge University Press) and in Technology Supporting Business Solutions (Nova Science Publishers). He has also published in journals and magazines, such as SOA World Journal, Web Services Journal, eai Journal, theserverside, ACM journals, etc. Matjaz has been involved in several large-scale projects. Matjaz is a member of the BPEL Advisory Board and author of courses and consultant.*

Gregor Srdic is a researcher and a Ph.D. student in the field of Cloud Computing. He has specialized for the development of web services and Software+Services applications.

Martin Potocnik is a researcher and a Ph.D. student in the field of Services and Cloud Computing. He has specialized for the development of Azure, Silverlight and ASP.NET user interfaces.

Matej Hertis is a researcher and a Ph.D. student in the field of Services and Cloud Computing. He has specialized for the semantic analysis of content.

Domain Specific Cloud Components for General Availability of Cloud Computing to Research

Marco Parenzan, Science

Università degli Studi di Trieste, Italy
marco.parenzan@libero.it

This paper deals with availability of cloud computing to computational research labs. We will focus to the concept of availability. This concept may have two different interpretations, namely:

- "Available" as an accessible resource, always, from everywhere
- "Available" as the ability to consume a service (as a client or as the publisher)

This paper will focus on the second interpretation: a cloud service is "available" if it is easy for anyone in the academic community (and not) to consume the cloud. Indeed, cloud allows sharing "knowledge" in form of components or data to be "executed" in the cloud. The challenge here is to make possible for researchers, not necessarily expert in programming and computer science, to make available her/his knowledge in form of components and data tables.

The solution we propose is based on Domain Specific Languages (DSL), by which a researcher will express the components in her/his specific language, that will be user-friendly since it is directly related to the particular research field. In this framework, cloud components will be expressed in terms of a generic mathematical model rather than a software component. This vision is quite common in computing thanks to the availability of many tools that simplify the development of DSL such as dynamic languages like IronRuby or revolutionary data access with SQL Server Modeling.

The objective of this work is to present a model of a general "Domain Specific Cloud Component" (DSCC) that can be expressed, published and consumed by the research community using tools that allow an easy and direct implementation for the mathematical algorithms developed by the scientists. The general concept will be applied to specific examples by developing frameworks customized to share a specific "DSCC".

Examples will be taken in the area of multiscale molecular modeling for the design of nanostructured polymer systems (nanotechnology) and the estimation of the environmental impact of a production process (sustainability).

Biography:

Marco Parenzan *is a contract professor at University of Trieste about Web Services programming. He is also a researcher in MOSE Laboratory in the field of programmability for the tools available in the lab. As a freelance programmer and consultant, his current research is in the field of Dynamic Languages and the emerging SQL Server Modeling. See http://blog.codeisvalue.com/.*

Towards an Open Service Framework for Cloud-based Knowledge Discovery

Domenico Talia
Università della Calabria, Italy
talia@deis.unical.it

Cloud computing was originally designed for dealing with problems involving large amounts of data and/or compute-intensive applications. Today, however, Clouds enlarged their horizon as they are going to run both scientific and business applications supporting scientists, professionals and end users. To face those new challenges, Cloud environments must support adaptive knowledge discovery and data mining applications by offering resources, services, and decentralized data analysis methods. In particular, according to the service oriented architecture (SOA) model, data mining tasks and knowledge discovery processes can be delivered as services in Cloud computing infrastructures. Through a service-based approach we can define integrated services for supporting distributed scientific data analysis tasks in Clouds. Those services can address all the tasks that must be considered in knowledge discovery processes from data selection and transport, to data analysis, knowledge model representation and visualization. We are working along this direction by providing service-oriented architectures and services for distributed knowledge discovery. This has been done by designing services corresponding to:

- Single steps that compose a KDD process such as pre-processing, filtering, and visualization;
- Single data mining tasks such as classification, clustering, outlier detection, and rule discovery;
- Distributed data mining patterns such as collective learning, parallel classification and meta-learning models;
- Knowledge discovery processes/applications including all or some of the previous tasks expressed through a multi-step scientific workflows.

This collection of data mining services composes an Open Service Framework for Cloud-based Knowledge Discovery. This framework allows developers to design distributed KDD processes as a composition of services that are available over a single Cloud or on Interclouds. Here we describe a strategy based on the use of Cloud services for the design of open distributed knowledge discovery services and outline how Cloud frameworks, can be developed as a collection of services and how they can be used to develop distributed data analysis tasks and knowledge discovery processes using the SOA model.

Biography:

Domenico Talia *is a full professor of computer engineering at the University of Calabria and the director of ICAR-CNR. His research interests include Cloud and Grid computing, distributed knowledge discovery, parallel data mining, and peer-to-peer systems. Talia published five books and more than 250 papers in international journals such as Communications of the ACM, Computer, IEEE TKDE, IEEE TSE, ACM Computing Surveys, IEEE Internet Computing and conference proceedings. He is a member of the editorial boards of the IEEE TKDE, Future Generation Computer Systems journal, the International Journal on Web and Grid Services, and the Web Intelligence and Agent Systems International journal. He is serving as a program committee member of several conferences and is a member of the ACM and the IEEE Computer Society.*

Microsoft Codename "Dallas"

Christian Liensberger, Microsoft

cliens@microsoft.com

Microsoft Codename "Dallas" in an information marketplace for consuming and selling data and web service. Dallas offers data from various premium content providers, such as InfoUSA, WeatherCentral, Associated Press, and from various government agencies and non-profit organizations, such as data.GOV and the UN. In this session we are going to explore opportunities that Dallas provides to institutes around the world. Discover how you could use the platform to augment existing on-premise data or to build/extend applications on top of your and Dallas' data.

Biography:

Christian Liensberger *is a Program Manger at Microsoft. He has been involved in the Dallas team since its early stages and helped shipping the first CTPs of the platform. Before joining Microsoft Christian studied computer graphics and image processing at the University of Technology Vienna.*

Managing the Cloud Infrastructure

Antonio Cisternino

Dipartimento di Informatica, University of Pisa
cisterni@di.unipi.it

Cloud computing is taking shape, shifting the focus from pure computing typical of the grid toward data distribution. The focus shift imposes several constraints in the management of computational resources assigned to a cloud. If a node of a grid is hopefully used during computations, state and applications for the cloud may have more dynamic behavior with respect to computational resources. The ability of manage efficiently the farm running cloud services is important both for granting high availability and an power efficiency. Cooling and power consumption have become a major issue and current infrastructure for managing clusters is still focused on distributing computations rather than packing computations and managing nodes.

At University of Pisa we developed a framework using F# called Octopus (http://octopus.codeplex.com) still in its infancy. This framework lays the foundations for managing a virtualized cluster of computers running only virtual machines. Through a Web page a user can asks for a virtual machine with a specific OS and use it for arbitrary computations. The framework attempts to automatically move virtual machines across nodes in order to pack them on a many-cores cloud. Users can access their virtual nodes from the Internet not knowing where the real hardware is situated. Automatic management of a possibly heterogeneous set of computing nodes is challenging even though systems features most of the required functionalities, nevertheless monitoring and cost models are required to decide when a particular action should be taken.

Biography:

Antonio Cisternino *is assistant professor in the Computer Science Department of the University of Pisa. His primary research is on meta-programming and domain-specific languages on virtual-machine-based execution environments. He's been active in the .NET community since 2001, and developed annotated C#, an extension of C#, and Robotics4.NET, a framework for programming robots with .NET. He is currently working on VSLab, a Visual Studio add-in based on F# to extend the environment into an interactive environment for scientific computing. He co-authored the "Expert F#" book on the F# programming language together with the author of the language Don Syme. Antonio holds a Ph.D. in computer science from the University of Pisa.*

QoS-aware Clouds

Fabio Panzieri
University of Bologna, Italy
panzieri@cs.unibo.it

The success of next-generation cloud computing infrastructures will depend on how effectively these infrastructures will be able to instantiate and dynamically maintain computing platforms, constructed out of cloud resources and services, that meet arbitrarily varying resource and service requirements of cloud consumer applications. Typically, these applications will be characterized by Quality of Service (QoS) requirements, such as timeliness, scalability, high availability, trust, security, specified in so-called Service Level Agreements (SLAs); i.e., legally binding contracts that state the QoS guarantees an execution environment, such as a cloud based computing platform, has to provide its hosted applications with. Thus, in this project we propose to design and develop a middleware architecture that enables SLA-driven dynamic configuration, management and optimization of cloud resources and services, in order to respond effectively to the QoS requirements of the cloud customer applications.

For the purposes of our discussion, we term "QoS-aware cloud" a cloud computing environment augmented with our middleware architecture. A QoS-aware cloud will be able to change dynamically, either "on-demand" or based on application run time needs, the amount of resources made available to the applications it hosts. Optimal resource utilization will be achieved by providing (and maintaining at run time) each hosted application with the number of resources which is sufficient to guarantee that the application SLA will not be violated. The architecture we propose, that will be described in detail in our presentation, extends an earlier middleware architecture we developed as part of the EU-funded project TAPAS (IST Project No. IST-2001-34069, http://tapas.sourceforge.net, 2006) in order to support clustering of QoS-aware application servers (G. Lodi, F. Panzieri, D. Rossi and E. Turrini, "SLA-Driven Clustering of QoS-Aware Application Servers", IEEE Trans. Soft. Eng. 33(3): 186-197, 2007). We believe that the design principles we adopted in the TAPAS project are equally appropriate in the context of cloud computing in order to construct what we have termed QoS-aware clouds.

Biography:

Fabio Panzieri *is a full professor of Computer Science at the University of Bologna (Italy), currently serving as Head of the Department of Computer Science of this University. He obtained the "Laurea" degree in Computer Science from the University of Pisa (Italy, February 1978), and the Ph.D. degree in Computer Science from the University of Newcastle upon Tyne (U.K., June 1985). His research activity is centered on issues of design of fault tolerant distributed systems, distributed real-time systems, and middleware and communication support for responsive distributed applications. He has (i) authored and co-authored over 60 research papers that have appeared in international journals and proceedings of international conferences, (ii) served as member of the programme committee of a number of international conferences and workshops, and (iii) participated to both national and international, EU-funded projects,including the recent the project TAPAS (IST-2001-34069) in which he was principal investigator for Bologna University.*

A New Class of Services for Cloud Computing: Real-Time Services Over Massive and Continuous Data Flows

Ricardo Jimenez-Peris

Universidad Politecnica de Madrid, Spain
ricardo.jimenezperis@gmail.com

Most cloud computing efforts have focused on the store-and-process paradigm. However, there are a number of potential services that do not fit well with this paradigm because they require to process massive data flows in an online manner.
In this position paper, we present a new area for cloud computing that will enable a new breed of real-time services based on the processing of massive continuous data flows.

We also identify some open challenges to cloudify data streaming and describe how we are dealing with them in StreamClouds, a large scale data streaming system we are developing.

Biography:

Prof. Ricardo Jimenez-Peris *is director of the Distributed System Lab at Technical University of Madrid. The emphasis of his research during the last decade has been on addressing the scalability and availability of transactional databases, multi-tier service platforms, and data streaming systems. Lately, his research is addressing cloud computing and he is coordinating two European projects on cloud platforms on platforms as a service for transactional multi-tier service platforms and large scale data streaming.*

Cognitive Publish/Subscribe for Heterogeneous Clouds

Sarunas Girdzijauskas

Swedish Institute of Computer Science, Sweden
sarunas@sics.se

A fundamental building block for any cloud network is an efficient and reliable communications infrastructure, e.g., a publish/subscribe service. However, the organization of a cloud can be very complex and may consist of arbitrary numbers and sizes of smaller microclouds and services running on many different physical networks. The capacity and costs of data transfer within and beyond an individual microcloud network could differ by several orders of magnitude. Besides that, the whole cloud system is usually powered by ever-changing and often erratic user behavior, which makes it difficult, if not impossible, to predict any upcoming usage patterns and trends. Failing to account for such heterogeneous nature of the clouds could render the existing publish/subscribe methods inefficient or even inoperable. Unfortunately, most of the current publish/subscribe solutions do not account for most of the above mentioned conditions and are incapable of providing efficient pub/sub dissemination structures over highly heterogeneous and complex topologies of the underlying decentralized network with unforeseeable utilization patterns.

In this talk we will present our ongoing work on a large scale and completely decentralized gossip-based publish/subscribe system. The system is able to continuously self-organize and adapt itself to ever-changing properties of the cloud networks. The awareness of the underlying connectivity and user behavior is achieved through sensing and monitoring the gossip layer. Using such techniques, we can dynamically organize nodes with similar subscriptions into efficient pub/sub dissemination structures which preserve locality in the subscription space as well as exhibit minimal bandwidth consumption and transmission costs. Our initial results suggest that we are able to significantly reduce the message propagation and data replication costs by taking advantage of the subscription correlation (as found in many real-world group-based applications). Since the system is based on an unstructured overlay network, it is very robust to failures and can operate in very volatile environments. The gossiping mechanisms allow the system to be always up-to date and exploit ever-changing user subscription patterns with minimal maintenance costs.

Biography:

Sarunas Girdzijauskas *is a post-doctoral researcher in the Computer Systems Laboratory at the Swedish Institute of Computer Science (SICS). He got his M.Sc. (2002) in Informatics from Kaunas University of Technology, Lithuania and PhD (2009) from Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland. His research focuses on large-scale distributed systems, peer-to-peer topologies, Small-World phenomenon in peer-to-peer overlays, scalable publish/subscribe systems and content distribution.*

Panel on Cloud Applications—New Experiences and Expectations.

Paul Watson, Newcastle University
paul.watson@ncl.ac.uk

Marty Humphrey, University of Virginia
humphrey@cs.virginia.edu

Bertrand Meyer, ETH Zurich
Bertrand.Meyer@inf.ethz.ch

Rob Gillen, Oak Ridge National Laboratory
gillenre@ornl.gov

Savas Parastatidis, Microsoft Corporation
savasp@microsoft.com

We're rushing towards pervasive, always-connected lives and augmented reality at breakneck speed, and the technology landscape is changing so fast we seldom take stock of how different our lives are – and how dependent we are on technology. Cloud computing is in its early days but holds the promise of powering the next step in the progression of human to computer interaction through the introduction of new applications and new interaction models. In this panel we'll take a provocative look at some of the ways cloud application are changing our lives today and what the future of client plus cloud applications might hold.

Data and the Cloud.

A Call for Improved Formats and Consistency in Data Services Supporting Scientific Research

Rob Gillen

Oak Ridge National Laboratory
gillenre@ornl.gov

A variety of cloud computing platforms are emerging and there exist increasing calls for research to be performed utilizing these platforms. While the computational paradigms are improving and the platforms are, in fact, becoming increasingly viable for such research, significant issues remain surrounding the usage and movement of data (specifically large data) within these environments. If cloud computing as a paradigm is to deliver on its promise of democratizing large-scale compute by serving as a natural extension of the scientific workstation, significant effort must be applied towards the issues of data locality, presentation, and interaction. This talk assumes a layman's perspective as work is reviewed illustrating challenges encountered while attempting to move and interact with large-scale data (subsets of the CMIP3 archive) within the cloud as well as from workstations outside of the cloud. Also discussed are issues surrounding the schism that exists between data formats that would be considered "Internet friendly" and those used in a variety of scientific research and how these formats exacerbate the problems surrounding data movement.

Biography:

Rob Gillen *is a member of the Computer Science Research Group at Oak Ridge National Laboratory currently focusing on the scientific applications of Cloud Computing and the democratization of technical computing (both platforms and tools). Prior to joining ORNL in 2007 Rob spent 8 years working in both the federal and commercial arenas delivering software solutions to various customer problems with a specific focus on large-scale datacenters (telcos, service providers, etc.).*

Experiences using Windows Azure to Process MODIS Satellite Data

Jie Li

University of Virginia, USA
jl3yh@Virginia.EDU

At the October 2009 Microsoft eScience workshop, we presented our first design and implementation for a Moderate Resolution Imaging Spectroradiometer (MODIS) satellite data processing framework in Windows Azure. We presented a novel approach to "reproject" the MODIS input data into timeframe and resolution aligned and uniform geographically-formatted data. Additionally, we outlined a novel "reduction" technique to derive important new environmental data through the integration of satellite and ground-based data. Our initial experiments show that by running a practical large-scale science data processing task in the pipeline with 150 moderately-sized Windows Azure virtual machine instances, we were able to produce analytical results in nearly 90X less time than was possible with a high-end desktop machine.

In this progress report, we discuss the issues encountered and our lessons learned since that time, attributable to the changes in Windows Azure, our better understanding and observing the capabilities of Windows Azure, our refinement of application requirements, etc. After reviewing the software architecture for our Windows Azure-based service, we address topics that include: how the newly-introduced Management APIs (to dynamically launch/shutdown computing instances) impacted our MODIS cloud application; how we selectively re-designed our service when Windows Azure transitioned from the "free model" (of the CTP) to the "cost model" in Feb 2010; how we addressed the need of end-scientists to interact with, debug, and use cloud-based services (the "reduction" routine is an arbitrary user-defined application that runs within the context of our MODIS service); and how we addressed the need of software developers to interact with, debug, and use cloud-based services (we built a custom application to monitor the performance of the Azure instances used by our MODIS service). We believe such observations can help contribute to the collective understanding of how to design and run large-scale scientific applications in Cloud infrastructures.

Biography:

Jie Li *received the BS degree in electrical engineering from Tsinghua University, China, and the MS degree in computing engineering from the University of Virginia in 2008. He is currently a PhD student in computer engineering at University of Virginia. His research interests include cloud computing and large-scale distributed systems*

Monitoring and Mining Sensor Data in Cloud Computing Platforms

Wen-Chih Peng

National Chiao Tung University, Taiwan
wcpeng@gmail.com

With the development of sensor devices and wireless networks, in recent years, a cyber-physical system (CPS) has attracted a considerable amount of research efforts. For example, vehicles with some sensor devices, such as GPS and CO2 sensors, are able to report their sensing readings to our developed cloud computing platform, called CarWeb. In light of these readings, our platform is able to estimate traffic status or the quality of air. We have developed a prototype platform CarWeb to collect GPS data and CO2 from vehicles. In CarWeb, users are able to obtain traffic status of roads interested. Since road networks contain a huge amount of roads, how to efficient estimate traffic status for a large-scale road network is a challenge problem. Though map services are implemented via cloud computing platforms, the road network is static information. Since sensor data is dynamically collected in our cloud computing platform, we should take real-time sensing data into consideration, which is an unique feature in our cloud computing platform. With the growth of sensor devices and wireless networks, it is expected that these sensor readings are collected in cloud computing platforms and then users are able to issue queries to collect and mine cyber-physical patterns from sensor data. Some challenge issues are (1) how to store the sensing data, (2) how to perform computations of the sensing data, and (3) how to mine knowledge via sensor data. In CarWeb, we demonstrate how we deal with the above issues via cloud computing platforms. In the future, cloud computing platforms should collect and mine sensor data to discover cyber-physical patterns, which justifies the importance of our CarWeb prototype.

Biography:

Wen-Chih Peng *was born in Hsinchu, Taiwan, R.O.C in 1973. He received the BS and MS degrees from the National Chiao Tung University, Taiwan, in 1995 and 1997, respectively, and the Ph.D. degree in Electrical Engineering from the National Taiwan University, Taiwan, R.O.C in 2001. Currently, he is an associate professor at the department of Computer, National Chiao- Tung University, Taiwan. Prior to joining the department of Computer Science and Information Engineering, National Chiao-Tung University, he was mainly involved in the projects related to mobile computing, data broadcasting and network data management. Dr. Peng serves as PC members in some prestigious conferences, such as ICDE, ACM CIKM and PAKDD. His research interests include mobile computing, network data management and data mining. He is a member of IEEE.*

Azure Academic Pilot

Krishna Kumar, Microsoft
kkumar@microsoft.com

Join us in this session to explore the various ways in which you can leverage Windows Azure for your teaching and research needs as part of the Azure Academic Pilot. This session will pick up where the "Microsoft Azure for Research" session from the previous day leaves off and walks through a start to finish application using the Windows Azure platform by placing special emphasis on the various design and architectural principles of the cloud. We will close by looking at some teaching and research grants available to academia and investigate integration points where content modules around cloud computing can be incorporated into existing or new courses in your curriculum.

Biography:

Krishna Kumar is an Academic Relations Manager at Microsoft where he partners with universities in the Midwest (WI, IA, IL, IN, MO) around their teaching and research needs. He also leads the academic effort around Azure nationwide by working with early adopters around cloud research and curriculum incorporation. He is also in the process of putting the finishing touches on a developer focused book on Azure for Addison-Wesley to be published in Summer 2010. He tweets at @KrishnaOnAzure and administers the http://AzurePilot.com portal. Krishna has been with Microsoft for 9 years in various roles working with multiple technologies, all of them developer focused. Forever the perennial student, he is finishing up his third Masters degree from Northwestern University in Medical Informatics. He lives with his wife and 1 year old son near Chicago.

Cloud Computing Support for Massively Social Gaming

Alexandru Iosup

Delft University of Technology, Netherlands
a.Iosup@tudelft.nl

Cloud computing is a promising paradigm for IT, in which the infrastructure, the platform, and even the software are outsourced services. As a new-formed community, we have already run out of hyperbole at the prospects of cloud computing. Still, we lack an understanding of the true capabilities of this paradigm. This talk covers the cloud-related research agenda and the vision for the future of a team of grid and peer-to-peer veterans becoming cloud explorers (www.pds.ewi.tudelft.nl).

Our main goal is to design and build fully functional, cloud-based systems, and to uncover in the process the fundamental laws that govern their operation. We currently focus on the continuous analysis of player status for massively social games such as RuneScape and FarmVille. As a use case, we consider third-parties that build and maintain player communities, and rely on continuous game analytics to improve the experience of their own users; these have millions of daily requests and relatively strict service response time requirements.

This talk includes a summary of our two-year, hands-on experience with clouds (http://www.st.ewi.tudelft.nl/~iosup/research_cloud.html). We have built CAMEO, a cloud-based, continuous analytics platform, and acquired and analyzed the status of almost 3,000,000 RuneScape players. We have also evaluated the performance of four clouds (Amazon Web Services, ElasticHosts, Mosso, and GoGrid) using various benchmarks. Further, we have assessed the performance variation over the complete year 2009 of AWS and Google App Engine services. Based on our experience, we foresee that clouds will become a viable solution for small enterprises and startup social applications.

Biography:

Alexandru Iosup *is assistant professor of computer science in the PDS Group of TUDelft. He has a PhD in computer science from TUDelft and an MSc in computer science from the Politehnica University of Bucharest, Romania. In the fall of 2006 he was a visiting researcher of Prof. Livny in the Condor group of U. Wisconsin-Madison, USA. He is the co-founder of the Grid Workloads Archive and of the Failure Trace Archive, the largest archives for workload and failure traces taken from parallel and distributed systems, world-wide. He led the team that performed the largest BitTorrent measurements and analysis to date. His work resulted in over 30 publications and several best paper awards including IEEE P2P (peer-to-peer work), IEEE/ACM SuperComputing (grid performance evaluation), and EuroPar (Massively Multiplayer Online Games). He is currently working on Massively Multiplayer Online Games, using clouds, grids, and P2P systems as infrastructure.*

CloudStudio: Collaborative, Cloud-Based Software Development

Bertrand Meyer

ETH Zurich, Switzerland
Bertrand.Meyer@inf.ethz.ch

Martin Nordio

ETH Zurich, Switzerland

Today's software production is increasingly distributed. Gone are the days of one-company, one-site projects; most industry developments involve teams split over locations, countries, cultures. Software tools have not kept up; they provide little if any support for this new reality of software development, relying on paradigms and practices such as traditional configuration management that were developed for earlier modes of operation.

The CloudStudio project at ETH Zurich is developing an IDE (integrated software development environment) enabling distributed projects to produce software "on the cloud". The environment enables every developer to work on a common project repository, shared "on the cloud". One of the main differences with traditional IDEs is that configuration management becomes unobtrusive; instead of the explicit update-modify-commit cycle, CloudStudio keeps track of successive versions and maintains the history automatically. Direct modification of a shared repository avoids raising spurious conflict notifications even when two developers are working on the same module; actual conflicts are detected early, and resolved through prevention rather painful post-hoc reconciliation of changes. Managers get an accurate and up-to-date view of the state of the development.

This presentation provides an overview of CloudStudio and describes the fundamental design choices.

Biography:

Bertrand Meyer *is professor of software engineering at ETH Zurich and chief architect at Eiffel Software. He is the author of a number of books, most recently the introductory programming textbook Touch of Class, and numerous articles.*

Cloud TV

Karin Breitman

PUC-Rio, Brazil
karin@inf.puc-rio.br

TV is dead – at least, as we know it. The traditional model, with clear separation between roles (producers, broadcasters and consumers) is giving way to a web based one, where users produce, distribute, combine and watch video content anywhere, anytime, and using a multitude of different devices.

The Cloud will play a major role in this scenario, as it will provide the infrastructure in which to store) & process (HaaS); platforms in which to encode and distribute (SaaS); and applications to submit, query and consume (AaaS) video content. In our presentation we'll discuss challenges and present solutions produced by our research group, in the following format:

Production challenges—The exponential growth of User-Generated-Content (UGG) makes it virtually impossible to estimate the volume of resources needed to run open submission systems, in particular those whose usage is seasonal. We introduce a cloud-based architecture that addresses this problem, and demonstrate an instance application that runs the registration system for candidates who wish to participate in the Brazilian Big Brother reality TV show .

Distribution challenges—The proliferation of mobile device types is pushing the demand for processing services to unprecedented levels. Every object is processed several times, to secure encoding standard compatibility with different devices (PC's, mobile phones, media centers, game consoles,etc), which support different codes (H.264 Baseline, Main and High Profiles, H.263,etc), as well as compression rates, to adapt to local storage and bandwidth restrictions. We introduce a generalization of the Map-Reduce architecture to tackle this problem, and demonstrate a running private-cloud implementation that reduces HD video encoding times dramatically – to pre-fixed values, independently of content duration .

Consuming challenges—Internet allows the mix and match of multi-source content. It's commonplace for networks to mix their own footage with UGC to provide: Haiti earthquake coverage featured as much mobile phone videos than professional footage. We discuss our recent research, applying Semantic Web and machine-learning techniques to improve indexing, querying and retrieval of data in the Cloud.

Biography:

Karin Breitman *Is a professor at PUC-Rio where she coordinates the Computer Engineering Program. Her research focuses on Software Engineering and its applications in Knowledge Representation, Requirements Engineering, Semantic Web and Cloud Computing. She is the Publications Director of the Brazilian Computer Society, and IFIP's TC-1 Secretary. In 2009 she was the PC Chair of two IEEE conferences (ICECCS'09 & SEW'09) as well as FME's sponsored ICFEM'09. Author of over 70 research papers in conference proceedings and journals received numerous research grants from public institutions and private industry, including NASA, CNRS, IBM and more recently Hewlett Packard. Her next book, "Analogy and Metaphor in Information Technology" is due Spring 2010, by Springer-Verlag. She actively collaborates with MS Research in their global gender initiative, and recently received the Google Brazil Women in Information Technology award for her contribution in promoting the importance of gender related issues in her country.*