

15-319: Cloud Computing Syllabus

School of Computer Science
Carnegie Mellon University, Qatar
Spring 2012

1 Overview

Title: Cloud Computing

Units: 9 units

Pre-requisites: A grade of “C” or better in 15-213, Introduction to Computer Systems

Lectures: Monday and Wednesday, 4:30 – 6:00 PM, Room 2147

Webpage: <http://www.qatar.cmu.edu/~msakr/15319-s12/>

Description:

This project-based course will give students a theoretical foundation and hands-on experience with the various technologies of the cloud computing paradigm. Cloud computing is the delivery of computing as a service, whereby distributed resources are provided by appropriate service suppliers and leased, rather than owned, by an end user as a utility (similar to electricity and water) over a network (typically the Internet). Cloud computing services are becoming ubiquitous and are being adopted by a growing number of fields. Organizations are recognizing the benefits of this new computing paradigm in terms of increased flexibility, elasticity as well as reduced upfront costs and carbon footprint.

The course will provide students with a thorough treatment of cloud computing and its applicability to commercial application development as well as research computing needs. The lectures will cover topics related to cloud infrastructure and software stack, programming models (e.g., MapReduce and Pregel), underlying distributed storage layers (e.g., HDFS and HBase), as well as enabling technologies such as virtualization. Students will also be exposed to various cloud frameworks and libraries (e.g., Mahout, Pig, and Hive). Since this is a project-based course, students will learn project design, management, implementation, testing and reporting skills. Students will also gain hands-on experience with a public cloud service (Amazon EC2, S3 and EBS), utilize it to lease and provision compute and storage resources and then program and deploy applications that use these resources. Students will use the Hadoop framework to solve large-scale data-intensive problems and then analyze the performance characteristics in the class project.

Instructors:

Majd F. Sakr

msakr@qatar.cmu.edu, 2121, 4454-8625, Office hours: Tuesday, 3:00 – 4:00 PM

Mohammad Hammoud

mhhammou@qatar.cmu.edu, 1013, 4454-8506, Office Hours: Thursday, 11AM – 12PM

Teaching Assistants:

Suhail Rehman

suhailr@qatar.cmu.edu, 2044, 4454-8680, Office Hours: By Appointment

Fan Zhang

zhang@qatar.cmu.edu, 1206, 4454-8482, Office Hours: By Appointment

2 Objectives

The course is meant to introduce students to the field of cloud computing. Students will work on a large semester-long project that will utilize the Amazon EC2 cloud. They will also learn about new programming paradigms that are developed for the cloud. Furthermore, they will understand and appreciate some of the current challenges and tradeoffs when mapping different applications to the cloud.

The course will serve as a firm foundation on many cloud computing principles and enablers such as distributed file systems and virtualization. Students will be able to design and implement parallel algorithms to efficiently distribute data intensive computation over virtualized cloud platforms. The class project in this CS 319 will focus on implementing MapReduce real-world applications, deploy them on the cloud and characterize their performances. As a result, students will have the foundation needed to match the future needs in the emerging field of cloud computing.

The course has three main goals:

- To learn the core concepts and principles of cloud computing as well as identify and explore some of the emerging research challenges in clouds.
- To gain hands-on experience in using cloud computing infrastructure by designing, developing and deploying applications on cloud infrastructures.
- To work on a large research project in cloud computing.

Through these objectives, the course will transform your computational thinking from designing applications for a single computer system to designing applications for a cloud distributed system.

3 Learning Outcomes

The primary learning outcome of the course is three-fold:

1. *Students will explain the **core concepts of the cloud computing** paradigm: how and why this paradigm shift came about and the influence of several **enabling technologies** in cloud computing.*
2. *Students will examine the process of **working on a large research project** under the mentorship of a teaching staff member. They will study **how applications for clouds are written, deployed and analyzed**. In the process, they will develop the needed skills to go through project planning, design, implementation, analysis and reporting.*
3. *Students will identify some of the emerging **cloud research challenges**.*

3.1 Understanding the core concepts of cloud computing and the enabling technologies:

Students will learn the core concepts of cloud computing. They will understand how the cloud computing paradigm evolved over the past few years as an answer to the growing needs of organizations. Cloud computing is an amalgam of various technologies. Students will be able to discuss many of these technologies including:

- *Programming Models*
- *Virtualization*
- *Distributed File Systems and Cloud Storage*
- *Emerging Cloud Tools*

3.1.1 Programming Models

Traditional programming models might not work efficiently in clouds. Students will identify the two main classical programming models, shared memory and message passing, as well as apply the novel programming models that are commonly adopted in clouds. Specifically, students will:

- Identify the design characteristics of the shared memory, message passing and MapReduce programming models.
- Describe the relationship between programming models and the architecture of the underlying system.
- Explain the Hadoop MapReduce program flow and how it communicates with the Hadoop distributed file system (HDFS).
- Identify several programming models as case studies such as Dryad, Pregel and GraphLab.

3.1.2 Virtualization

Students will explain the fundamental concepts of virtualization, where a state of a computer is abstracted from the underlying hardware. They will describe how virtualization applies to cloud computing, and identify various capabilities provided by virtualization to cloud providers and users. Specifically, students will:

- Discuss the types of virtualization: process versus system and software-based (or full virtualization) versus hardware-assisted (or paravirtualization) virtualizations.
- Discuss resource virtualization: CPU, Memory, Disk, and Network virtualizations.
- Describe distributed resource management, distributed resource monitoring, and distributed scheduling in clouds.
- Identify Xen and VMWare as case studies.

3.1.3 Storage Technologies and Distributed File Systems

Storage technologies and distributed file systems play a major role in enabling cloud computing, by allowing for fast, reliable, and parallel access to large amounts of data distributed across multiple machines. Students will identify storage technologies suitable for clouds as well as describe the fundamental principles of distributed file systems (DFSs) and how they apply to cloud computing. Specifically, students will:

- Identify external, network-based storage suitable for clouds: SAN, NAS, and iSCSI.
- Discuss various DFS architectures: cluster-based versus client-server architectures.
- Describe various aspects of DFSs including communication, synchronization, replication, fault tolerance, and security.
- Identify the difference between distributed and parallel file systems.
- Identify GFS and HDFS, PVFS, BigTable and Hbase as case studies.

3.1.4 Emerging Cloud Tools

One criticism of cloud programming models is that the development cycle might take long time. For instance, writing a MapReduce program involves coding the map and reduce functions, compiling and packaging the program, submitting the job(s), and retrieving the results. Researchers and engineers might require a faster model to quickly mine huge datasets. In this course, students will:

- Identify various Hadoop extensions which simplify large-scale data processing, such as Apache Pig and Hive.
- Apply the Apache Mahout Machine learning library and explore its usage in some machine learning tasks such as clustering and classification.
- Identify how to build general distributed applications using Hadoop's distributed coordination service, ZooKeeper.

3.2 Building Cloud Applications

Students will explore the applicability of different application domains to cloud computing. Specifically, students will:

- Embark upon a full-semester research project that will allow them to gradually master MapReduce and investigate its applicability to various domains, such as natural language processing, machine learning, bioinformatics and image processing.
- Gain insights on MapReduce performance under various domains, analyze its ensuing behaviors, and optimize performance through making changes in cluster configurations and provisioning.
- Apply workload characterization as a crucial component for their performance analysis.

Each student will be mentored by a teaching staff member and will deliberately acquire the required skills to pursue project planning, design, implementation, analysis and result reporting, much needed in academia as well as industry.

3.3 Emerging Research Challenges

While many existing techniques served in realizing the realm of cloud computing, several new research challenges swiftly emerged in an attempt to enable the full potential of the paradigm. Students will identify the following research challenges:

- Cloud security such as developing cloud security models, end-to-end methods for enforcing security policies, and programming models with privacy-aware APIs.
- Quality of service (QoS) and service level agreements (SLAs) (e.g., completion time, availability, response time) in clouds.
- Energy-efficient clouds that entail more elaborate energy consumption metrics, energy-aware cloud applications, and data centers with renewable energy sources (e.g., solar and wind powers) and low power processing units (e.g., GPUs).

4 Textbooks

The primary textbook for this course is:

- Tom White, *Hadoop: The Definitive Guide (2nd Edition)*, O'Reilly Media, 2010

In addition, it will be useful to have the following reference books:

- James E. Smith and Ravi Nair, *Virtual Machines : Versatile Platforms for Systems and Processes*, Morgan Kaufman, 2005
- Jurg van Vilet and Flavia Paganelli, *Programming Amazon EC2*, O'Reilly Media, 2011
- Jothy Rosenberg and Arthur Mateos, *The Cloud at Your Service*, Manning Publications, 2010
- Sean Owen, Robin Anil, Ted Dunning and Ellen Friedman, *Mahout in Action*, Manning Publications, 2011
- Chuck Lam, *Hadoop in Action*, Manning Publications, 2011

We have several reference books in the library covering most of the topics of the course. We will also be reading tutorials, journals and conference publications on the subject.

5 Course Organization

Your participation in the course will involve several forms of activity:

- Attending and participating in the lectures and discussions
- Projects and Posters
- Project and Poster Designs and Status Updates

Attendance will be taken at the beginning of each lectures, it will be worth 5% of your grade. Before each class, you are required to briefly read about the topics that will be covered. You will be responsible for all material presented during the lectures.

6 Getting Help

For urgent communication with the teaching staff, it is best to send an email (preferred) or call the office phone. If you want to talk to a staff member in person, remember that our posted office hours are merely nominal times when we guarantee that we will be in our offices. You are always welcome to visit us outside of our office hours if you need help or want to talk about the course.

We ask that you follow a few simple guidelines. Prof. Sakr, Dr. Hammoud, Suhail and Dr. Zhang normally work with their office door open and welcome visits from students whenever the doors are open. However, if their door is closed, then they are busy with a meeting or a phone call and should not be disturbed.

We will use the course web-page as the central repository for all information about the class. Using the web-page, you can:

1. Obtain copies of any handouts or assignments. This is especially useful if you miss class or you lose your copy.
2. Find links to any electronic data you need for your assignments
3. Read clarifications and changes made to any assignments, schedules, or policies.
4. Provide healthy feedback about the course

You can use the mailing list (15319-s12@lists.qatar.cmu.edu) to post messages, make queries about the course and specific project requirements. The messages on this mailing list will be distributed to all the students and staff of the course.

7 Policies

Working Alone on Project Phases and Posters

Project phases and posters that are assigned to single students should be performed individually.

Handing in Project Phases and Posters

All assignments/projects are due at 11:59 PM (one minute before midnight) on the specified due date. All hand-ins are electronic using the AFS file system: /afs/qatar.cmu.edu/usr16/msakr/www/15319-s12/handin/*userid*/, *userid* is your qatar user id.

Appealing Grades

After each project phase is graded, you have seven calendar days to appeal your grade. All your appeals should be provided in writing. If you are still not satisfied, please come and visit Prof. Sakr. If you have questions about an exam grade, please visit Prof. Sakr directly.

8 Assessment

Final Grade Assignment and Assessment methods

Each student will receive a numeric score for the course, based on a weighted average of the following:

1. **Project:** The project will count a combined total of 75% of your score. There are 3 project phases throughout the course. The first phase is worth 15% each. The second and third phases are worth 30% each, and it will involve a presentation and a paper as well as the project code. Take into account that small differences in scores can make the difference between two letter grades.

You are encouraged to submit the project phase deliverables on time. For the first two project phases, the following rules apply. If you submit one day late, there will be deducted 25% of the project score as penalty. If you are two days late, 50% will be deducted. The project will not be graded (and you will receive a zero score) if you are more than two days late. However, there is a **grace-days quota** for projects; you are given **3 grace days** for the first two project phases. You can use the grace days as needed. For example, you can submit your project 1, three days late and still not get any penalty. Your penalty starts from 4th day after the deadline if you use your grace days. However, since you have used up all your grace days from your quota, you do not have any grace days for other projects. Plan how to utilize your grace day quota judiciously.

Note that the final project phase is unique. You cannot use grace days for final project. There will not be any penalty system for this project either. If you are one day late in submitting final project, your project will not be graded (and you will receive a zero score).

2. **Student Project Update Presentations:** You will be required to brief the instructors and the class about the status of your project in a short presentation that outlines your project status, the milestones you have achieved and the next steps to completing the project. At the end of each project, you are required to present your project to the class as well. These will count towards 20% of your final grade.
3. **Class participation and attendance:** Your attendance and participation in the different discussions held in class will account towards 5% of your final grade.

Type	#	Weight
Project Phases I, II & III	3	75%
Project Update Presentations	6	20%
Class Participation and Attendance	28	5%

Grades for the course will be determined by absolute standards. The total score will be plotted as a histogram. Cutoff points are determined by examining the quality of work by students on the borderlines. Individual cases, especially those near the cutoff points may be adjusted upward or downward based on factors such as attendance, class participation, improvement observed throughout the course, and special circumstances.

9 Cheating

Each project must be the sole work of the student turning it in. Projects will be closely monitored by automatic cheat checkers, and students may be asked to explain any suspicious similarities with any piece of code available. The following are guidelines on what collaboration is authorized and what is not:

What is cheating?

1. Sharing code or other electronic files: either by copying, retyping, looking at, or supplying a copy of a file.
2. Sharing written assignments: Looking at, copying, or supplying an assignment.

What is NOT cheating?

1. Clarifying ambiguities or vague points in class handouts.
2. Helping others use the computer systems, networks, compilers, debuggers, profilers, or other system facilities.
3. Helping others with high-level design issues.
4. Helping others debug their code.

Cheating in projects will also be strictly monitored and penalized. Be aware of what constitutes cheating (and what does not) while interacting with students. You cannot share or use written code, and other electronic files from students. If you are unsure, ask the teaching staff.

Be sure to store your work in protected directories. The penalty for cheating is severe, and might jeopardize your career – cheating is not worth the trouble. By cheating in the course, you are cheating yourself; the worst outcome of cheating is missing an opportunity to learn. In addition, you will be removed from the course with a failing grade. We also place a record of the incident in the student's permanent record.

10 Class Schedule

Table 1 shows the tentative schedule for the class. The schedule also indicates the project activities. Any changes will be announced on the class distribution list (15319-s12@lists.qatar.cmu.edu). An updated schedule will be maintained on the class Web page.

Week	Session	Date	Topic	Teaching Method	Instructor	Projects
1	1	16 Jan	Administrivia and Introduction to Cloud Computing	Lecture	MFS	Project Start
	2	18 Jan	Introduction to Cloud Computing and Cloud Software Stack	Lecture	MFS	
2	3	23 Jan	Course Project and Amazon AWS	Lecture	MFS	
	4	25 Jan	Principles of Parallel Programming	Lecture	MFS	Project Update
3	5	30 Jan	Introduction to Programming Models	Lecture	MFS	
	6	1 Feb	Introduction to MapReduce	Lecture	MHH	Project Update
4	7	6 Feb	MapReduce Algorithms	Lecture	MHH	
	8	8 Feb	Apache Mahout	Guest Lecture	Shannon	Project Update
5	9	13 Feb	Pregel, Dryad, GraphLab	Lecture	MHH	
	10	15 Feb	Major Project Milestone Presentations	Project Presentation		Project Phase I End, Paper V1 Due, Project Phase II Start
6	11	20 Feb	Introduction to Cloud Storage and Distributed File Systems (DFSs)	Lecture	MFS	
	12	22 Feb	Cloud Storage and DFSs– HDFS, PVFS	Lecture	MHH	Project Update
7	13	27 Feb	Cloud Storage and DFSs- BigTable/HBASE	Lecture	MHH	
	14	29 Feb	Apache Pig, Hive, Zookeeper	Lecture		Project Update
8			Spring Break – No Classes			
	15	12 Mar	Introduction to Virtualization	Lecture	MHH	
9	16	14 Mar	Resource Virtualization (CPU/Memory/Disk/Network)	Lecture	MHH	Project Update
	17	19 Mar	Virtualization Case Study: Xen	Lecture	MHH	
10	18	21 Mar	Virtualization Case Study: VMware	Lecture	MHH	Project Update
	19	26 Mar	Benchmarking and Application Characterization	Lecture	MHH	
11	20	28 Mar	Major Project Milestone Presentations	Project Presentation		Project Phase II End Project Phase III Start
	21	2 Apr	Cloud Monitoring and Diagnostics	Lecture	MSR	
12	22	4 Apr	Cloud Security	Lecture	Thierry	Project Update
	23	9 Apr	Cloud QoS and SLAs	Lecture	MFS	
13	24	11 Apr	Green Cloud	Lecture	MFS	Project Update
	25	16 Apr	How to Write a Scientific Paper	Lecture		
14	26	18 Apr	Guest Lecture	Lecture		Project Update
	27	23 Apr	Final Project Presentations I	Project Presentation		Project End
15	28	25 Apr	Final Project Presentations II	Project Presentation		Project End

Table 1: Tentative time-line of the course. Notations used are as given below:

- *Instructors: MFS=Majd Sakr, MHH=Mohammad Hammoud, MSR=Suhail Rehman, FZ=Fan Zhang*