

Wright State University

## CORE Scholar

---

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

---

Spring 2012

### CS 499/699: Cloud Computing

Keke Chen

*Wright State University - Main Campus, keke.chen@wright.edu*

Follow this and additional works at: [https://corescholar.libraries.wright.edu/cecs\\_syllabi](https://corescholar.libraries.wright.edu/cecs_syllabi)



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

---

#### Repository Citation

Chen, K. (2012). CS 499/699: Cloud Computing. .  
[https://corescholar.libraries.wright.edu/cecs\\_syllabi/466](https://corescholar.libraries.wright.edu/cecs_syllabi/466)

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact [library-corescholar@wright.edu](mailto:library-corescholar@wright.edu).

# CS499/699 – Cloud Computing

## Instructor

Keke Chen, Ph.D., Office: 385 Joshi  
Email: [keke.chen@wright.edu](mailto:keke.chen@wright.edu) , Phone: (937) 775-4642  
Web: <http://www.cs.wright.edu/~keke.chen/>  
office hours: 2-3:30pm TR

## Course Description

This is an introductory course to cloud computing. In this course, we will explore a few aspects of cloud computing: distributed data crunching with MapReduce, cloud and datacenter filesystems, virtualization, cloud security&privacy, Amazon Web Services, and interactive web-based applications. Students are expected to read extra materials including papers and online resources, finish several mini projects, and take the final exam. Participation in the class discussion is strongly encouraged. Guest speakers might be invited for some particular topics. (3 Hours Lecture + 1 Hour lab).

**Class meeting time:** 8-9:15pm, TR  
**Classroom:** Russ 208

## Prerequisite:

CS400/600, CEG433/633 (good knowledge of data structures, algorithms, databases, operating systems, and distributed computing). The projects will require good Java programming skills and sufficient knowledge of Python and script programming. Be prepared to learn new programming frameworks. You should have good experience working in the Linux environment, since our projects will be done in Linux.

## Text Books and Materials

There is no textbook for this course. All materials will come from recently published papers and online documents. Please check some sample references at the end of this page.

## Grading Policy

Mini projects	60%
Reading	10%

Final exam	20%
Class participation	10%

A[90-100] B[80-89] C[70-79] D[60-69] F[<60]. The instructor will curve the final grades based on the distribution of scores.

## Covered Topics (tentative)

1. Introduction	1 class
2. Cloud and datacenter file systems	1 class
3. MapReduce programming	3~4 classes
4. Virtualization	1 classes
5. Amazon Web Services and Eucalyptus	2~3 classes
6. Interactive Web-based applications	1~2 classes
7. Security and Privacy issues	2 classes
8. Mini project discussion	2 classes
9. Advanced Research Topics	1~2 classes

These topics may be covered in different ordering.

## Mini-projects

Several mini projects will be given. Students will get familiar with hadoop, map-reduce programming, AWS, and possibly interactive applications in these projects.

## References

1. “Above the Clouds: A Berkeley View of Cloud Computing”, Michael Armbrust, et al. Technical Report, University of Berkeley, 2009, <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf>
2. “The Claremont Report on Database Research”, 2008, <http://db.cs.berkeley.edu/claremont/claremontreport08.pdf>
3. hadoop, <http://hadoop.apache.org/>
4. Pig <http://hadoop.apache.org/pig/>
5. Hbase <http://hadoop.apache.org/hbase/>
6. Hive <http://hadoop.apache.org/hive/>
7. “The Google File System”, Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung, OSDI, 2003, <http://labs.google.com/papers/gfs-sosp2003.pdf>
8. “Bigtable: A Distributed Storage System for Structured Data”, Fay Chang, et al. OSDI 2006, <http://labs.google.com/papers/bigtable-osdi06.pdf>
9. “MapReduce: Simplified Data Processing on Large Clusters”, Jeffrey Dean and Sanjay Ghemawat, OSDI 2004, <http://labs.google.com/papers/mapreduce-osdi04.pdf>
10. “Map-Reduce for Machine Learning on Multicore”, Cheng-Tao Chu et al. NIPS, 2006, <http://www.cs.stanford.edu/people/ang/papers/nips06-mapreducemulticore.pdf>

11. A comparison of approaches to large-scale data analysis. A. Pavlo et al. SIGMOD2009, <http://database.cs.brown.edu/sigmod09/benchmarks-sigmod09.pdf>
12. Amazon Web Services, <http://aws.amazon.com/>
13. Eucalyptus (<http://www.eucalyptus.com/>)
14. AppEngine <http://code.google.com/appengine/>
15. Azure <http://www.microsoft.com/azure/>
16. “Xen and the Art of Virtualization”, Paul Barham, et al., SOSP 2003, <http://www.cl.cam.ac.uk/research/srg/netos/papers/2003-xensosp.pdf>
17. “Benchmarking cloud serving systems with YCSB” Brian F. Cooper, Adam Silberstein, Erwin Tam, Raghuram Ramakrishnan, Russell Sears, ACM Symposium on Cloud Computing, 2010, [\[PDF\]](#)
18. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice) by Tim Mather, Subra Kumaraswamy and Shahed Latif, 2009

more recent papers...