COMP90015: Distributed Systems

Dr. Rajkumar Buyya

Redmond Barry Distinguished Professor & Director
Cloud Computing and **D**istributed **S**ystems (CLOUDS) Laboratory
School of Computing and Information Systems
The University of Melbourne, Australia
http://www.buyya.com

http://clouds.cis.unimelb.edu.au/~rbuyya/

Teaching Staff

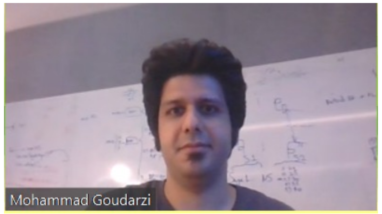
- Lecturer 1: Prof. Rajkumar Buyya
  - Email: rbuyya@unimelb.edu.au
  - Office: 2.333, Melbourne Connect Building, 700 Swanston St, Carlton
  - Raj’s Consulting Time:
    - After our formal lecture, I will hand around upto 30 minute in case any of you want to clarify or discuss or talk to me.
    - We will offer one possible during a week before the Assignment deadline if required.

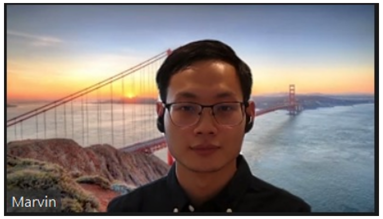
Head Tutor

- Head Tutor: Siddharth Agarwal
- Handles lectures in my absence and assists with labs/projects.
  - Email:siddharth.agarwal1@unimelb.edu.au
  - Office: Desk 2.126 (in same level as Prof. Buyya)
- Consulting Time:
  - We will offer one possible during a week before the Assignment deadline if required.

Tutors (6)

- Handle all tutorials/workshops, assisting with labs/projects and marking of assignments (associated with their tutorials)
  - Siddharth Agarwal siddharth.agarwal1@unimelb.edu.au
  - Mohammad Goudarzi: mgourdazi@unimelb.edu.au
  - Marvin Bai: marvin.bai@unimelb.edu.au
  - Duneesha Fernando: dtfernando@student.unimelb.edu.au
  - Mashnoon Islam: mashnoon.islam1@unimelb.edu.au
  - Hoa Nguyen: thanhhoan@student.unimelb.edu.au
- Note: Please contact only those who are in-charge of your tutorial.



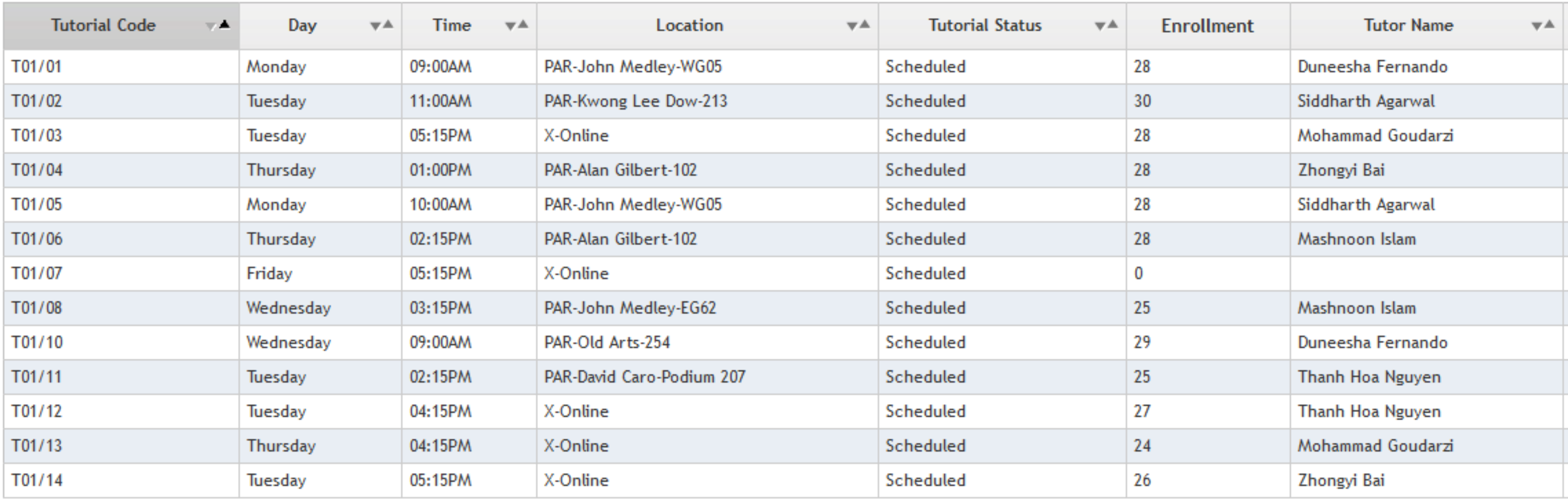


Web and Course Schedule

- Course Web Site:
  - http://clouds.cis.unimelb.edu.au/652/
  - Note: LMS gives link to this.
  - All announcements, notes, etc. via this page only. LMS can be used for discussions, video lectures, and for assignments.
- Lectures:
  - Time:
    - Friday: 3:15-5.15pm - 2 hours – with 5-10 minutes break.
    - Venue: PAR-Old Arts-122 (Public Lecture Theatre - PLT)
- Workshops/Tutorials – 12
    - Each session accommodates ~28; Must attend your own Tutorial
    - Please make friends in your tutorial!

Tutorials: Time, Venue and Tutors

- Hybrid - Campus/Online



Tutorial Code	Day	Time	Location	Tutorial Status	Enrollment	Tutor Name
T01/01	Monday	09:00AM	PAR-John Medley-WG05	Scheduled	28	Duneesha Fernando
T01/02	Tuesday	11:00AM	PAR-Kwong Lee Dow-213	Scheduled	30	Siddharth Agarwal
T01/03	Tuesday	05:15PM	X-Online	Scheduled	28	Mohammad Goudarzi
T01/04	Thursday	01:00PM	PAR-Alan Gilbert-102	Scheduled	28	Zhongyi Bai
T01/05	Monday	10:00AM	PAR-John Medley-WG05	Scheduled	28	Siddharth Agarwal
T01/06	Thursday	02:15PM	PAR-Alan Gilbert-102	Scheduled	28	Mashnoon Islam
T01/07	Friday	05:15PM	X-Online	Scheduled	0	
T01/08	Wednesday	03:15PM	PAR-John Medley-EG62	Scheduled	25	Mashnoon Islam
T01/10	Wednesday	09:00AM	PAR-Old Arts-254	Scheduled	29	Duneesha Fernando
T01/11	Tuesday	02:15PM	PAR-David Caro-Podium 207	Scheduled	25	Thanh Hoa Nguyen
T01/12	Tuesday	04:15PM	X-Online	Scheduled	27	Thanh Hoa Nguyen
T01/13	Thursday	04:15PM	X-Online	Scheduled	24	Mohammad Goudarzi
T01/14	Tuesday	05:15PM	X-Online	Scheduled	26	Zhongyi Bai

- Pls choose available slot (if not yet)

Zoom Links for each Tutorial: Please login into LMS/Canvas
Please join only during your Tutorial slot!

Wide-Background of Students???

- Master of IT
  - MIT (Comp), MEDC/MIT (DC), MIT (Spatial, CyberSec, HCI, AI)
    - MIT (Distributed Computing) – foundation subject.
- Master of Computer Science
- ME (Software Engineering)
- Master of Data Science
- ++ Students from all over the world joining our Masters programs with varied background.
- So, please understand that we are trying our best to satisfy all of you although it is difficult to please everyone ☺

Background expectation

- Pre-requisites (All completed):
  - COMP90041 Programming and Software Development (Java)
  - COMP90038 Algorithms and Complexity
  - COMP90007 Internet Technologies (No Sockets/Threads taught)
  - **OR** Equivalent subjects (with formal evidence)
- If you know “MORE” than pre-requisite subjects, then this subject is NOT for you.
  - Better take Advanced related/follow-up subject if you know “More” than pre-requisite subject coverage (e.g., UniMelb: OS and Network Services).

DS subject is a “foundation” (pre-requisite) for many advanced subjects

- Distributed Algorithms
- Mobile Computing Systems Programming
- Cluster and Cloud Computing
- Distributed Computing Project (for MIT(DC))
- Sensor Networks and Applications
- Parallel and Multi-core Computing
- Some special offerings:
  - Stream Computing?
  - Management and Mining of Spatio-Temporal Data (MapReduce application)

Why study distributed computing now?

- We have started MEDC, now MIT(DC) degree at a time when distributed systems, particularly the Web and Internet applications/services, are of unprecedented interest and importance.
  - Microsoft .NET
  - Oracle – Oracle 21c
  - IBM – On Demand
  - SAP – enterprise management software
  - Cloud Computing: Amazon EC2, Microsoft Azure, Google AppEngine, Aneka, Force.com, Alibaba China Cloud, Apple iCloud
  - Social Networks: Facebook, WhatsApp, Skype, WeChat….
  - Academic R&D worldwide: Service computing, e-Science, etc.
- MIT(DC) degree and this subject in particular aims to convey insight into, and knowledge of the principles and practice underlying the design of distributed systems.
- The depth covered in this subject enables you to evaluate existing systems or design new ones.

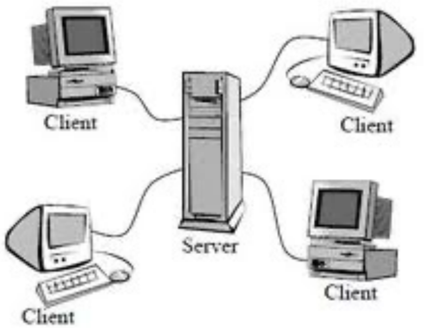
DS Subject Overview

- Part I: Foundations – approx. 5 weeks
  - Introduction, Inter-process Communication, Socket and Thread Programming, and System Models
- Part II: Programming and Principles – 4 weeks
  - Distributed Objects and Programming,
  - Operating System support services, Distributed Shared Memory Systems
- Part III: Paradigms/Platforms - 3 weeks:
  - RMI, Kerberos, NFS etc. taught during Part I & II
  - Distributed File Systems, Security and Naming Services
- Guest Lectures / Advanced Topics (not in exam)
  - CDN, Cloud, BlockChain, IoT, and industrial applications
- Depth of some parts may be reduced as our School has dedicated subjects on some of these topics:
  - Distributed Algorithms, Software Systems Security, Cluster and Cloud Computing, High-Performance Database Systems

Course Assessment

- Project work and some short assignments:
  - During the semester worth 40%
    - Assignment 1 (Single): 15%
    - Assignment 2 (Single): 25%
- Written examination:
  - A written examination (three hours) at the end of the semester worth 60%
- All components **must** be completed satisfactorily (50% marks) to pass the subject.

Assignment 1

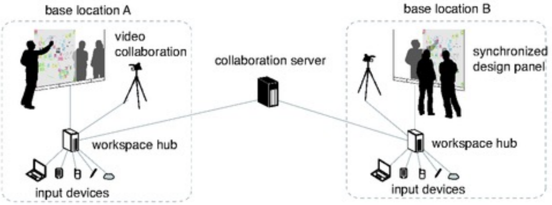


Each server provides services to multiple clients.

- Multi-Threaded Dictionary Server
  - Design and Implementation of a Simple Multi-Threaded Distributed System Supporting Access to a Remote Dictionary
- Aim:
  - Enhance Understanding of Socket Programming and Multi-Threading
  - Gain experience in implementing a simple distributed, client server application.
  - "Using a client-server architecture, design and implement a multi-threaded server that returns the meaning of a word as stored in a remote dictionary."
  - Do some smart design/architecture (networking, storage)!

Assignment 2

- Distributed Applications Project
  - To be decided:
    - Distributed, Shared White Board OR Net Games along with a chat box.



- Individual Assignment like A1
- You are given a chance to show some creative thinking / architecture (e.g. you can “use client/server or P2P”, “TCP or UDP”)
- Will recommend as Multi-stage project (even if not assessed at each stage)

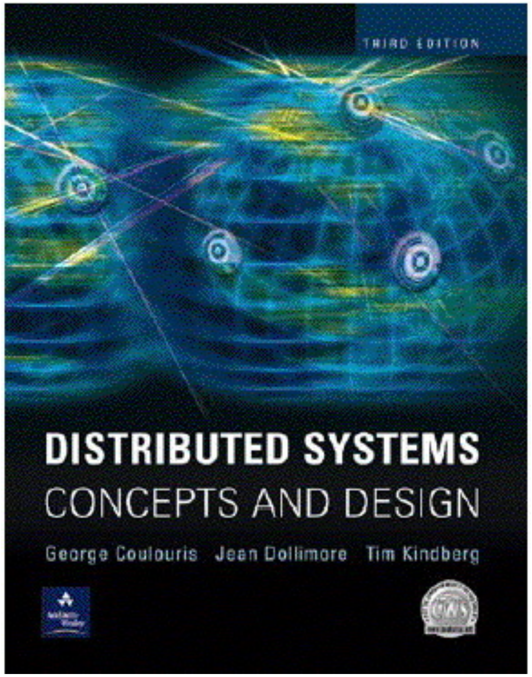
Computational Resources

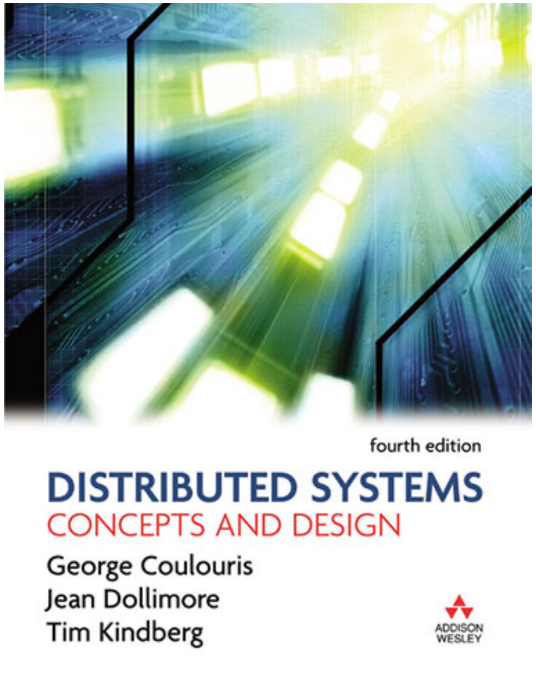
- Your laptop!
  - Use it for both assignments..
- Uni. Computing Resources:
  - Can also be used for simple assignments and learning
  - For demonstration of assignments (along with your own laptops)
  - Your own computing resources!

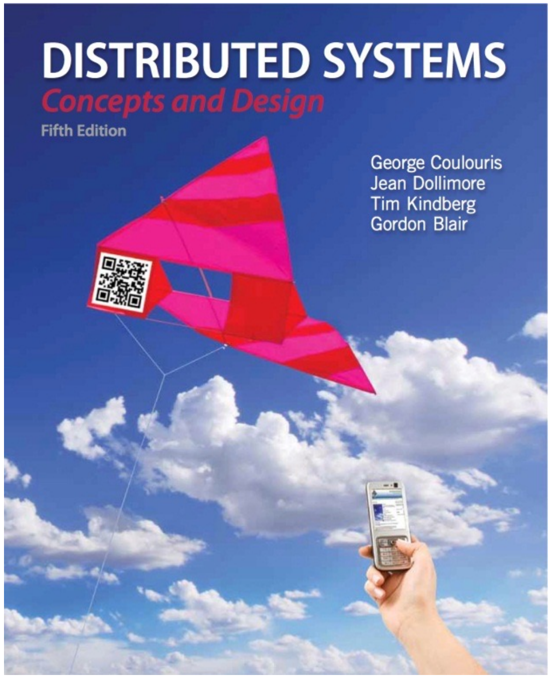
Books and References

- Main Text Book:
  - CDK: G. Couloris, J. Dollimore, T. Kinberg, and G. Blair, ***Distributed Systems - Concepts and Design***, 5th Edition, Addison-Wesley, Pearson Education, UK, ISBN 0132-143-011.
- Programming Reference:
  - R. Buyya, S. Selvi, X. Chu, **"Object Oriented Programming with Java: Essentials and Applications",** McGraw Hill, New Delhi, India, 2009.
  - Sample chapters: clouds.cis.unimelb.edu.au/~rbuyya/java/
- Research Articles:
  - To be supplied by the Lecturer (if used)!

Text Book

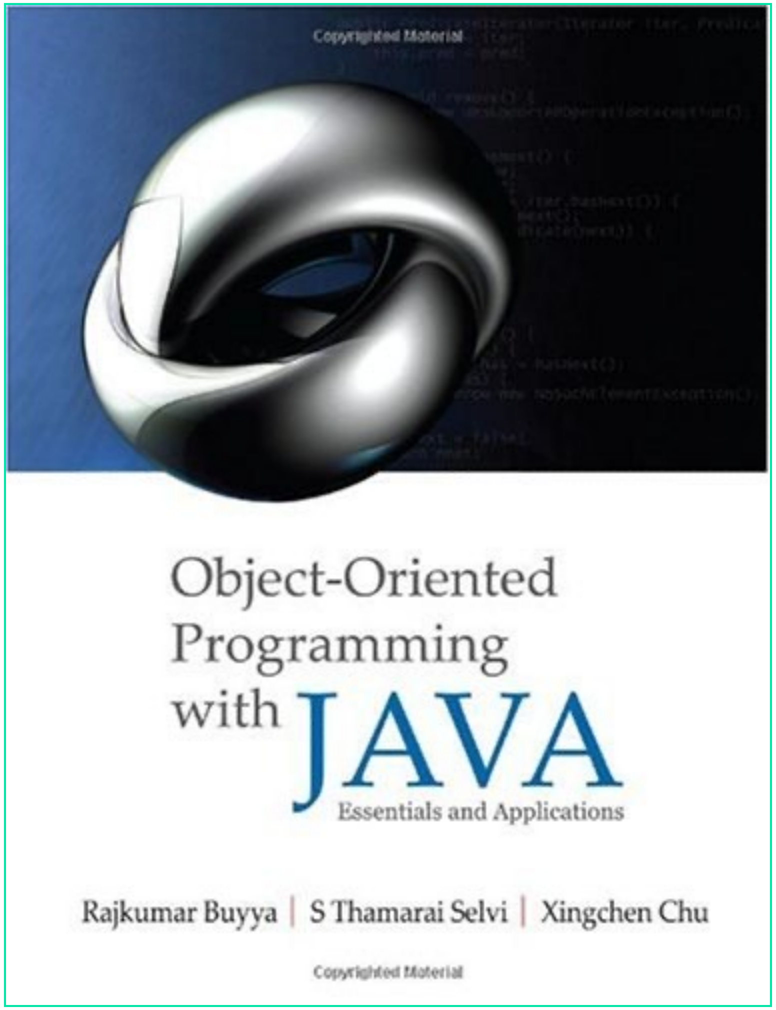






G. Couloris, J. Dollimore, T. Kinberg, and G. Blair,
Distributed Systems - Concepts and Design,
5th Edition, Addison-Wesley/Pearson Education, UK.

Programming Reference


Copyrighted Material
Object-Oriented Programming with JAVA
Essentials and Applications
Rajkumar Buyya | S Thamarai Selvi | Xingchen Chu
Copyrighted Material

Buyya, R. Selvi, S.T., Chu, X., ***Object Oriented Programming with Java: Essentials and Applications***, McGraw Hill, New Delhi, India.

Sample chapters at book website: http://clouds.cis.unimelb.edu.au/~rbuyya/java/

Presentation Slides

- Usually on the web before the lecture
- They may be fine tuned/updated slightly a day before the lecture to reflect latest developments
  - No need to read Today’s lecture content beforehand!
  - You only need to read & understand previous lecture!
  - Do online Quiz (Multiple choice test) on previous lecture topic prior to tutorial –ask Q on difficult topic from quiz.
- Mostly derived from the textbook.
  - *Please procure (or own) the prescribed textbook.*
- Good ideas and figures from alternative textbook or reference may also be used.

What do we expect from you?

- 1. Regular attendance of lectures
  - Pay full attention, be enthusiastic, fully committed to learn new things, ask questions during the class (especially in Tutorials), participate in discussion.
  - If the class overlaps with others, please choose one subject. This is a great **favour you can do** for yourself.
- 2. Review previous lecture material before coming to the class. – read material from the Textbook
- 3. Start working on assignments right from the day they are announced and submit on time.
- 4. If you have some problem with the lectures/subject/??, please discuss with us **early**.
  - Don’t take out your frustrations on me during QoT/SES☺

QoT (Quality of Teaching) / SES (Subject Experience Survey)

- # I had a clear idea of what was expected of me in this subject

5. Strongly agree :

4. Agree :

3. Neutral :

2. Disagree :

1. Strongly disagree:

Mean :