COMP3100

Distributed Systems

Session 1, In person-scheduled-weekday, North Ryde 2024

School of Computing

Contents

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff
Unit Convenor and Lecturer
James Zheng
james.zheng@mq.edu.au

Lecturer and Tutor
James Phung
james.phung@mq.edu.au

Tutor
Jayden King
jayden.king@mq.edu.au

Tutor
Ferdous Sharifi
ferdous.sharifi@mq.edu.au

Tutor
Yuzhe Tian
yuzhe.tian@mq.edu.au

Credit points
10

Prerequisites
130cp at 1000 level or above including (COMP2100 or COMP202) and (COMP2250 or COMP247)

Corequisites

Co-badged status
COMP6105

> Unit description
>
> A distributed system traditionally refers to a group of networked computers; however, it should be today understood in a much wider sense including applications consisting of multiple processes. This unit studies the fundamentals of distributed systems from both hardware perspective and software perspective. The unit also gives some hands-on experience. Topics include distributed systems principles (concurrency and scheduling), paradigms (cloud computing, serverless computing, mobile computing and Internet of Things), architectures (client-server model, peer-to-peer model, microservices and distributed file systems) and techniques (shared memory, message-passing and containersation).

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at https://www.mq.edu.au/study/calendar-of-dates

Learning Outcomes

On successful completion of this unit, you will be able to:

ULO1: Describe the complexities of distributed system development and approaches to solve those complexities.

ULO2: Apply theoretical principles and models to design distributed systems.

ULO3: Explain important issues in modern distributed systems.

ULO4: Configure and deploy cloud services and technologies that support distributed applications.

ULO5: Analyze and design distributed systems.

General Assessment Information

The University's academic honesty policy will be enforced. You may assist your fellow students with general concepts, pointers to resources and useful tools or commands that are publicly available. You may not become involved in any way in helping a fellow student to find the solution to their particular task, nor may you share with them any aspect of the solution of your particular task. If you decide to develop or modify a tool (including software tools, procedures or methods) to assist you in solving your programming task, you may not provide that tool to your fellow students, nor may you publish it.

Each assessment task must be the sole work of the student turning it in. Any cheating will be handled under the University's Academic Honesty Policy.

Requirements to Pass this Unit

To pass this unit you must:

- Attempt all assessments, and

- Achieve a total mark equal to or greater than 50%, and

- Participate in weekly practicals, and achieve a minimum of 8 of the 20 marks

Hurdle assessment (Practical tasks)

This unit has weekly practical classes and you must demonstrate your progress in developing and communicating knowledge and skills and achieve a minimum of 8 marks of the 20 marks allocated, i.e., 40%. Failure to meet this requirement may result in a **fail** grade for the unit. In the case of missing your designated practical class or failing to show your progress, another opportunity might be given if a Special Consideration is approved (see below).

Late Assessment Submission Penalty

Unless a Special Consideration request has been submitted and approved, a 5% penalty (of the total possible mark) will be applied each day a written assessment is not submitted, up until the 7th day (including weekends). After the 7th day, a grade of '0' will be awarded even if the assessment is submitted. Submission time for all written assessments is set at **11:55 pm**. A 1-hour grace period is provided to students who experience a technical concern.

For any late submission of time-sensitive tasks, such as scheduled tests/exams, performance assessments/presentations, and/or scheduled practical assessments/labs, students need to submit an application for Special Consideration.

Assessments where Late Submissions will be accepted

In this unit, late submissions will be accepted as follows:

- Practical tasks – NO, unless Special Consideration is granted

- Weekly quizzes – NO, unless Special Consideration is granted

- Major assignments – YES, Standard Late Penalty applies

Special Consideration

If you experience serious and unavoidable difficulties that affect your ability to meet the due dates for progress or the closing date of an assessment task, you may apply for special consideration as explained at https://students.mq.edu.au/study/my-study-program/special-consideration. If the request is accepted, the action may be to grant an extension of the relevant due date(s), or it may be to require you to submit an alternative assessment item.

If you apply for special consideration, please note:

- Apply promptly. Late applications may make it impossible to sensibly offer an extension, and you may risk having to complete a different assessment task which would mean starting from scratch. For example, if you are ill for two days just before the due date, an extension of two days would be reasonable, but that extension cannot be granted more than two days after the due date since the extension end date would have already passed!

- Email the convenor (young.lee@mq.edu.au) and unit lecturer to let us know what is

happening. This will make it easier for us to respond in a timely manner

Assessment Tasks

Name	Weighting	Hurdle	Due
Quizzes	10%	No	Weekly
Final Exam	30%	No	Exam period
Practice-based tasks	20%	Yes	Weekly
Major Assignments	40%	No	Weeks 7 and 13 (TBC)

Quizzes

Assessment Type [1]: Quiz/Test
Indicative Time on Task [2]: 10 hours
Due: **Weekly**
Weighting: **10%**

Quizzes assess students' knowledge and understanding on distributed systems fundamentals, such as architectures, paradigms, principles and models of distributed systems. Specific topics are concurrency, scheduling, virtual machines, containers, serverless computing, microservices and internet of things.

On successful completion you will be able to:
- Describe the complexities of distributed system development and approaches to solve those complexities.
- Explain important issues in modern distributed systems.

Final Exam

Assessment Type [1]: Examination
Indicative Time on Task [2]: 30 hours
Due: **Exam period**
Weighting: **30%**

The final examination will assess your understanding of the unit content and your ability to integrate concepts learned throughout the unit to solve problems.

On successful completion you will be able to:

- Describe the complexities of distributed system development and approaches to solve those complexities.
- Apply theoretical principles and models to design distributed systems.
- Explain important issues in modern distributed systems.
- Configure and deploy cloud services and technologies that support distributed applications.

Practice-based tasks

Assessment Type [1]: Practice-based task
Indicative Time on Task [2]: 20 hours
Due: **Weekly**
Weighting: **20%**
This is a hurdle assessment task (see assessment policy for more information on hurdle assessment tasks)

Practical tasks help guide students to learn practical skills on distributed systems. In particular, they consist of preparatory steps and milestones for assignments. Students will be exposed to many techniques used in practice, such as virtual machines, containers and function as a service. Their configuration and deployment in clouds will also be practiced.

The practical tasks are a **hurdle** in this unit. You must achieve at least 8 marks out of 20 in order to pass the unit.

On successful completion you will be able to:

- Describe the complexities of distributed system development and approaches to solve those complexities.
- Apply theoretical principles and models to design distributed systems.
- Configure and deploy cloud services and technologies that support distributed applications.
- Analyze and design distributed systems.

Major Assignments

Assessment Type [1]: Programming Task
Indicative Time on Task [2]: 40 hours
Due: **Weeks 7 and 13 (TBC)**
Weighting: **40%**

There are two major assignments in which students are required to design and implement parallel and distributed software systems. Specific steps are designed as either milestones/ stages or practical tasks.

On successful completion you will be able to:

- Apply theoretical principles and models to design distributed systems.
- Configure and deploy cloud services and technologies that support distributed applications.
- Analyze and design distributed systems.

---

[1] If you need help with your assignment, please contact:

- the academic teaching staff in your unit for guidance in understanding or completing this type of assessment
- the Writing Centre for academic skills support.

[2] Indicative time-on-task is an estimate of the time required for completion of the assessment task and is subject to individual variation

Delivery and Resources

Classes

Each week you should attend a two-hour online lecture and a two-hour in-person practical. Lectures are a core learning experience where we will discuss the theoretical underpinnings and concepts that are essential to this unit. Key ideas for assessment tasks (technical report and individual distributed systems development project in particular) will be discussed from time to time in lectures. Lecture recordings will be provided in iLearn.

Practicals provide an opportunity for you to ensure your understanding of the key concepts of the unit and develop skills to apply these concepts to practical distributed systems. Practicals combine tutorial-style discussion with practical programming experience, particularly in the later weeks of session. Each week you should start to prepare your solutions to questions for an online quiz.

iLearn Web Site

All learning materials will be published on iLearn including lecture slides and assessment details.

You are required to check the iLearn website at least once a week to ensure that you are aware of the latest materials available there.

Unit Forum

A forum for unit discussions is provided on iLearn. Students are free to post questions, comments or hints in relation to any aspect of the unit, except that you should avoid posting any questions, hints, comments or solutions that could be interpreted as cheating.

Textbooks*

1. "Distributed Systems: Principles and Paradigms" by Maarten van Steen and Andrew Tanenbaum, 3rd (3.01) edition.
2. "Distributed Systems: Concepts and Design" by George Coulouris, Jean Dollimore, and Tim Kindberg, Addison Wesley, 5th edition.
3. "Distributed and Cloud Computing: From Parallel Processing to the Internet of Things" by Geoffrey C. Fox, Jack Dongarra, and Kai Hwang, 1st edition.

* A soft copy of each of these three books is freely available online through publisher's websites.

Methods of Communication

We will communicate with you via your university email or through announcements on iLearn. Queries to the teaching staff members including the unit convenor can either be placed on the iLearn discussion board or sent to corresponding email addresses from your university email address.

COVID Information

For the latest information on the University's response to COVID-19, please refer to the Coronavirus infection page on the Macquarie website: https://www.mq.edu.au/about/coronavirus-faqs. Remember to check this page regularly in case the information and requirements change during semester. If there are any changes to this unit in relation to COVID, these will be communicated via iLearn.

Unit Schedule

The detailed unit schedule will be available on iLearn. The unit is organised into two 6-week periods, with topics approximately as follows.

Week 1-6: Key distributed concepts, such as System models, Architectures, Communications, Synchronisation and Fault tolerance.

Weeks 7-12: Applied distributed computing models and emerging distributed systems, such as virtualisation, cloud computing and the Internet of Things (IoT).

Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central (https://policies.mq.edu.au). Students should be aware of the following policies in particular with regard to Learning and Teaching:

- Academic Appeals Policy
- Academic Integrity Policy
- Academic Progression Policy
- Assessment Policy
- Fitness to Practice Procedure
- Assessment Procedure
- Complaints Resolution Procedure for Students and Members of the Public
- Special Consideration Policy

Students seeking more policy resources can visit Student Policies (https://students.mq.edu.au/support/study/policies). It is your one-stop-shop for the key policies you need to know about throughout your undergraduate student journey.

To find other policies relating to Teaching and Learning, visit Policy Central (https://policies.mq.edu.au) and use the search tool.

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: https://students.mq.edu.au/admin/other-resources/student-conduct

Results

Results published on platform other than eStudent, (eg. iLearn, Coursera etc.) or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in eStudent. For more information visit ask.mq.edu.au or if you are a Global MBA student contact globalmba.support@mq.edu.au

Academic Integrity

At Macquarie, we believe academic integrity – honesty, respect, trust, responsibility, fairness and courage – is at the core of learning, teaching and research. We recognise that meeting the expectations required to complete your assessments can be challenging. So, we offer you a range of resources and services to help you reach your potential, including free online writing and maths support, academic skills development and wellbeing consultations.

Student Support

Macquarie University provides a range of support services for students. For details, visit http://students.mq.edu.au/support/

The Writing Centre

The Writing Centre provides resources to develop your English language proficiency, academic writing, and communication skills.

- Workshops
- Chat with a WriteWISE peer writing leader

- Access StudyWISE
- Upload an assignment to Studiosity
- Complete the Academic Integrity Module

The Library provides online and face to face support to help you find and use relevant information resources.

- Subject and Research Guides
- Ask a Librarian

Student Services and Support

Macquarie University offers a range of Student Support Services including:

- IT Support
- Accessibility and disability support with study
- Mental health support
- Safety support to respond to bullying, harassment, sexual harassment and sexual assault
- Social support including information about finances, tenancy and legal issues
- Student Advocacy provides independent advice on MQ policies, procedures, and processes

Student Enquiries

Got a question? Ask us via AskMQ, or contact Service Connect.

IT Help

For help with University computer systems and technology, visit http://www.mq.edu.au/about_us/offices_and_units/information_technology/help/.

When using the University's IT, you must adhere to the Acceptable Use of IT Resources Policy. The policy applies to all who connect to the MQ network including students.

Changes from Previous Offering

We greatly appreciate student feedback as it enables us to continuously enhance our course offerings. In response to feedback from previous sessions, we've simplified the programming assignment and added a project report assignment. These assignments are designed to complement the weekly tutor class practices, during which tutors will offer detailed and constructive feedback to assist students with their work. Furthermore, we've implemented adaptive teaching methods to tailor instruction and improve student comprehension of essential concepts, guided by their input during practical classes.

Changes since First Published

Date	Description
16/02/2024	Add roles and changes since last offering

Unit information based on version 2024.02 of the Handbook