BITS Pilani

BITS Pilani
Hyderabad Campus

Dr. Manik Gupta
Associate Professor
Department of CSIS

innovate achieve lead

BITS Pilani
Hyderabad Campus

Distributed Data Systems (CS G554)
Lecture 1-2

Tuesday, 6th August 2024

Today's agenda

- Introduction
- Course Objectives
- Course Structure
- Reading Materials

- Introduction to Distributed Systems

About myself

Dr. Manik Gupta

Email:

manik@hyderabad.bits-pilani.ac.in

Google Classroom:

Class code **qc34tyx**

https://classroom.google.com/c/NzAyMjc0MTM1Mzg0?cjc=qc34tyx

About myself

innovate achieve lead

- 20+ years in computer science industry & research
- 13+ years of research and academic experience
  - Postdoc (UCL, LSBU)
  - PhD (Data analysis, WSN, IoT) from QMUL (2014)
  - M.Tech.(CS) and B.E.(CS) from IITD and PEC
- Seven years work experience in telecom embedded software industry

About yourself

- Tell me more about yourself, your background and interests?
- Why have you decided to take this module?

- Fill the ***pre assessment form***
  - https://forms.gle/azevuMDPEciyHWQF7

Course scope

- With the ever-growing **pervasive data** and the subsequent increasing computational requirements, **distributed systems** are becoming more and more widespread.
- They are a vast and complex field of study in computer science. In this course, the focus is particularly upon **distributed data systems -** network of interconnected computers that work together to **store, manage, and process data**.
- Distributed data systems are essential for modern applications that require **high scalability, reliability, and performance.**

Course scope

- The course aims at familiarizing the students with the concepts of Distributed Data Systems which includes topics like
  - **Distributed Databases**
  - **Distributed File Systems**
  - **NoSQL Databases**
  - **Data on the Web with Web serving as a distributed data repository**

There is a lot to learn and explore!!

Course objectives

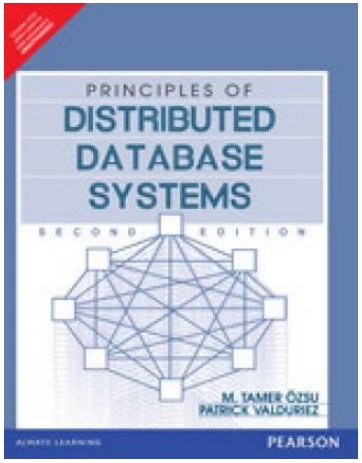
- To gain an understanding of how data distribution is planned, designed and implemented for **distributed databases**.
- To understand challenges in distributed database query processing and optimization, transaction processing and concurrency control.
- To be able to understand the working of **distributed file systems**.
- To gain knowledge in design and implementation of **NoSQL databases.**
- To gain knowledge about new paradigms of **web data systems**.
- To gain hands on experience in both practical as well as design aspects of distributed data systems.

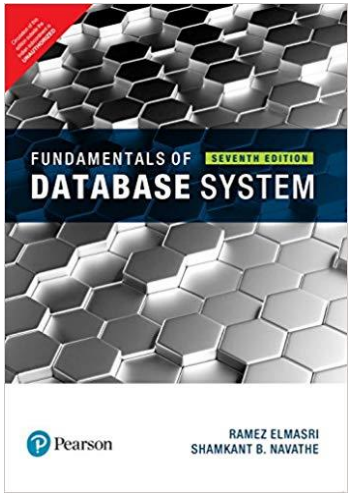
Course structure

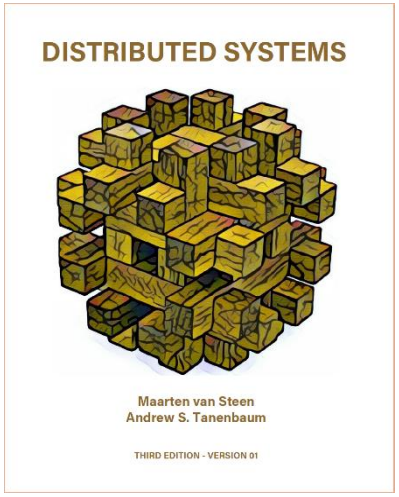
- There will be around 12-14 weeks of teaching time
  - 38-42 lectures (3 hrs per week)
  - 12-14 labs (2 hrs per week)
- Course Evaluation Scheme
  - How your learning will be measured?

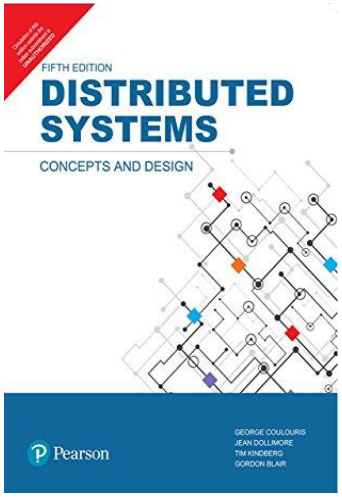
Component	Duration	Weightage (%)	Date & Time	Nature of Component
Mid semester exam	1.5 Hrs	25%	As per the timetable	Closed Book
Project	-	30%	To be Announced	Open Book
Paper Presentations	-	10%	To be Announced	Open Book
Comprehensive exam	3 Hrs	35%	As per the timetable	Closed Book

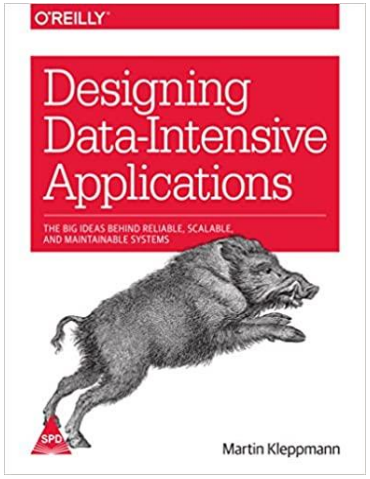
Reading materials









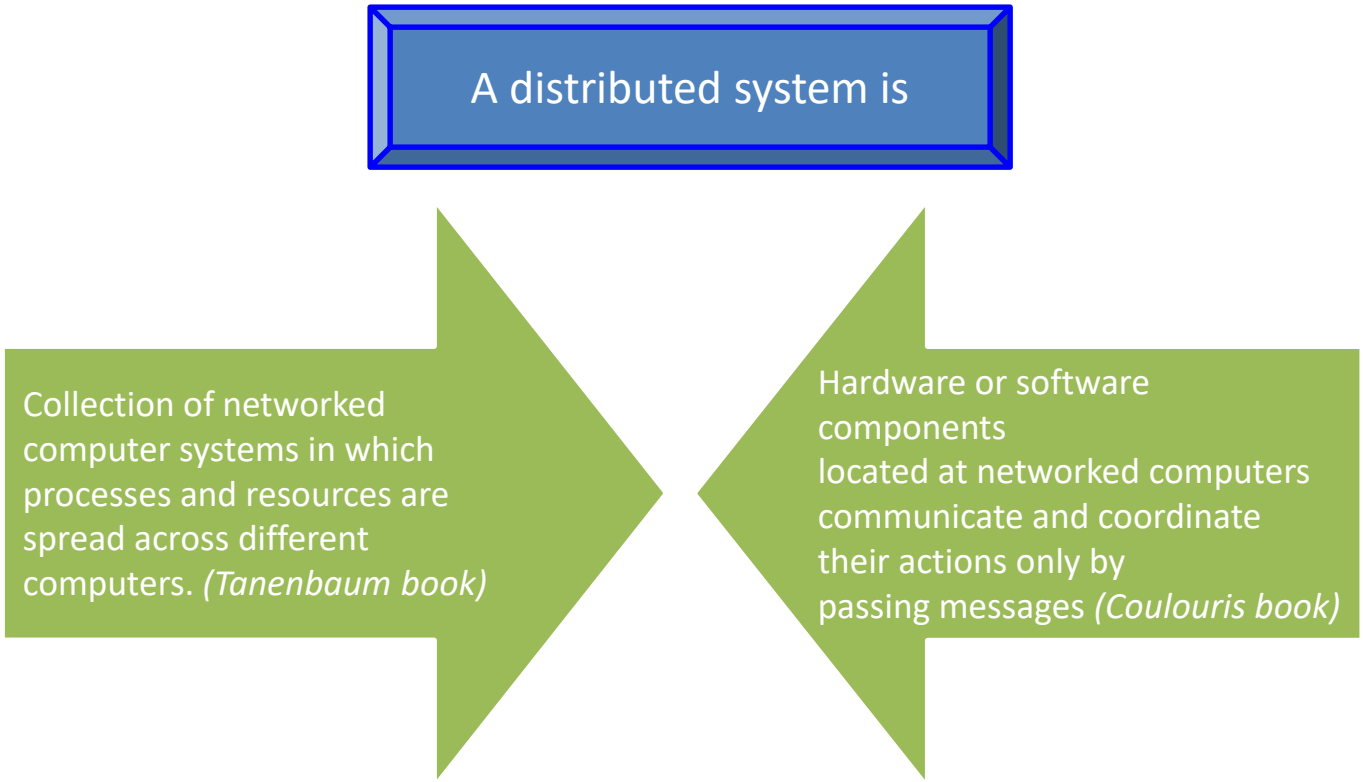


Few ground rules…

- Be ***regular*** and ***attend*** the classes
- ***Submit*** homework, exercises, assignments on time
  - **Pre readings and exercises will be issued on google classroom**
- ***Check*** your emails, google classroom regularly
- ***Be involved*** and make the best utilization of the resources and reading materials – ***Lots of reading*** involved..

Have fun and enjoy the learning journey!!

Definition of a distributed system



What are distributed systems?

Characteristics of distributed systems

- Collection of autonomous computing elements, also referred to as nodes, can be hardware devices or software processes.
- Single coherent system: users or applications perceive a single system and nodes need to collaborate.

Collection of autonomous nodes

- Dealing with independent nodes
  - Each node is autonomous and can act independently from each other. Thus each node has its own notion of time: there is no global clock. Leads to fundamental synchronization and coordination problems.
- Dealing with a collection of nodes
  - Manage the membership and organization of collection of nodes
    - How to manage group membership?
    - How to know that you are indeed communicating with an authorized (non)member?
    - Organise as an overlay network

Single Coherent system

- End users should not notice that they are dealing with processes, data and control are dispersed across a computer network
  - An end user cannot tell where a **computation** is taking place
  - Where **data** is exactly stored should be irrelevant to an application If or not data has been replicated is completely hidden
  - Keyword is **distribution transparency** and is an important design goal!
- It is inevitable that at any time only a part of the distributed system fails owing to multiple, networked nodes
  - Hiding **partial failures** and their recovery is often very difficult and in general impossible to hide.

Design goals of distributed systems

- Supporting resource sharing
- Making distribution transparent
- Openness
- Scalability

Supporting resource sharing

Distributed systems should make it easy for users and applications to **access and share remote resources**.

Why would you want to share resources?

- Economics
  - Cheaper to have a single high end reliable storage facility
- Easier to collaborate and exchange information
  - Exchanging files, mail, documents, audio and video
  - Software for collaborative editing, teleconferencing

Distribution transparency

- A distributed system that is able to present itself to users and applications as if it were only a **single computer system** is said to be **transparent**.

Transparency	Description
Access	Hide differences in data representation and how an object is accessed
Location	Hide where an object is located
Relocation	Hide that an object may be moved to another location while in use
Migration	Hide that an object may move to another location
Replication	Hide that an object is replicated
Concurrency	Hide that an object may be shared by several independent users
Failure	Hide the failure and recovery of an object

Distribution transparency

- Can you think of an example how location transparency is achieved?
  - Use of ***URLs*** give no clue about actual location of the server
- What is the difference between relocation and migration transparency? Can you think of examples?
  - Entire site being moved from one data centre to another
  - Communication between mobile phones
- Can you think of an example where concurrency transparency will be needed?
  - Accessing same tables in a shared database

Openness of distributed systems

Be able to interact with services from other open systems, irrespective of the underlying environment

- Systems should conform to well-defined interfaces
- Systems should easily interoperate
- Systems should support portability of applications
- Systems should be easily extensible

Openness

- Interfaces
  - Components should adhere to standard rules that describe the syntax and semantics that the components need to offer
- Interoperability
  - Extent to which two implementations of systems or components from different manufacturers can co-exist and work together
- Portability
  - To what extent an application developed for system A can be executed without modification on another system B that implements same interfaces as A
- Extensibility
  - Easy to add new components or replace existing ones without affecting components that stay in place

Scale in distributed systems

Scalability can be measured along 3 different dimensions

- Add more number of users and/or processes (size scalability)
- Maximize distance between nodes, but communication delays are hardly noticed (geographical scalability)
- Number of administrative domains governed by policy (administrative scalability)

- Can you think of scaling techniques?

Discussion – Distributed System Examples

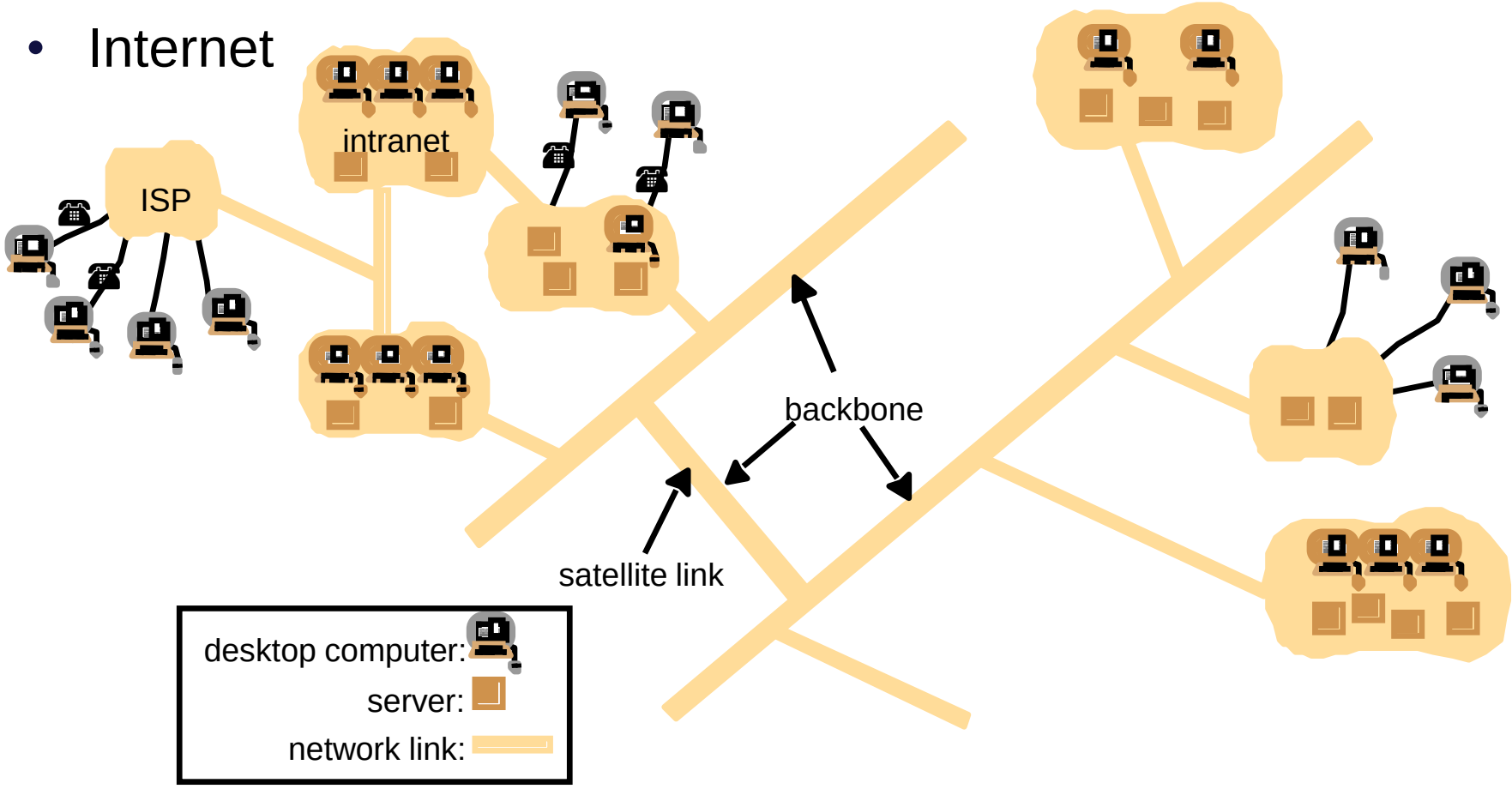
https://www.menti.com/aliehqia34rj

Application domains and corresponding networked applications

Finance and commerce	eCommerce e.g. Amazon and eBay, PayPal, online banking and trading
The information society	Web information and search engines, ebooks, Wikipedia; social networking: Facebook and MySpace
Creative industries and entertainment	Online gaming, music and film in the home, user-generated content, e.g. YouTube, Flickr
Healthcare	Health informatics, online patient records, remote monitoring of patients
Education	e-learning, virtual learning environments; distance learning
Transport and logistics	GPS in route finding systems, map services: Google Maps, Google Earth
Science	Grid as an enabling technology for collaboration between scientists
Environmental management	Sensor technology to monitor earthquakes, floods or tsunamis

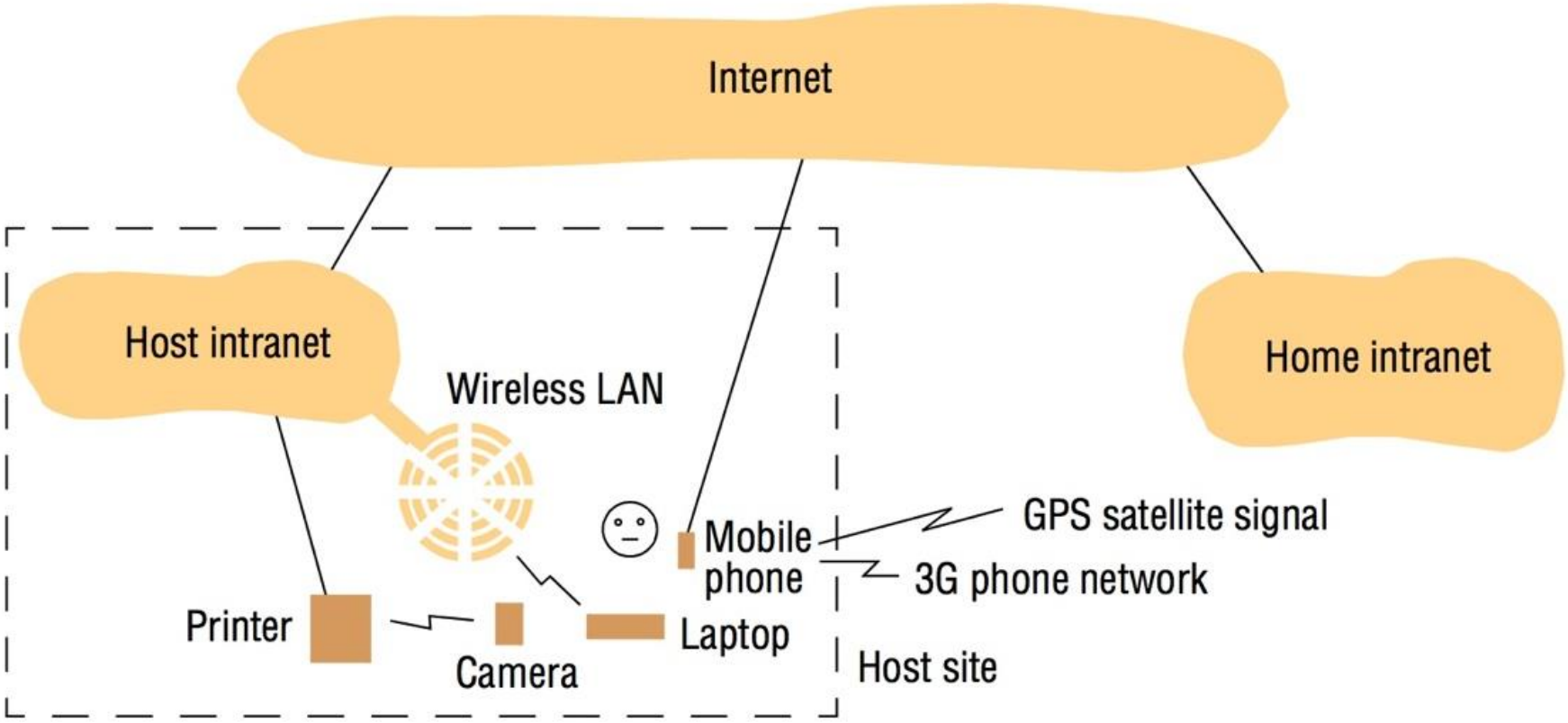
Distributed system: Example 1

- Internet



Distributed system: Example 2

- Mobile and ubiquitous computing
  - Portable and handheld wirelessly connected computing devices



Distributed system: Example 3

Distributed system infrastructure to support Google web search

- an underlying ***physical infrastructure*** consisting of very large numbers of networked computers located at data centers all around the world;
- a ***distributed file system*** designed to support very large files and heavily optimized for the style of usage required by search and other Google applications;
- an associated structured ***distributed storage system*** that offers fast access to very large datasets;
- a ***lock service*** that offers distributed system functions such as distributed locking and agreement;
- a ***programming model*** that supports the management of very large parallel and distributed computations across the underlying physical infrastructure.

Lecture summary

innovate achieve lead

Topics Covered

- What is a distributed system?
- What are characteristics, design goals of distributed systems?
- Examples of distributed systems

Essential Readings

- Chapter 1: Tanenbaum
- Chapter 1: Coulouris

Thanks…

Next Lecture

- Distributed system architectures

Questions??