

# CSIE52400/CSIEM0140 Distributed Systems

Shiow-yang Wu (吳秀陽)  
Department of Computer Science  
and Information Engineering  
National Dong Hwa University

CSIE52400/CSIEM0140 Distributed Systems

1



## Happy Lantern Festival (元宵節)

- Lunar New Year Holiday is not over until the **Lantern Festival** !



CSIE52400/CSIEM0140 Distributed Systems

2

## What is a distributed system?

- **Definition:**

– A distributed system is a collection of **independent computers** and related **software** that **appears** to its users **as a single coherent system**.

– **Hardware** or **software** components in **networked computers** communicate and coordinate by passing **messages**.

- **Motivation?**

– **Sharing** of resources, information and services

– Improving **availability**, **reliability**, **fault tolerance**, **performance** and **scalability**

- **Consequences:**

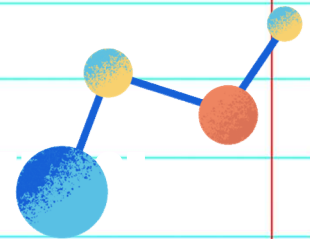
– **Concurrency** ⇒ How to communicate and coordinate?

– **Delay** ⇒ How to cope with network transmission delay?

– **No global clock** ⇒ How to synchronize?

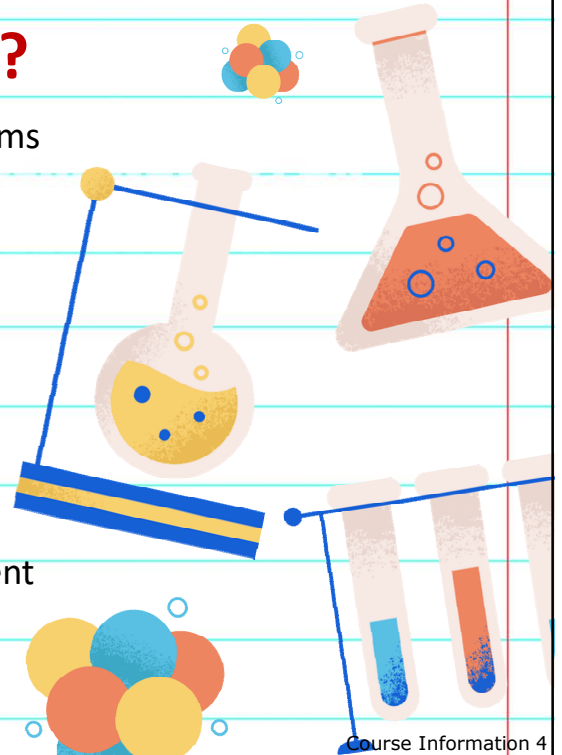
– **Independent failures** ⇒ How to achieve fault tolerance?

– ... (Assignment 0)



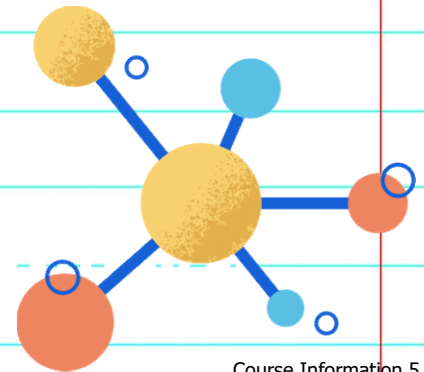
## What is this course about?

- Architectures and models of distributed systems
- Distributed processes, virtualization
- Communication
- Synchronization & coordination
- Addressing & naming
- Replication & consistency
- Distributed storage & file systems
- Distributed algorithms & computation
- Distributed programming and app development
- Fault tolerance
- Security



## Advanced Topics\*

- Service computing(Web services, SOA, microservices)
- Mobile and pervasive computing
- Grid, cloud, fog and edge computing
- Big data systems
- P2P(Peer-to-peer) systems
- Wireless sensor networks(WSN)
- Internet of Things(IoT)
- Distributed data stream processing
- Crowd computing
- Social networks and computing



## Semester Theme: AIoT

- A **main theme** is picked for each semester.
- The papers for your **independent study** and **presentation** should be selected under the theme.
- Theme for this semester: **Artificial Intelligence of Things (AIoT)**
- Combining **IoT** with **AI** to get **Edge Intelligence (Edge AI)**.
- AIoT market size is expected to grow exponentially (from **\$7.52 billion** in 2023 to **\$9.98 billion** in 2024).
- More about this in later lectures and your presentation.

## Course Information

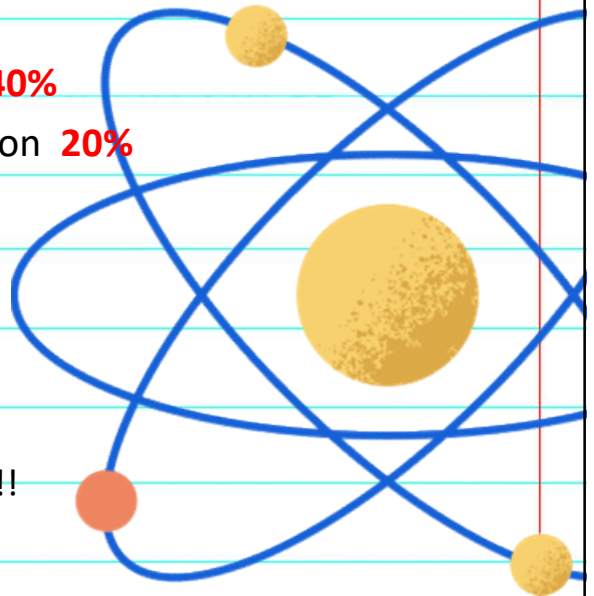
- **Time:** Tue 9:10~12:00
- **Class homepage:**  
<http://web.csie.ndhu.edu.tw/showyang/DistrSys2024s/index.html>
- **Instructor's homepage:**  
<http://web.csie.ndhu.edu.tw/~showyang>
- **Office Hours:** Tue 17:00-18:00
- **Office:** Eng Building C308 (理工二館 C308)
- **Phone:** (03)890-3020
- **E-mail:** [showyang@gms.ndhu.edu.tw](mailto:showyang@gms.ndhu.edu.tw)

## Online Course Links

- **Online Course Teams Link:**  
[https://teams.microsoft.com/l/team/19%3aNK6KBeeGD2Nz\\_tWglHy4FLpKOYGZBK4JzgHhswSS\\_Jo1%40thread.tacv2/conversations?groupId=40213531-a422-491c-a5c1-b8b3d71f72f6&tenantId=edba3211-8174-4411-b089-357c588fa127](https://teams.microsoft.com/l/team/19%3aNK6KBeeGD2Nz_tWglHy4FLpKOYGZBK4JzgHhswSS_Jo1%40thread.tacv2/conversations?groupId=40213531-a422-491c-a5c1-b8b3d71f72f6&tenantId=edba3211-8174-4411-b089-357c588fa127)
- **Join Code:** **z18wcz5**
- **Online Course Link:** [https://teams.microsoft.com/l/meetup-join/19%3aNK6KBeeGD2Nz\\_tWglHy4FLpKOYGZBK4JzgHhswSS\\_Jo1%40thread.tacv2/1621231206604?context=%7b%22Tid%22%3a%22edba3211-8174-4411-b089-357c588fa127%22%2c%22Oid%22%3a%22e83708da-2e73-4b78-a037-e2bbca1f4d94%22%7d](https://teams.microsoft.com/l/meetup-join/19%3aNK6KBeeGD2Nz_tWglHy4FLpKOYGZBK4JzgHhswSS_Jo1%40thread.tacv2/1621231206604?context=%7b%22Tid%22%3a%22edba3211-8174-4411-b089-357c588fa127%22%2c%22Oid%22%3a%22e83708da-2e73-4b78-a037-e2bbca1f4d94%22%7d)

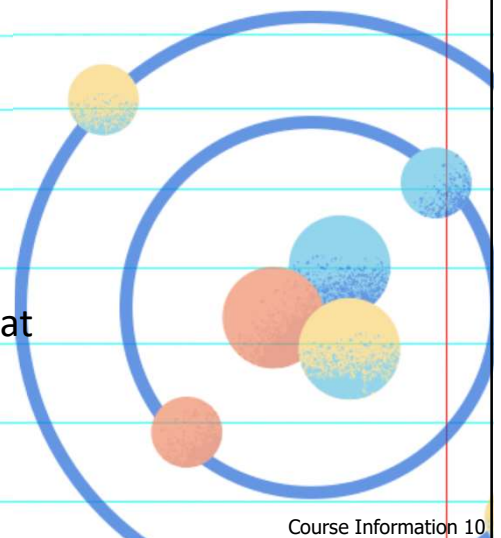
## Grading Policy

- Homework/programming assignments **40%**
- Independent study and paper presentation **20%**
- Final exam **20%**
- Term project **20%**
  - Project and demonstration
  - Report
  - Due **one** week **before** the final exam.
  - Must** do the term project to pass the class!!
- May change if necessary.



## Assignments

- There will be several homework/programming assignments.
- Homework is your **food for thought**.
- Programming assignments are for you to get **hands-on experience**.
- We will use **open-source tools**. (more on this later)
- For term project, you may choose any tool that suits your target domain.



## Independent Study & Presentation

- **Theme:** **Artificial Intelligence of Things (AIoT, 人工智慧物聯網)**
- Select your **topic** and discuss with the instructor.
- Do a **literal search** for candidate **papers**.
- Find at least **three candidate papers** to the instructor no later than **one week before midterm week** to finalize the topic and papers.
- The presentation **schedule** will be finalized during the midterm week.
- Your presentation should cover the **essentials** of the selected topic and one **in-depth** technical case study.

## Term Project

- Each student is required to conduct a **semester long** term project on a selected topic.
- The topic can be **selected jointly** so all students work on the same topic.
- It can also be **individual project**.
- The decision will be announced before midterm.
- Sample projects can be found on the project page.
- Demo & due date: **June 3~7, 2024** (1 week **before** final)
- Must prepare a **project report** at the time of **demo**.

## Project Demonstration and Reports

- You should turn in the following items and prepare for a demonstration to the instructor.
  - **Project proposal** (Due date: 1 week **after** midterm week)
  - **Project demonstration** (Schedule the time with the instructor at any time during the demo week.)
  - **Project report** (Due date: at the time of demonstration)

## Main References

- Maarten Van Steen and Andrew S. Tanenbaum. ***Distributed Systems, 4th Edition***. 2023. (<https://www.distributed-systems.net/index.php/books/ds4/>)
- Roberto Vitillo. *Understanding Distributed Systems: What every developer should know about large distributed applications*. Roberto Vitillo, 2021. (<https://leanpub.com/understanding-distributed-systems>)

## References: Distributed Systems

- George Coulouris, Jean Dollimore and Tim Kindberg. *Distributed Systems: Concepts and Design, 5th Edition*. Addison-Wesley, 2012.
- Sukumar Ghosh. *Distributed Systems: An Algorithmic Approach, 2nd Edition*. CRC Press, 2014.
- Fourre Sigs. *Distributed Algorithms: A Verbose Tour*, Independently published, 2019.
- Brendan Burns. *Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services*, O'Reilly Media, 2018.
- Ajay D. Kshemkalyani and Mukesh Singhal. *Distributed Computing: Principles, Algorithms, and Systems*. Cambridge University Press, 2008, 2012(online).

## References: Distributed Cloud

- Hong Lin and Weiqi Tian, *Distributed Cloud: Reference Architecture Design*, Independently published, 2023.
- Theo Lynn, John G. Mooney, Brian Lee and Patricia Takako Endo(Eds.). *The Cloud-to-Thing Continuum: Opportunities and Challenges in Cloud, Fog and Edge Computing*, Palgrave Macmillan, 2020.
- Theo Lynn, John G. Mooney, Jorg Domaschka and Keith A. Ellis(Eds.). *Managing Distributed Cloud Applications and Infrastructure: A Self-Optimising Approach*, Palgrave Macmillan, 2020.
- Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra. *Distributed and Cloud Computing: From Parallel Processing to the Internet of Things*, Morgan Kaufmann, 2012.



## References: Edge Computing

- K. Anitha Kumari, G. Sudha Sadasivam and D. Dharani, M. Niranjanamurthy. *Edge Computing: Fundamentals, Advances and Applications*. CRC Press, 2021.
- Javid Taheri and Shuiguang Deng. *Edge Computing: Models, Technologies and Applications*. Institution of Engineering & Technology, 2020.
- Taheri, Javid, et al. *Edge Intelligence: From Theory to Practice*. Springer International Publishing, 2023.
- Wang, Xiaofei, et al. *Edge AI: Convergence of Edge Computing and Artificial Intelligence*. Springer Nature Singapore, 2020.
- Wang, Dong. And Zhang, Daniel Yue. *Social Edge Computing: Empowering Human-Centric Edge Computing, Learning and Intelligence*. Springer International Publishing, 2023.

## References: IoT

- Rajiv Ranjan, Karan Mitra, Prem Prakash Jayaraman, Albert Y. Zomaya (Eds.). *Managing Internet of Things Applications Across Edge and Cloud Data Centres - Computing and Networks*. The Institution of Engineering and Technology, 2024.
- Brojo Kishore Mishra and Amit Vishwasrao Salunkhe (Eds.). *Internet of Things - Technological Advances and New Applications*. Apple Academic Press, 2023.
- F. John Dian. *Fundamentals of Internet of Things: For Students and Professionals*. Wiley-IEEE Press, 2022.
- Ammar Rayes and Samer Salam. *Internet of Things from Hype to Reality: The Road to Digitization, 3<sup>rd</sup> ed.* Springer, 2022.
- Sandeep Saxena and Ashok Kumar Pradhan (Eds.). *Internet of Things: Security and Privacy in Cyberspace*. Springer, 2022.
- Sachi Nandan Mohanty, Jyotir Moy Chatterjee and Suneeta Satpathy (Eds.). *Internet of Things and Its Applications*. Springer, 2021.
- Farshad Firouzi, Krishnendu Chakrabarty and Sani Nassif (Eds.) *Intelligent Internet of Things: From Device to Fog and Cloud*. Springer International Publishing, 2020.

## Spark Books

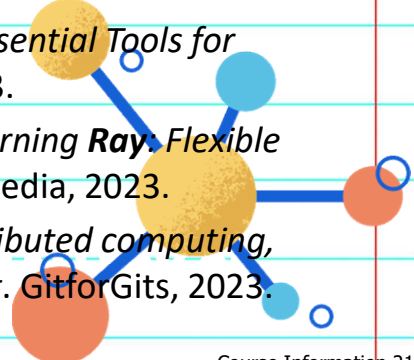
- Jules S. Damji, Brooke Wenig, Tathagata Das, and Denny Lee. *Learning Spark: Lightning-Fast Data Analytics, 2<sup>nd</sup> Edition*, O'Reilly Media, 2020.
- Mahmoud Parsian. *Data Algorithms with Spark: Recipes and Design Patterns for Scaling Up using PySpark*. O'Reilly Media, 2022.
- Adi Polak. *Scaling Machine Learning with Spark: Distributed ML with MLLib, TensorFlow, and PyTorch*. O'Reilly Media, 2023.
- Wenqiang Feng. *Learning Apache Spark with Python*. 2021. (free online and pdf)(<https://runawayhorse001.github.io/LearningApacheSpark/>)
- Jacek Laskowski. *The Internals of Spark Core*. 2024. (online book)(<https://books.japila.pl/apache-spark-internals/>)
- Cybellium Ltd and Kris Hermans. *Mastering Apache Spark: A Comprehensive Guide to Learn Apache Spark*. Independently published, 2023.
- Akash Tandon, Sandy Ryza, Uri Laserson, Sean Owen and Josh Wills. *Advanced Analytics with PySpark: Patterns for Learning from Data at Scale Using Python and Spark*, O'Reilly Media, 2022.

## Python Programming Books

- Eric Matthes. *Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 3rd Edition*. No Starch Press, 2023.
- Steve Holden, Anna Ravenscroft and Alex Martelli. *Python in a Nutshell: A Desktop Quick Reference, 4th Edition*. O'Reilly Media, 2023.
- Johannes Ernesti and Peter Kaiser. *Python 3: The Comprehensive Guide to Hands-On Python Programming*. Rheinwerk Computing, 2022.
- Brett Slatkin. *Effective Python: 135 Specific Ways to Write Better Python, 3rd Edition*. Addison-Wesley Professional, 2024.
- Luciano Ramalho. *Fluent Python: Clear, Concise, and Effective Programming, 2<sup>nd</sup> Edition*. O'Reilly Media, 2022.
- Wes McKinney. *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter, 3rd edition*. O'Reilly Media, 2022.
- A Byte of Python (free online book)(<https://python.swaroopch.com/>)

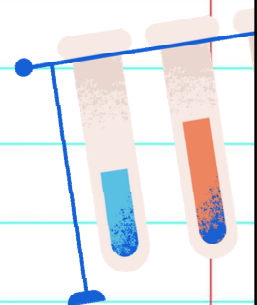
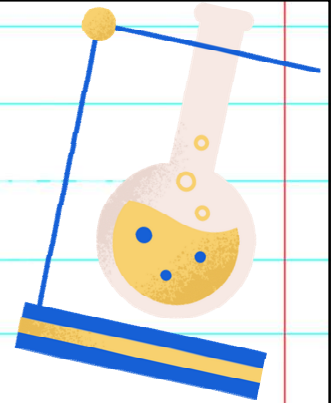


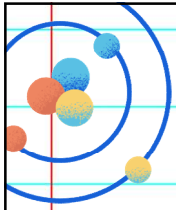
## Parallel/Distributed Data with Python

- Fabio Nelli. *Parallel and High Performance Programming with Python: Unlock parallel and concurrent programming in Python using multithreading, CUDA, Pytorch and Dask*, AVA, 2023.
  - Yuli Vasiliev. *Python for Data Science: A Hands-On Introduction*. No Starch Press, 2022.
  - Jake VanderPlas. *Python Data Science Handbook: Essential Tools for Working with Data, 2<sup>nd</sup> Edition*. O'Reilly Media, 2023.
  - Max Pumperla, Edward Oakes and Richard Liaw. *Learning Ray. Flexible Distributed Python for Machine Learning*. O'Reilly Media, 2023.
  - Tim Peters. *Parallel Python with Dask: Perform distributed computing, concurrent programming and manage large dataset*. GitforGits, 2023.
- 

## Lecture Topics

- **Introduction (1 week)**
  - Introduction to distributed systems
  - Networking and internetworking essentials
  - Design goals
  - Classification of distributed systems
  - Examples of modern distributed systems
  - Current and future trends

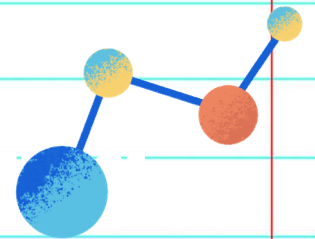




## Lecture Topics (contd.)

- **Architectures and Middlewares (1 week)**

- Architectural styles
- System models and architectures
- Middlewares
- Layered-system architectures
- Symmetrically distributed system architectures
- Hybrid system architectures
- Self-management



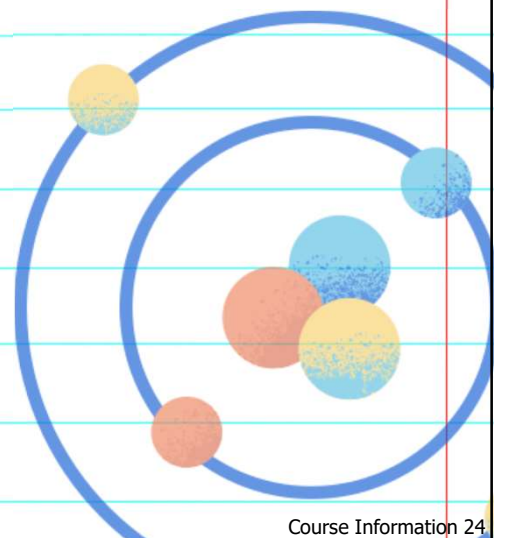
## Lecture Topics (contd.)

- **Distributed Processes (2 weeks)**

- Processes and threads (Python threads)
- Clients and servers
- Operating system support
- Code mobility and agents
- Virtualization
- Microservices

- **Communication (2 weeks)**

- Interprocess communication models
- Remote invocation (RPC, RMI)
- Message-oriented communication
- Multicast and group communication



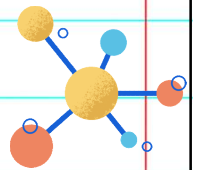
## Lecture Topics (contd.)

- **Addressing and Name Services (1 week)**

- Fundamentals
- Flat vs. structured naming
- Attribute-based naming
- Directory services

- **Time, Synchronization & Coordination (2 weeks)**

- Time and clock synchronization
- Logical clocks
- Global state and snapshot
- Mutual exclusion
- Location systems
- Gossip-based coordination



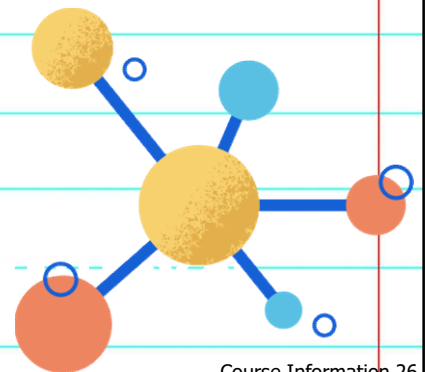
## Lecture Topics (contd.)

- **Distributed Algorithms & Computation (2 weeks)**

- Election
- Consensus
- Distributed event processing
- Distributed graph algorithms
- MapReduce
- BSP(Bulk Synchronous Parallel)

- **Distributed Programming (1 week)**

- Python distributed computing with Ray
- Python for Spark programming with PySpark
- IoT with Python



## Lecture Topics (contd.)

- **Distributed Storage & Data Management\* (1 week)**
  - Distributed storage
  - File service architecture
  - Network File System
  - Mobile File System
  - Distributed database systems
- **Consistency and Replication\* (1 week)**
  - Consistency models
  - Replica management
  - Consistency protocols

## Lecture Topics (contd.)

- **Fault Tolerance\* (1 week)**
  - Failure Models and Process Resilience
  - Reliable Communication
- **Security\* (1 week)**
  - Security Models
  - Access Control
  - Security Management
- **Distributed Transactions\* (1 week)**
  - Transactions and concurrency control
  - Distributed commit
  - Distributed transaction management

## Lecture Topics (contd.)

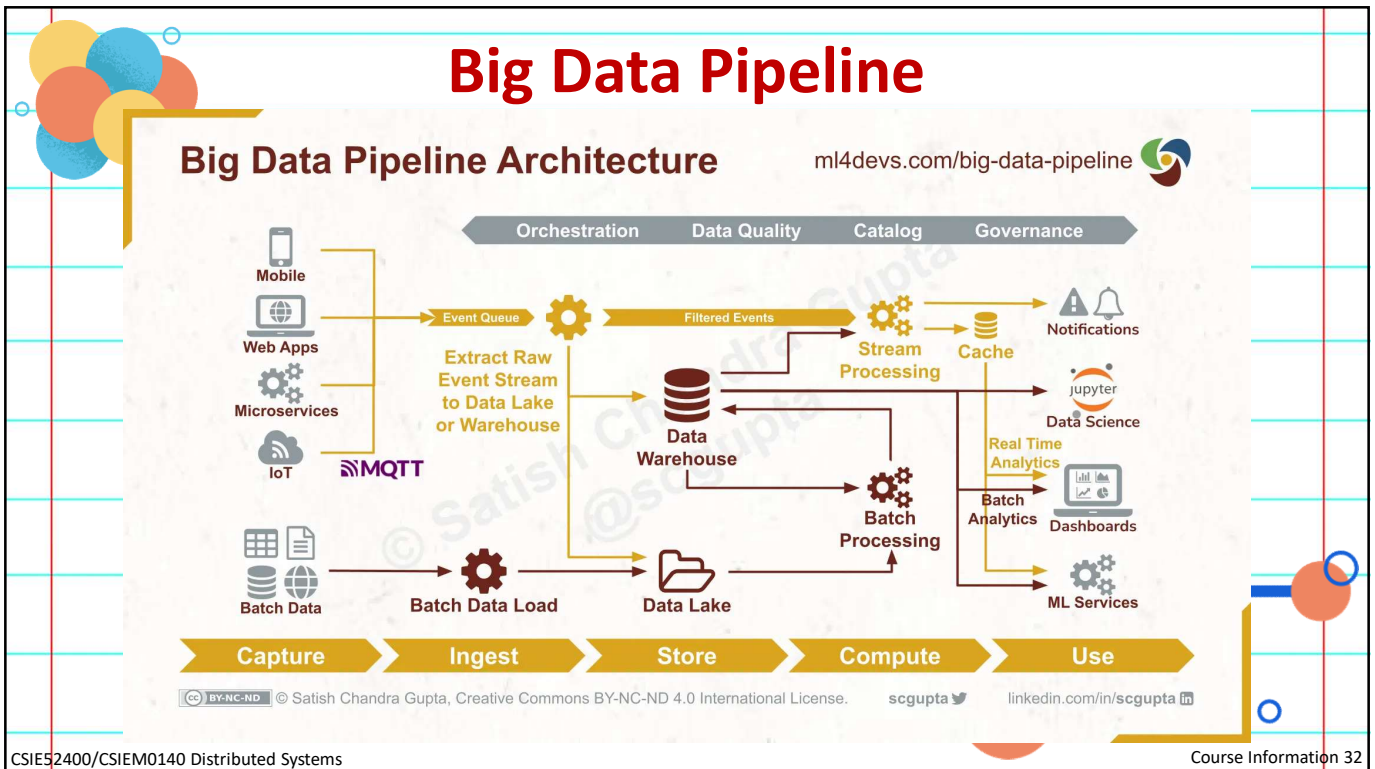
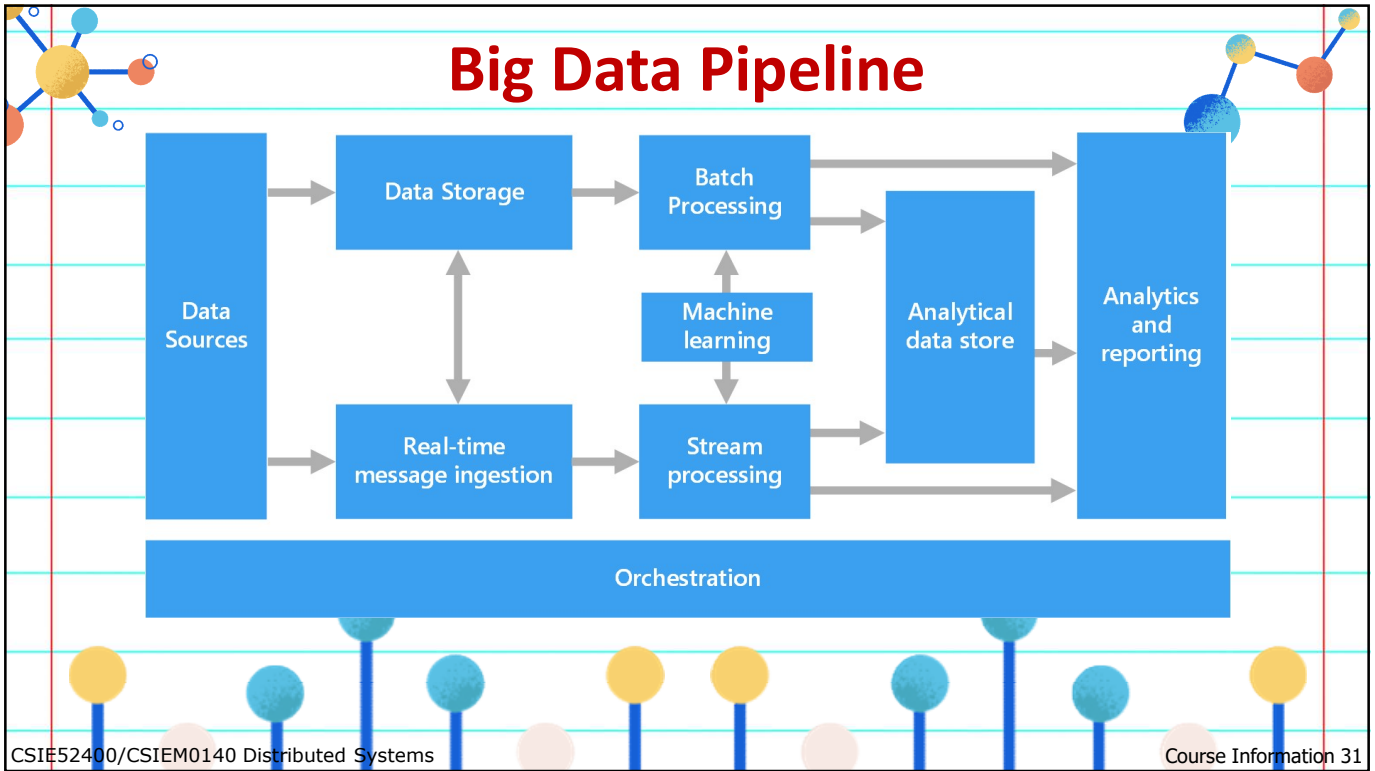
- **Advanced Topics\* (1 week)**

- Web services
- Mobile and pervasive computing
- Grid, cloud, fog and edge Computing
- P2P(Peer-to-peer) systems
- Wireless sensor networks(WSN) and Internet of Things(IoT)
- Crowd computing
- Mobile sensing
- Social networks and computing

## Main Theme: AIoT

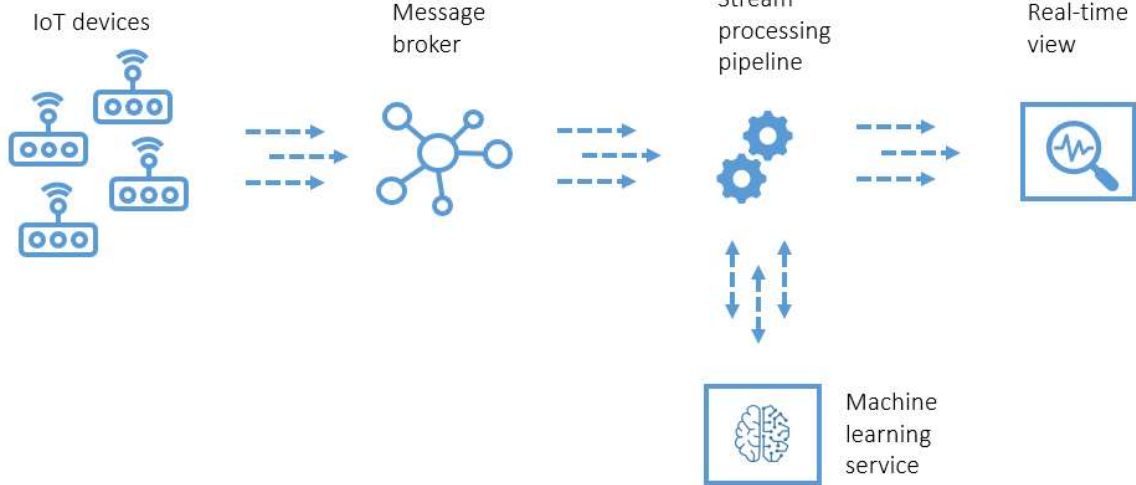
- **Theme: Artificial Intelligence of Things (AIoT)**

- Combination of AI and IoT
- Definitions, challenges and opportunities
- Benefits of AI and IoT integration
- AIoT for Edge AI or Edge Intelligence
- On-device machine learning with AIoT devices
- Edge-to-Cloud AI architecture
- AIoT platforms and tools
- AIoT applications



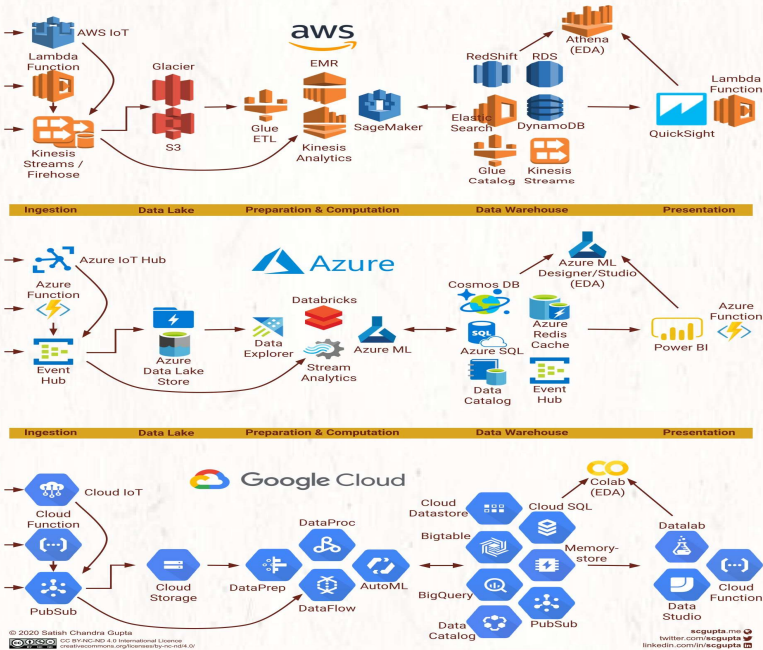


# IoT Streaming Data Pipeline



# Data Pipelines: AWS, Azure, GCP

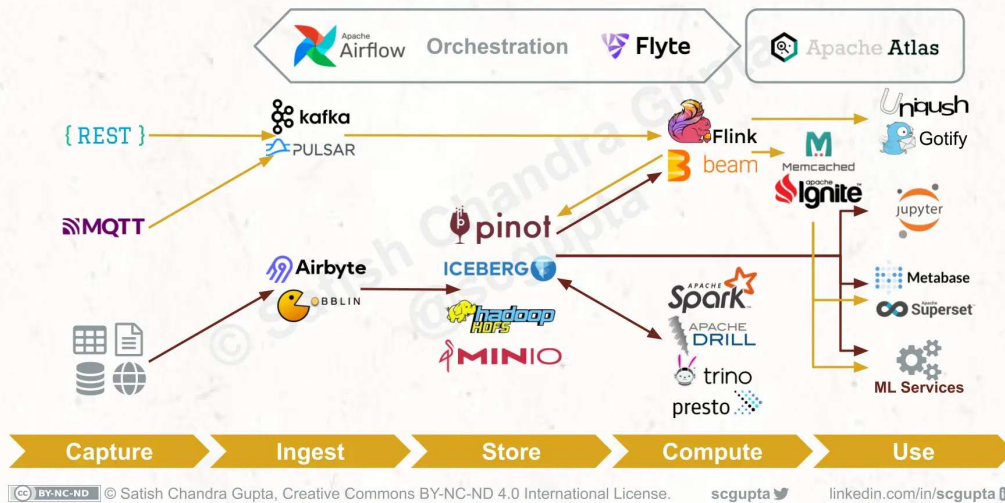
## Big Data Pipelines on AWS, Microsoft Azure, and GCP



# Open Source Data Pipeline

## Data Pipeline – Open Source Stack

ml4devs.com/big-data-pipeline




## Why Python ?

- Python is **easy** to learn and easy to use.
- Python is **versatile**(multi paradigm).
- Python is more **productive** !!
- Python has amazing **libraries**.
- Python has a healthy, active and supportive **community**.
- Python has some great corporate **sponsors**.
- Python is popular in **data science**.
- Python is **reliable** and **efficient**.
- Python is **accessible**.
- With Python, there really are no limits!



# TIOBE & PYPL Index

## TIOBE Index

Feb 2024	Feb 2023	Change	Programming Language	Ratings	Change
1	1		 Python	15.16%	-0.32%
2	2		 C	10.97%	-4.41%
3	3		 C++	10.53%	-3.40%
4	4		 Java	8.88%	-4.33%
5	5		 C#	7.53%	+1.15%
6	7	▲	 JavaScript	3.17%	+0.64%
7	8	▲	 SQL	1.82%	-0.30%
8	11	▲	 Go	1.73%	+0.61%
9	6	▼	 Visual Basic	1.52%	-2.62%
10	10		 PHP	1.51%	+0.21%

## PYPL Index

Worldwide, Feb 2024 :

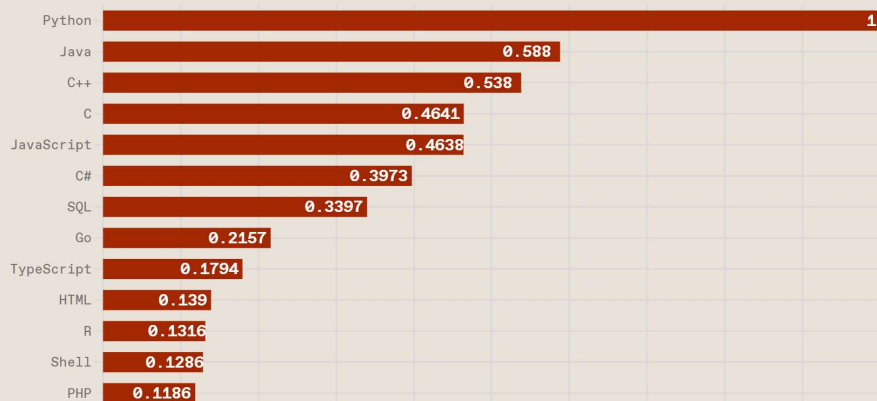
Rank	Change	Language	Share	1-year trend
1		Python	28.11 %	+0.6 %
2		Java	15.52 %	-1.0 %
3		JavaScript	8.57 %	-1.0 %
4	▲	C/C++	6.92 %	+0.1 %
5	▼	C#	6.73 %	-0.1 %
6	▲	R	4.75 %	+0.7 %
7	▼	PHP	4.57 %	-0.6 %
8		TypeScript	2.78 %	-0.0 %
9		Swift	2.75 %	+0.5 %
10		Objective-C	2.37 %	+0.1 %

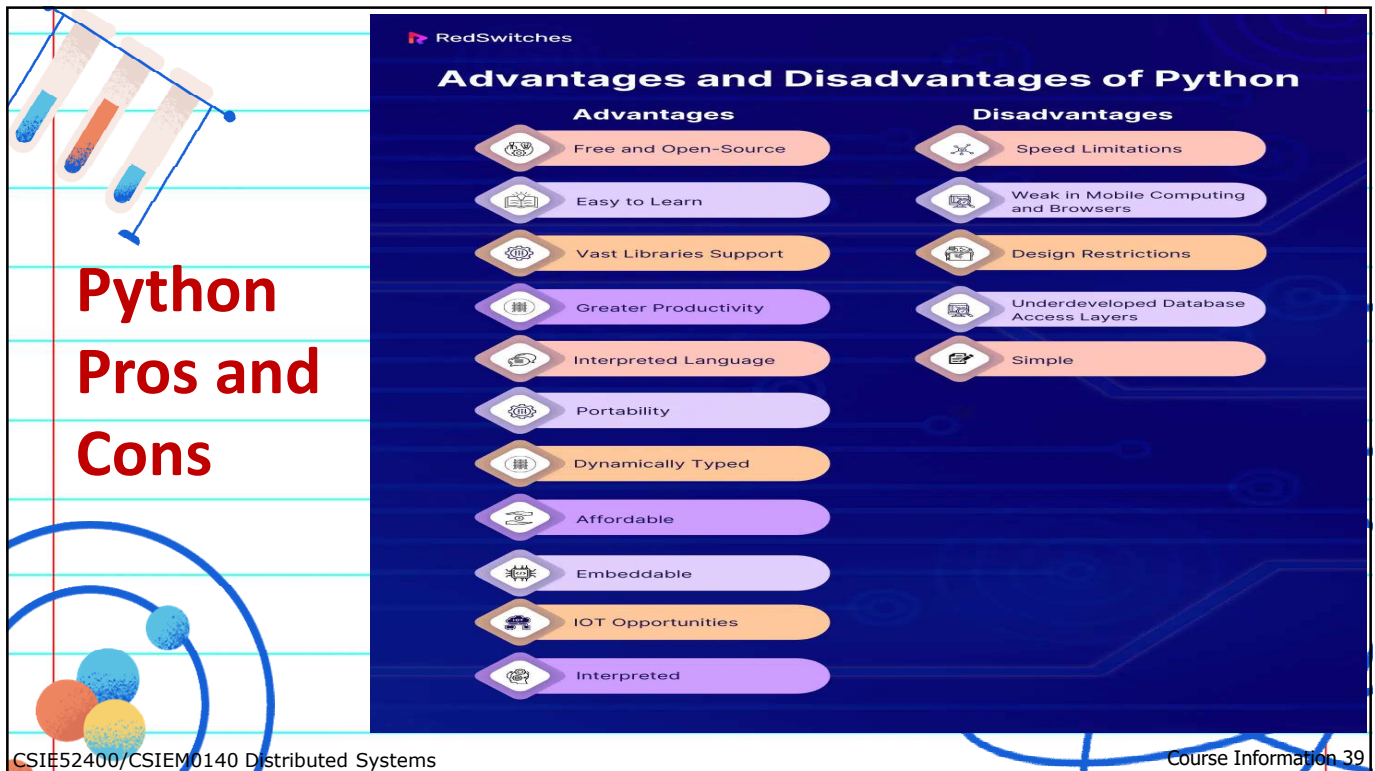
# IEEE Spectrum Lang Ranking

## Top Programming Languages 2023

Click a button to see a differently weighted ranking

**Spectrum** Jobs Trending





## Python Pros and Cons

**Advantages**

- Free and Open-Source
- Easy to Learn
- Vast Libraries Support
- Greater Productivity
- Interpreted Language
- Portability
- Dynamically Typed
- Affordable
- Embeddable
- IOT Opportunities
- Interpreted

**Disadvantages**

- Speed Limitations
- Weak in Mobile Computing and Browsers
- Design Restrictions
- Underdeveloped Database Access Layers
- Simple

CSIE52400/CSIEM0140 Distributed Systems Course Information 39

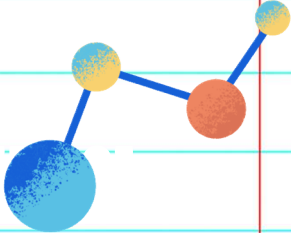
## Distributed Computing with Python

- Popular frameworks/libraries for distributed computing with Python:
  - **PySpark**: Python API for Apache Spark.
  - **Ray**: Parallel and distributed process-based execution framework for Python.
  - **MQTT (Message Queuing Telemetry Transport)**: facilitates communication between devices and servers in IoT applications.
  - **Blynk**: a platform that for building IoT applications on microcontrollers like Arduino using Python.
  - **Dask**: A flexible library for parallel/distributed computing with special focus on data science.
  - **dispy**: A generic and comprehensive framework for parallel/distributed computing in Python.
  - **Charm4py**: General-purpose parallel/distributed computing framework with Python and Charm++.

CSIE52400/CSIEM0140 Distributed Systems Course Information 40

## Why Java?

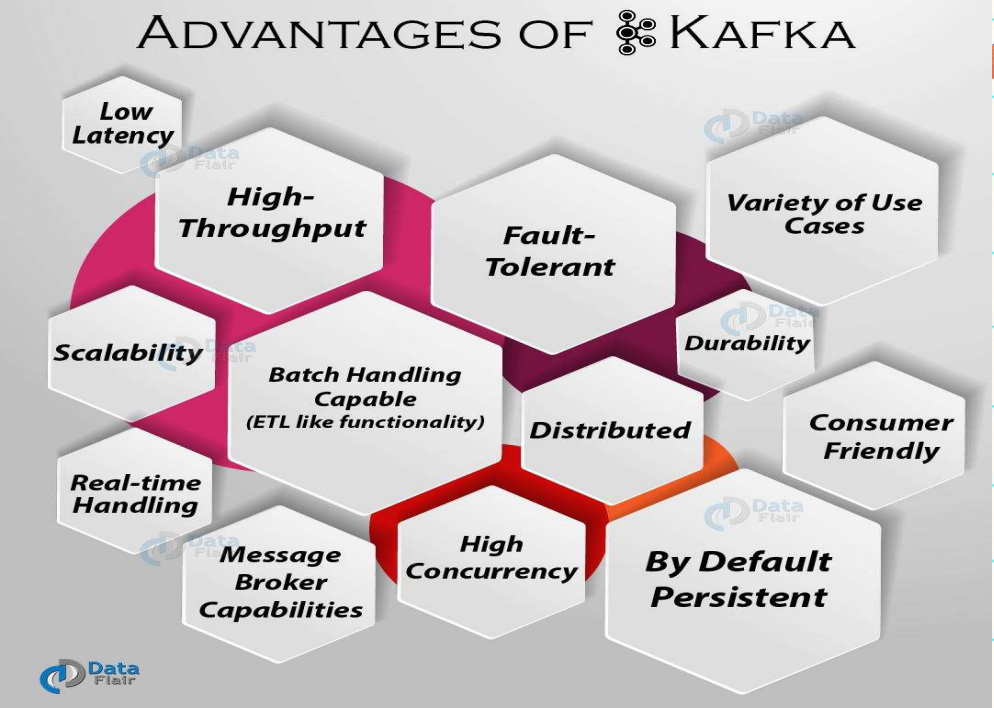
- Object-oriented environment (support distributed objects)
- Abstract interfaces (good for distributed communication)
- Platform independence (good for constructing distributed systems)
- Fault tolerance through exception handling (the `try/catch/finally` statement)
- Built in support for network/Internet/Web, XML, database, GUI, ...
- Multithreading and concurrency support
  - `java.lang.Thread` and `java.lang.Runnable`
  - Thread control and synchronization
- Mobile and cloud computing
- Security support
- It's FREE!!



CSIE52400/CSIEM0140 Distributed Systems Course Information 41

## ADVANTAGES OF KAFKA

### Why Kafka?



- Low Latency
- High-Throughput
- Fault-Tolerant
- Variety of Use Cases
- Scalability
- Batch Handling Capable (ETL like functionality)
- Distributed
- Durability
- Consumer Friendly
- Real-time Handling
- Message Broker Capabilities
- High Concurrency
- By Default Persistent

CSIE52400/CSIEM0140 Distributed Systems Course Information 42

## Why Spark?

- A fast and general engine for large-scale data processing
- Improve over Hadoop MapReduce
- Seamless combination of different models and workloads (batch, interactive queries, streaming, machine learning, graph, ...)
- Easy to use (from laptop to Hadoop cluster)
- Highly accessible (rich APIs for Python, Java, Scala, SQL, ...)
- Stream processing with Spark/Structured streaming
- Rich libraries for AI, ML, IoT, graph, data science, ...
- Integrate closely with other Big Data tools
- It's FREE!! (open source)

## Assignment 0

- What are the **consequences** of moving from a central system to a distributed system?
- Download and install the latest **Python** and **Spark**.
- Give it a try!!
- Can be done on **real cluster** or **VirtualBox**
  
- No due date, discuss in the class next time
- Nothing to turn in

## Quotes for the Start Date

- “The beginning is always today.”

— Mary Wollstonecraft

- “Where there is a will, there is a way.”

— Pauline Kael

- “Where there is a will, there is an **A.**”