2110625 - Data Science Architecture

Scalable Data Services

Asst.Prof. Natawut Nupairoj, Ph.D.
Department of Computer Engineering
Chulalongkorn University
natawut.n@chula.ac.th

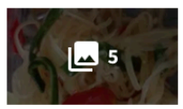
CHULA ENGINEERING
Foundation toward Innovation

Traditional Database SQL

- Based on ”relational model”
  - Data is split and stored into tables
  - Tables can be processed together using set-like operations JOIN SQL
  - Data model is usually normalized to remove duplication
- Very suitable for OLTP or transaction systems สิ่งที่ราคาแพงคือพื้นที่จัดเก็บ
  - Provide lots of complicated SQL operations - select
  - Lots of inserts and updates

Advertisement


Advertisement



Prep Time:	Total Time:	Servings:
20 mins	20 mins	6

Jump to Nutrition Facts

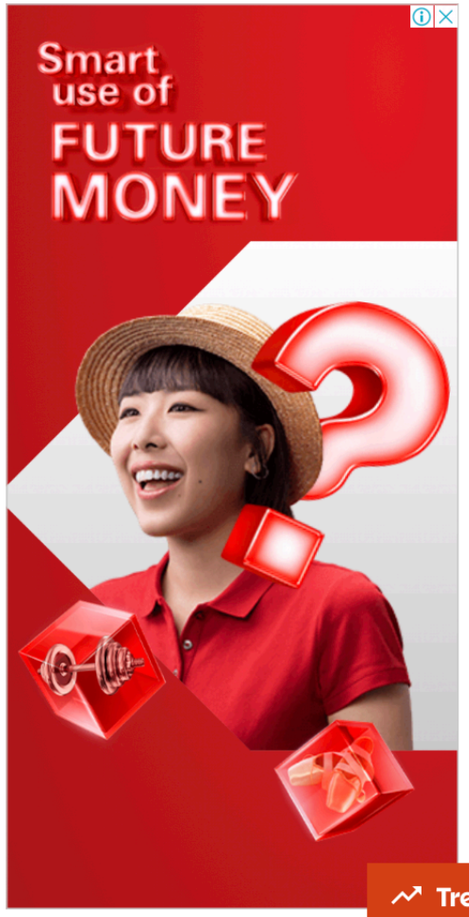
Ingredients

- 3 cloves garlic, peeled
- 3 Thai green chiles
- 6 green beans, cut into 1-inch pieces
- 1 large unripe papaya, peeled and cut into

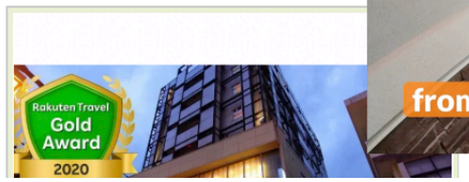
Local Offers

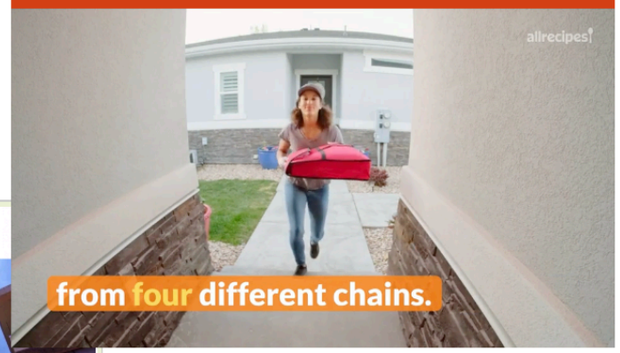
00000 Change

Oops! We cannot find any ingredients on sale near you. Do we have the correct zip code?

Advertisement


Advertisement

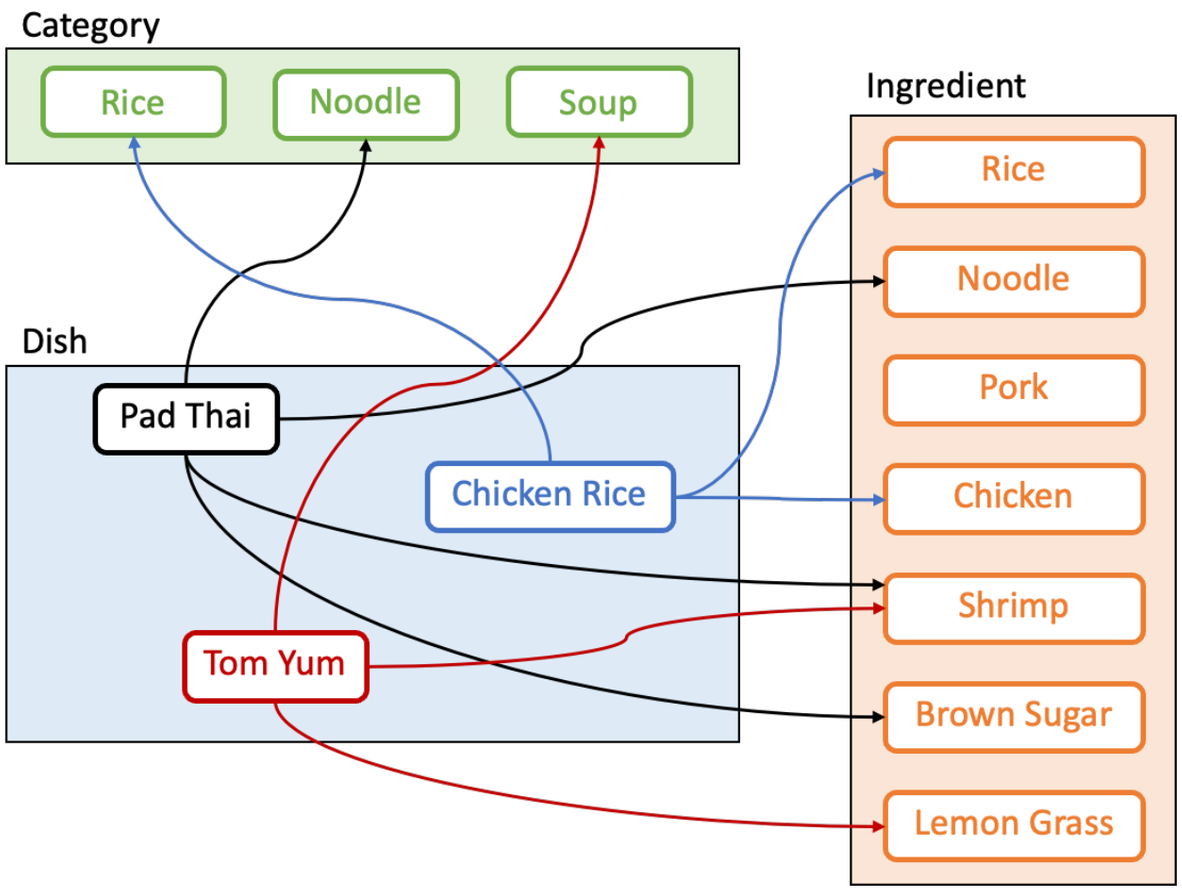
Advertisement


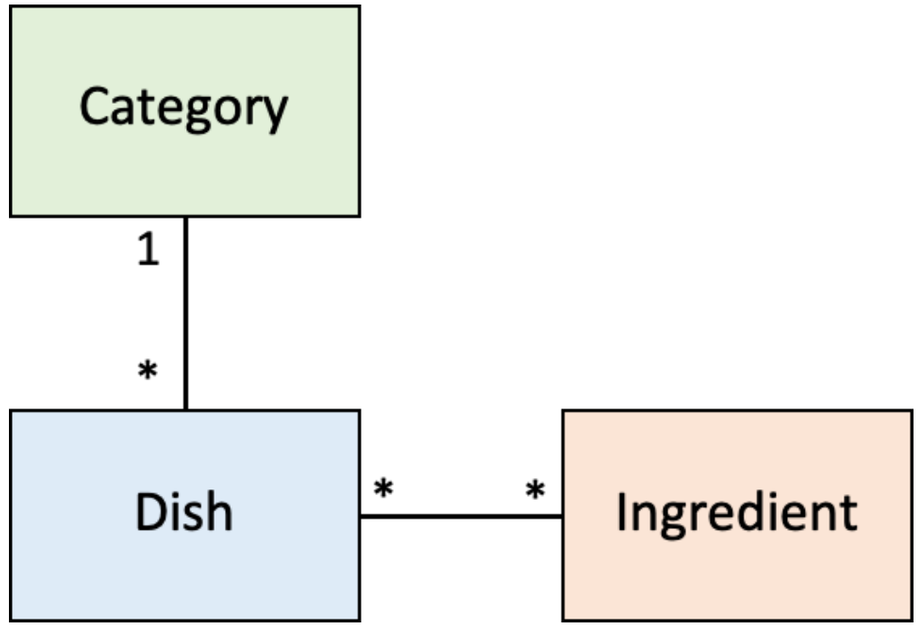
Trending Videos


Recipe Data is Hierarchical

- A dish has the following data:
  - belong to a category
  - consists of one or more main ingredients
    - Ingredients are different in portions and units
- For example:
  - Padthai belongs to a noodle category with 3 main ingredients noodle, shrimp, brown sugar
  - Tomyum belongs to a soup category; shrimp and lemon grass are main ingredients

Relational Database: Focus on Relation of Tables

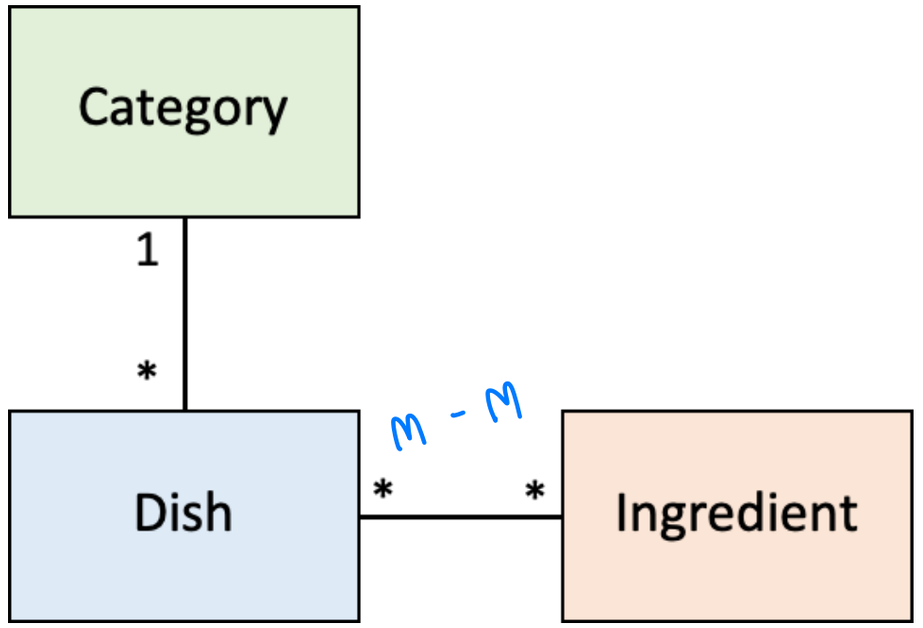




Example RDBMS: Recipe

Category

id	category
1	Soup
2	Noodle
3	Rice



Dish

id	name	category	…
1203	Pad Thai	2	
1288	Chicken Rice	3	

DI_link

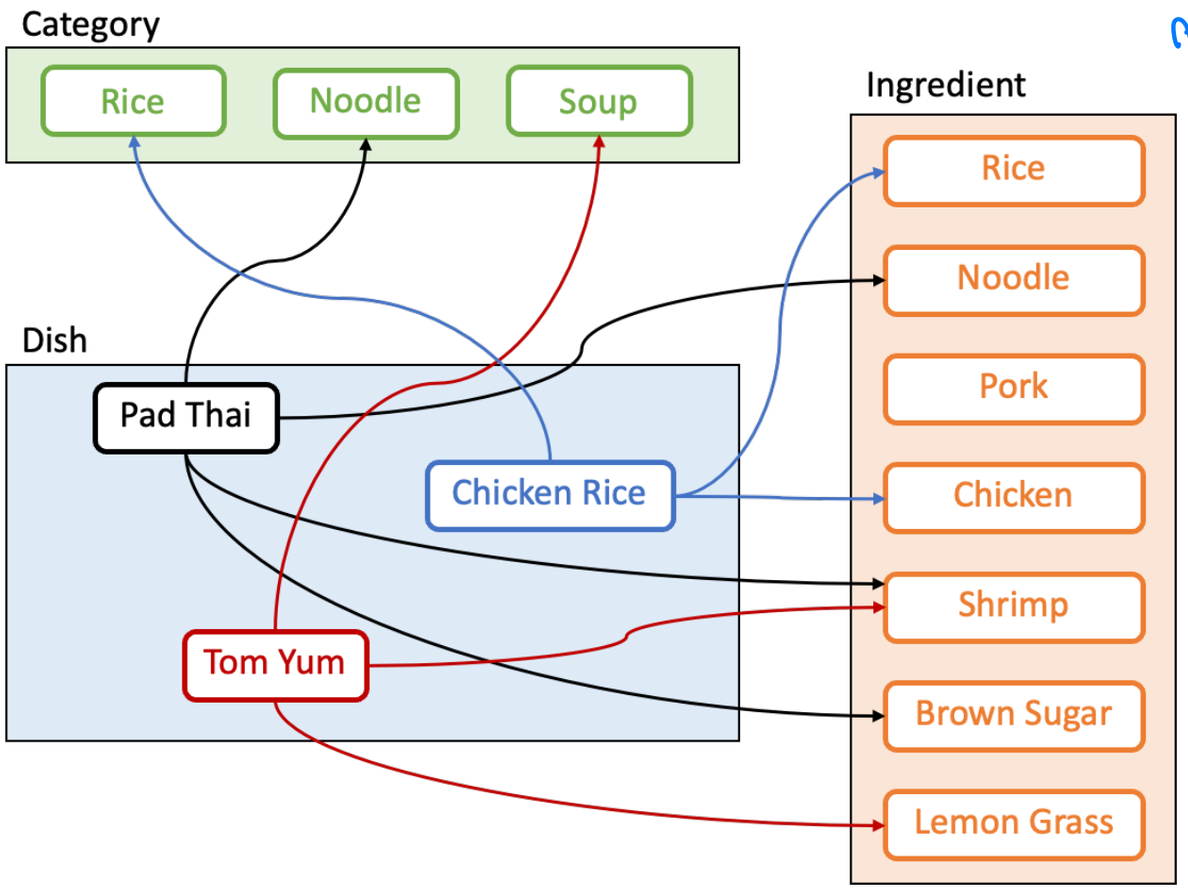
d_id	i_id	quantity	unit
1203	45	125	g
1203	48	150	g
1203	52	3	tbsp

Ingredient

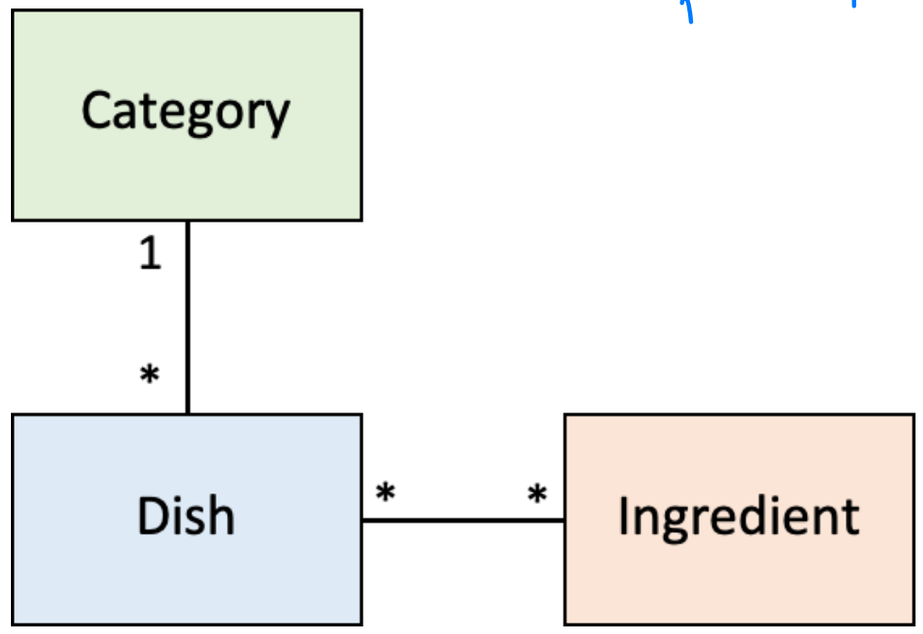
id	item
45	noodle
46	rice
48	chicken
52	brown sugar

SQL แบ่ง partition ยากมาก
ต้องแก้ตัว model

Relational Database: Normalized Data Model



ตารางต่างๆ ไปเกี่ยวข้องกัน
เราพยายามทำให้คุยน้อยสุด



How can we split these tables to lots of machines?

- Dishes of the same category in the same machine
- How about dishes and ingredients?

Or we can replicate data — lead to data consistency problems

Problems of Data Science Storage

- There are several needs for data analytics purposes e.g. traditional data store, caching, feature store
- Data is historical data and its volume can be huge
- Scalability is extremely important and “Relational + Consistency” can limit scalability
- SQL command can be very complex and time-consuming
  - It requires the synchronization of data accessing between multiple tables
  - It will be poor when using on more than a few servers in the same cluster

CAP Theorem (Brewer's Theorem)

- By Eric Brewer (University of California, Berkeley)
- It is impossible for a distributed computer system to simultaneously provide all three of the following guarantees: Consistency, Availability, Partition tolerance
- Scenario
  - Distributed system (clients and servers)
  - Multiple servers working together
  - Multiple clients may read or write on the same data at the same time

CAP Guarantees only 2 out of 3

- Consistency
  - Every read receives the most recent write or and error
  - Linearizable Consistency - every operation on an object should be atomic and consistent with the real-time order of operations on such as object ต่อให้มีหลาย server ก็ทำงานเป็นลำดับ (strong)
- Availability
  - Every request receives a (non-error) response - without guarantee that it contains the most recent write
  - Response from any server is good

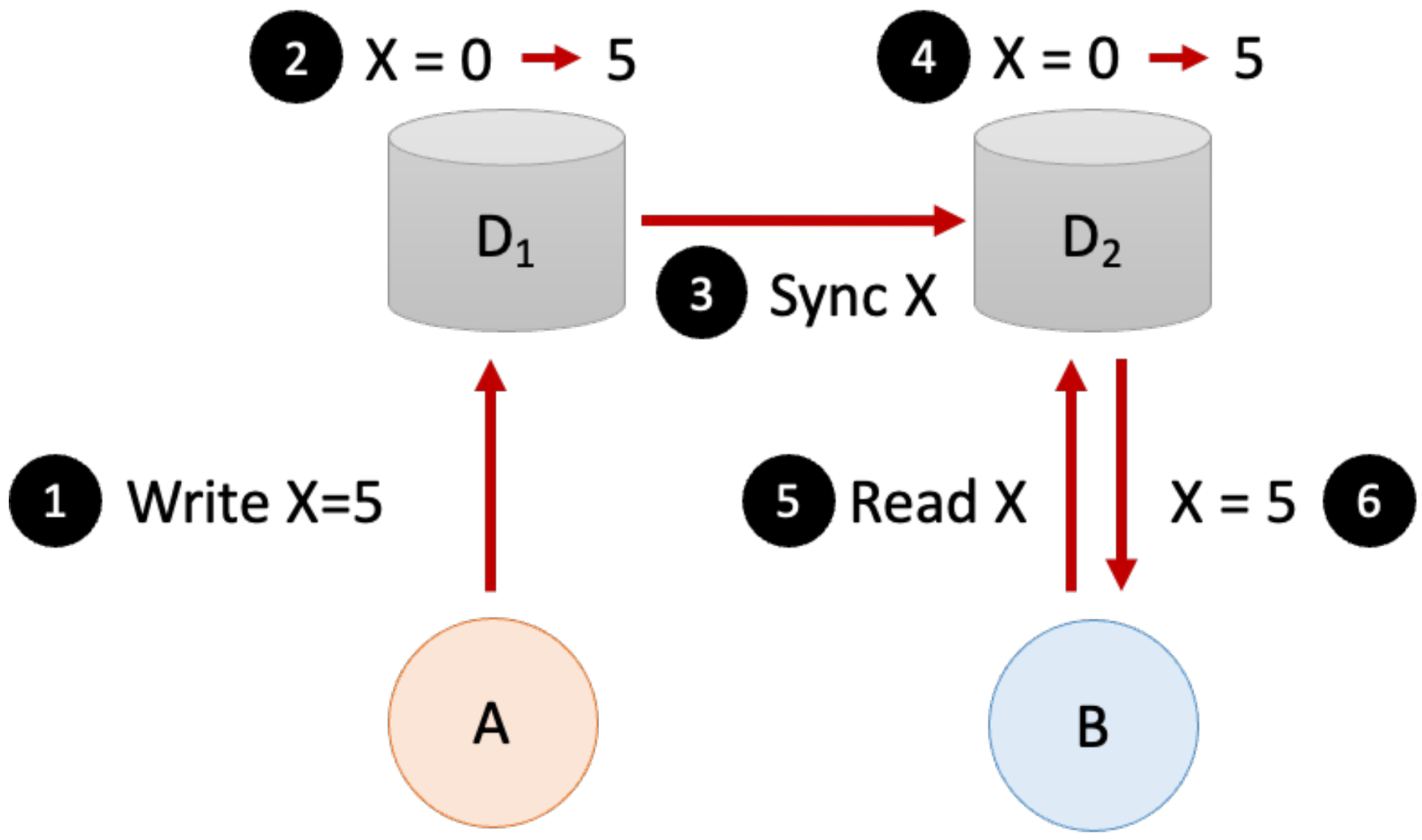
ทุก request ต้องตอบ ได้ค่าอะไรไม่ว่ากัน

CAP Guarantees only 2 out of 3

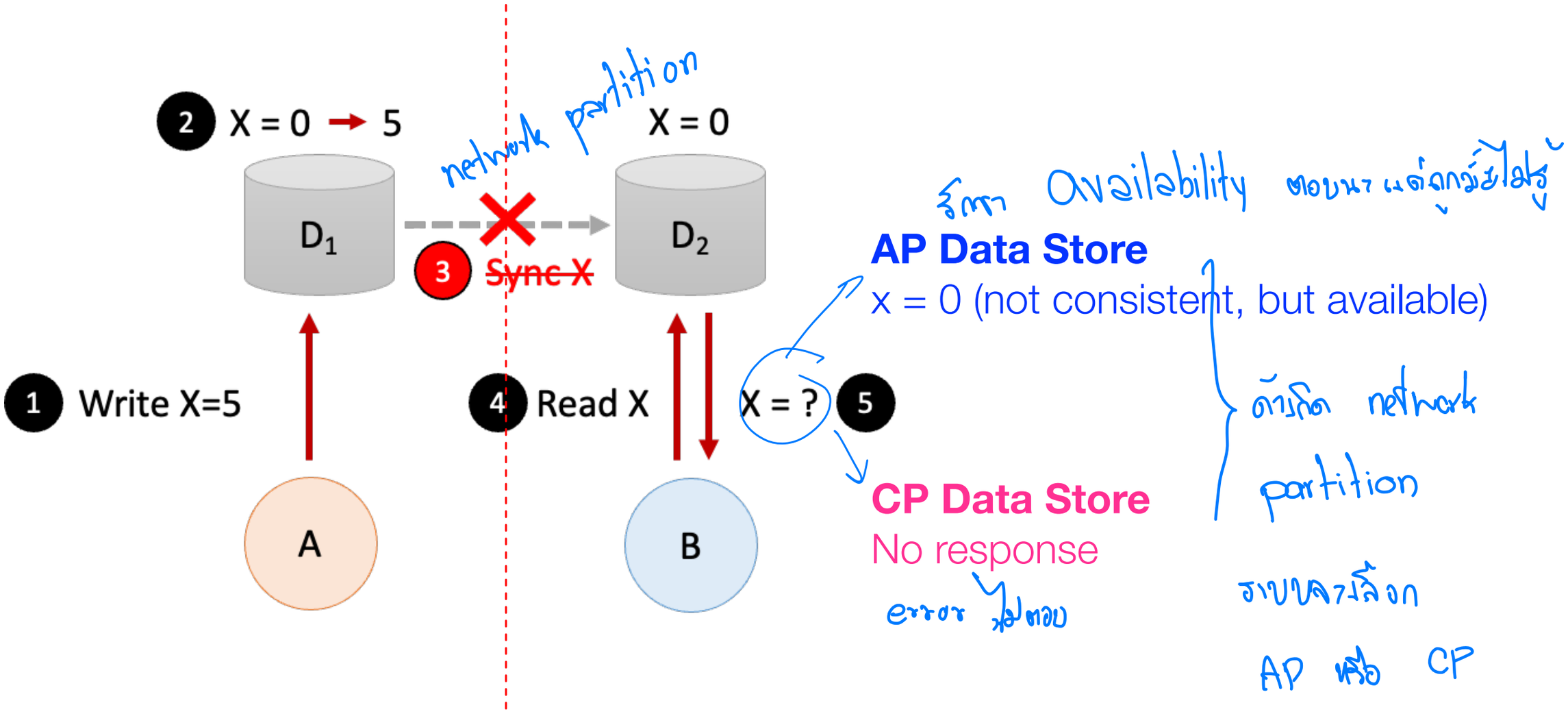
- Partition tolerance
  - the system should continue to operate correctly in case of a network partition สามารถทำงานต่อได้ แม้ NW ขาดจากกัน ไม่เห็น node กันและกัน
  - Scenarios
    - Other nodes may not see one node in the system.
    - A group of nodes may see each other but may not see another group of nodes
    - One node can see all other nodes, but those nodes can’t see each other
    - The network may lose packets

CA Data Store

server ต้องส่งข้อมูลถึงกัน



Given P, Choosing between C and A (at step 5)



PACELC Theorem

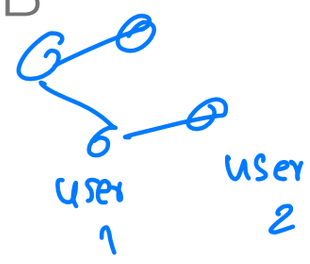
- An extension to the CAP theorem by Prof. Daniel J. Abadi from Yale University in 2010

- In case of network partitioning (P) in a distributed computer system, one has to choose between availability (A) and consistency (C), but else (E), even when the system is running normally in the absence of partitions, one has to choose between latency (L) and consistency (C)

- Reading assignment

The Landscape of NoSQL

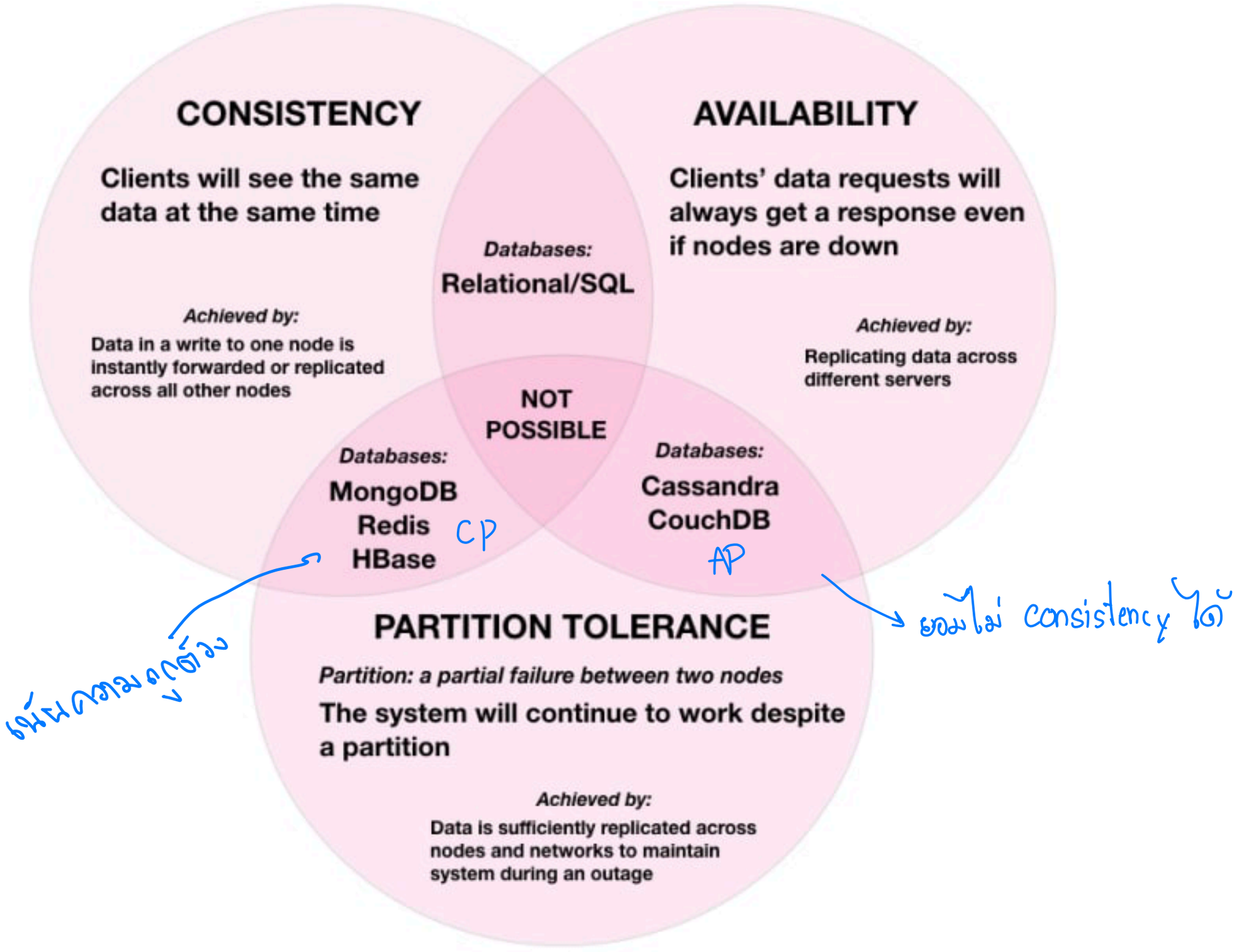
- Alternatives to SQL database - non-relational, distributed, and horizontally scalable
- Data is shared and distributed across multiple servers
- Typically use weak consistency model (but not always)
- Examples
  - Document: MongoDB, DynamoDB, CosmosDB, Couchbase, Firebase
  - Column: Cassandra, HBase, CosmosDB, Accumulo
  - Key-value: Redis, DynamoDB, CosmosDB, MemcacheDB
  - Graph: Neo4J, CosmosDB, ArrangoDB, OrientDB
  - Search Engine: Elasticsearch, Splunk, Solr



How NoSQL can “Scale”

- Principle ideas
  - Split data into chunks or shards
  - Distribute data across multiple servers
  - Must require minimum synchronization
- Have to give up some traditional features
  - No complex relational model
  - Relax consistency
  - Duplicated information (not space optimized)
  - Fast to insert new record, but not so fast to update the existing one

แบ่งข้อมูลเป็นก้อนๆ แล้วกระจายไปยัง server ตัวไหนก็ได้
ต่าง RDBMS แบ่ง chunk ยาก



Source: https://dev.to/katkelly/cap-theorem-why-you-can-t-have-it-all-ga1

References

- V. Joshi, “Ordering Distributed Events”, https://medium.com/baseds/ordering-distributed-events-29c1dd9d1eff

- W. Vogels, "Eventually consistent”, Communications of the ACM, Vol. 52, pp. 40–44, 2009.

- W. Springer, "Eventually consistent”, https://www.slideshare.net/springerw/eventually-consistent

- D. Abadi,"Consistency tradeoffs in modern distributed database system design: CAP is only part of the story”, Computer 45.2, pp. 37-42, 2012.