<cpt201_week2_note1> ——→ **Data Storage Structures**

Disk drive access time (tutorial 1)

<cpt201_week3_note1> ——→ **Indexing Techniques & B+ Tree Indexing**

ordered index, dense index, sparse index, primary index (clustering index), secondary index (non-clustering index)

B+ tree indexing (feature)

<cpt201_week4_note1> ——→ **Hash Indexing**

Hash indexing (static & dynamic) [compare]

<cpt201_week5_note1> ——→ **Relational Model**

Relational algebra operators

Operations of relational model (project, union, set difference, join, division)

<cpt201_week6_note1> ——→ **Query Evaluation**

Query evaluation (linear search, binary search, primary search on candidate key)

External sort-merge(cost)

Nested loop join, block nested loop join, indexed nested loop join, merge join, hash join (cost) (tutorial 4)

(r 是主表 relation，s 是附表，n 是 number of，b 是 blocks of)

(记得最后要分别乘以 block transfer t_T 以及 seek t_S 来对比)

<cpt201_week7_note1> ——→ **Query Optimization Introduction 1**

Materialization, Pipelining, Equivalence Rules [等价规则]

Cost, estimated size, number (tutorial 5)

(σ在越下面越好，使传递的表更小)

Tree optimization

<cpt201_week7_note2> ——→ **Query Optimization Introduction 2**

Estimation of the Size of Joins

method of V(A, r)

Heuristic Optimization [启发式优化]，Cost-based optimization 很昂贵

<cpt201_week9_note1> ——→ **Transaction Management**

Transaction Management, ACID, Concurrency, Schedule [调度]

Serializability [可串行性], Conflicts [冲突], View Serializability

Difference between view serializable and conflict sterilizable

Recoverable Schedules [可恢复的调度]

<cpt201_week10_note1> ——→ **Concurrency Control**

Concurrency Control[并发控制], Deadlock Handling [死结处理]

Lock-Based Protocols [基于锁定的协议] (Two phases of Locking Protocol) S&X : Lock-compatibility matrix [锁相容性矩阵]

Starvation and Deadlock

<cpt201_week10_note2> ——→ **Failure Recovery**

Failure Recovery: log, stable storage, redo, undo, checkpointing, recovery algorithm

<cpt201_week11_note1> ——→ Introduction to Object-Oriented Databases

object-oriented database, ORM

Comparison of Databases (不同数据库之间的比较，在尾)

<cpt201_week11_note2> ——→ Introduction to Distributed Databases

distributed database:

Fragmentation, Replication, Distributed Queries, Distributed Joins (Semi-join & bloom join),

Transaction coordinator [事务协调器], Distributed Deadlock, Failure Recovery in Distributed Databases,

Two Phase Commit (2PC)

<cpt201_week12_note1> ——→ Web Technologies and Data Storage 1

Basic concepts on XML, RDF

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The design goals of XML focus on simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages.

The **Resource Description Framework (RDF)** is a framework for representing information in the Web. An RDF statement consists of three components, referred to as a triple:

1. Subject is a resource being described by the triple.
2. Predicate describes the relationship between the subject and the object.
3. Object is a resource that is related to the subject.

<cpt201_week12_note2> ——→ Web Technologies and Data Storage 2

Concepts on SPARQL, semantic Web and linked data

SPARQL express queries across diverse data sources whether the data is stored natively as RDF or viewed as RDF via middleware. Contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions. Supports extensible value testing and constraining queries by source RDF graph and the results of SPARQL queries can be results sets or RDF graphs.

The **semantic Web** is an extension of the current web in which information is given well defined meaning, better enabling computers and people to work in cooperation.

Linked data design principles are use URIs as names for things, use HTTP URIs so that people can look up those names and when someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL). Include links to other URIs, so that they can discover more things.

<cpt201_week13_note1> ——→ Big Data Storage

Big data storage categorization, Concept on eventual consistency

Big data is a broad term for data sets and it can be described by the following characteristics:

①Volume [容积] (huge large amount of data: terabytes, petabytes, exabytes)

②Velocity [速率] (speed of data in and out: real-time, streaming)

③Variety [多样性] (range of data types and sources, non-relational data such as nested relation, documents, XML data, web data, graph, multimedia, flexible schema or no schema)

④Veracity [准确] (correctness and accuracy of information: data quality and reliability)

⑤Value [价值] (use machine learning, data mining, statistics, visualization, decision analysis techniques to extract/mine/derive previously unknown insights from data and become actionable knowledge, business

value)

Eventual Consistency is a guarantee that when an update is made in a distributed database, that update will eventually be reflected in all nodes that store the data, resulting in the same response every time the data is queried.

<cpt201_week13_note2> ⟶ Blockchain-based Storage

Basic concepts on consensus algorithms used in public blockchain based storage, Properties of blockchain

A **consensus algorithm** is the pre-determined process by which a blockchain reaches consensus. When a blockchain reaches consensus, enough nodes agree about the content and veracity of a block to mint a fresh block. Involved proof of work, proof of stake, proof of activity, proof of elapsed time and Byzantine Consensus

Summary of **blockchain properties**:

- Decentralization[去中心化] – majority consensus with no central authority.
- Tamper resistance[抗干扰] – infeasibility of changing the contents of blocks on the blockchain.
- Irrefutability[不可反驳性] – user cannot deny having submitted a transaction.
- Anonymity[匿名] – IDs not directly tied to any real-world entity

<cpt201_week14_note1> ⟶ Data Analytics

Basic concepts on classification, clustering and regression

Basic ideas of common algorithms for classification

Basic concepts for association rules and possible applications

Concepts on Support and confidence of association rules

Classification is the type of supervised machine learning, for any given input, the classification algorithm helps in the prediction of the class of the output variables; **Clustering** is unsupervised machine learning algorithm, it is used to group data point having similar characteristics as cluster; **Regression** is the type of supervised machine learning, When the output is continuous.

Basic ideas of **common algorithms for classification** have Bayesian classifiers, Support Vector Machine, Neural Network and Deep Neural Network.

Association rules is given a set of transactions, find rules that predict the occurrence of an item based on the occurrences of other items in the transaction. A typical example is a Market Based Analysis.

Support and confidence of association rules: Confidence is not the optimal method for every concept in association rule mining. The disadvantage of using it is that it does not offer multiple difference outlooks on the associations. Unlike support, confidence does not provide the perspective of relationships between certain items in comparison to the entire dataset.