THE UNIVERSITY OF WESTERN AUSTRALIA

Data Warehousing

Lecture 1 Introduction to Data Warehousing

CITS3401
CITS5504

Zeyi Wen

Computer Science and Software Engineering

School of Maths, Physics and Computing

Acknowledgement: The lecture slides are adopted from online sources.

Outline of Today's Lecture

- Unit Structure and Overview
- A Brief History of Data(base) Technologies
- Storing Data in Databases and Data Warehouses
- What is Data Warehouse
- OLTP vs. OLAP
- Data Mining

Teaching team for CITS3401&5504

- Lecturer and Unit Coordinator
  - **Dr. Zeyi Wen**
    - Email: zeyi.wen@uwa.edu.au
    - Office: CSSE Room 1.16
    - Webpage: https://research-repository.uwa.edu.au/en/persons/zeyi-wen
- Lab Facilitators
  - **Nur Al Hasan Haldar**
    - recent UWA PhD graduate
    - areas of research: databases and data science
  - **Yin Zhong**
    - Awarded UWA Master's degree with Distinction
- Post general questions (which other students are also likely to have) in help3401 or help5504 forums.

CITS3401 and CITS5504

THE UNIVERSITY OF WESTERN AUSTRALIA

- Same Lecture Hours and Venue:
  - Time: Thursdays 12:00 – 14:00 pm
  - Venue: ARTS: [G57] Alexander Lecture Theatre
  - All the lectures will be recorded.
  - Lectures will be live-streamed with Microsoft Teams (link here).
- Same Consultation Hour:
  - Thursdays 14:00 – 15:00 pm (no appointment needed)
  - Other times can be arranged too (send an email to book)
  - Zeyi is in CSSE Room 1.16, and will also be on Microsoft Teams (link here).
- Similar Teaching Material
  - Same lecture slides and lab sheets
  - **Similar** projects (e.g. extra challenging tasks for CITS5504 students)
- Different Web Pages on LMS

CITS3401 and CITS5504 Labs

- Different Lab Sessions (from Week 2 onward):

CITS3401 Lab	Mondays	12:00 - 14:00	CSSE: [201] Computer Lab
	Mondays	16:00 - 18:00	Online (Microsoft Teams link)
	Tuesdays	14:00 - 16:00	CSSE: [201] Computer Lab
	Tuesdays	16:00 – 18:00	CSSE: [201] Computer Lab
	Wednesdays	9:00 – 11:00	MATH: [123] Computer Lab

Online sessions

CITS5504 Lab	Wednesdays	12:00 - 14:00	CSSE: [205] Computer Lab
	Wednesdays	14:00 - 16:00	Online (Microsoft Teams link)
	Thursdays	8:00 - 10:00 am	ENCM: [207B] South Civil Computer Room B

Lab sessions

- In **Week 2**, **no Monday lab** sessions due to Labour Day.
  - Affected students can join lab sessions on Tue, Wed and Thu
  - Lab sheets with instructions will be provided—you can complete the Week 2 lab tasks at home.
  - Post questions on help forums.

Same Assessment Structures

- Two Projects: 50%
  - Each project contributes to 25% of this unit.
  - Project specifications will be available on the LMS page.

- Final Examination: 50%

Same Assessment Structures (cont.)

THE UNIVERSITY OF WESTERN AUSTRALIA

Two projects worth 50%; each contributes 25%.

- Project 1 submission
  - individual effort
  - (5%) software environment setup evaluation in Week 4
    - during lab time
  - (95%) final submission at the end of Week 6

- Project 2 submission
  - group of 1-2 students
  - final submission at the end of Week 12

- Project 1: analysis of a business scenario through an OLAP (Online Analytical Processing) tool
  - a suite of Microsoft Tools
    - Microsoft SQL Server, and SQL Server Management Studio (SSMS)
    - Visual Studio, and Microsoft SQL Server Data Tools (SSDT)
    - Microsoft PowerBI
- Project 2: analysis of a data mining and exploration problem using Weka.
  - Weka (Waikato Environment for Knowledge Analysis)
  - Weka is a collection of algorithms for data mining tasks.
  - http://www.cs.waikato.ac.nz/ml/weka/

Setting up Software Environment

There are three approaches

1. Install all the software in your **own computer**
   - Recommended approach
   - **Pros**: fast and accessible at any time
   - **Cons**: time to install and fix some tech issues
2. Install OpenVPN and Remote Desktop in your **own computer** to connect to a Virtual Machine (VM) in Azure Cloud
   - **Pros**: easier to setup
   - **Cons**: limited usage **5 hours per week**; slow
3. Use browsers to connect to a VM in Azure Cloud
   - **Pros**: easiest to setup; don't need own computer
   - **Cons**: limited usage **5 hours per week**; **slowest**

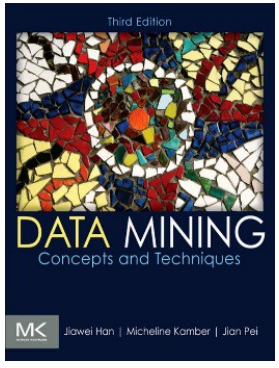
Penalty may apply if > 5 hours per week.

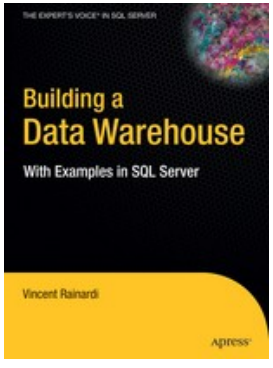
- If you use a VM in Azure Cloud, **always backup your work or important data to OneDrive**.
- All your activities on setting the VM are stored in log files, so **use VM for this unit only**.

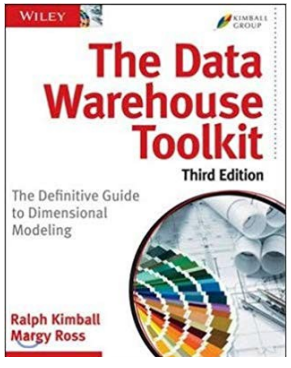
Text Books and Recommended Readings

THE UNIVERSITY OF WESTERN AUSTRALIA

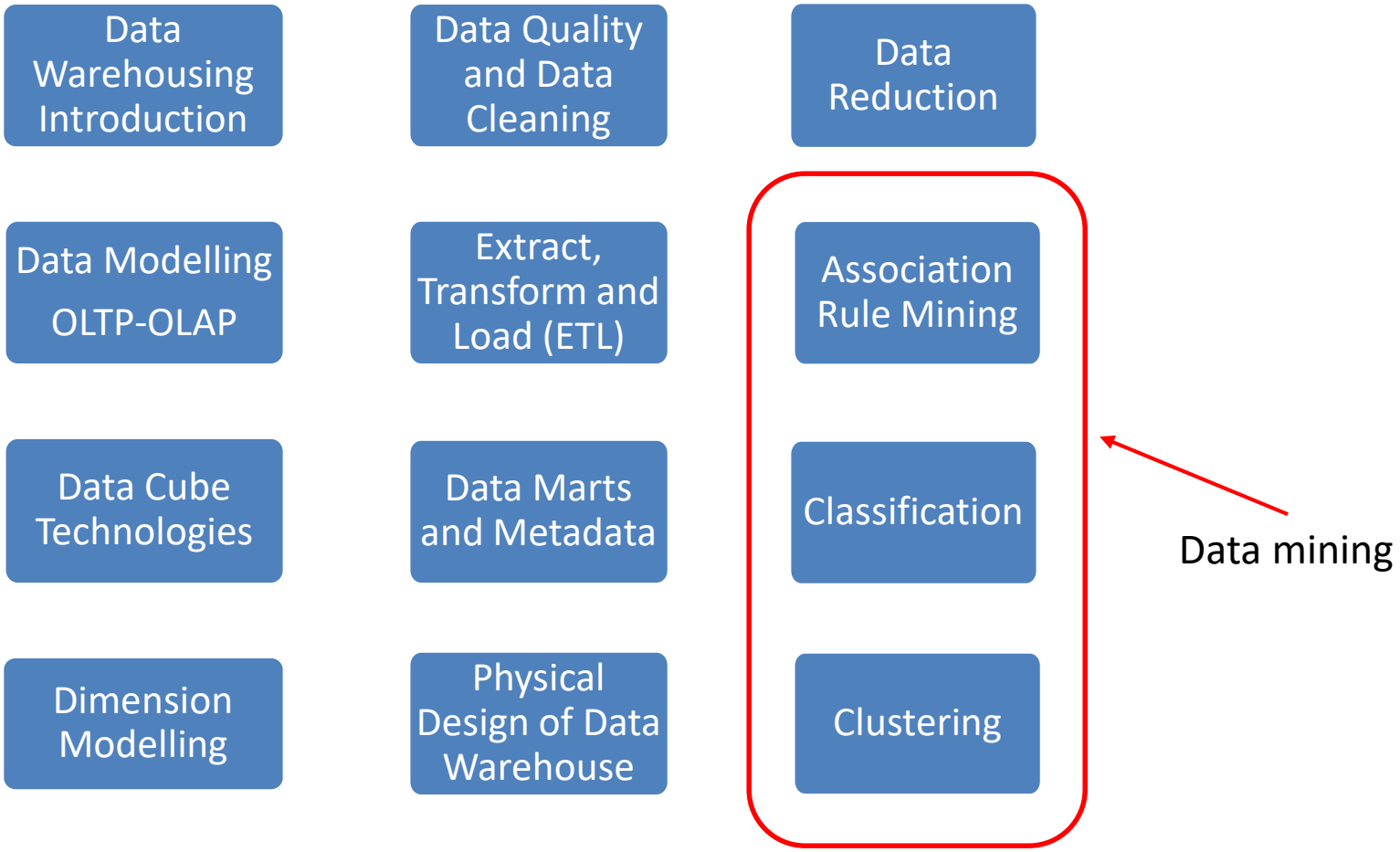
- **Course Text Books:**
  - **Data Mining: Concepts and Techniques**
    - 3rd ed., by J. Han, M. Kamber, and J. Pei. 2011
- **Recommended Readings:**
  - **Building a Data Warehouse with Examples in SQL Server**
    - by Vincent Rainardi. 2008
  - **The Data Warehouse Toolkit, The Definitive Guide to Dimensional Modelling**
    - 3rd ed., by Ralph Kimball and Margy Ross







Unit Overview



Outline of Today's Lecture

- Unit Structure and Overview
- A Brief History of Data(base) Technologies
- Storing Data in Databases and Data Warehouses
- What is Data Warehouse
- OLTP vs. OLAP
- Data Mining

The Evolution of Data Technologies

THE UNIVERSITY OF WESTERN AUSTRALIA

- The “dark age”: paper forms in file cabinets
- Computerised systems emerged
  - Initially for big projects like Social Security
  - Same functionality as old paper-based systems
- The “golden age”: databases are everywhere
  - Most activities tracked electronically
  - Stored data provides detailed history of activity
- The next step: use data for decision-making
  - Knowledge discovery from data (a.k.a. Data Mining)
  - One of the enabling technologies: data warehousing

Cambridge Analytica Ltd was founded in 2013.

Evolution of Database Technology

THE UNIVERSITY OF WESTERN AUSTRALIA

- **1960s:**
  - (Electronic) Data collection, database creation
  - IMS (database system by IBM) and network DBMS
  - IMS introduced application code should be separated from data.

- **1970s:**
  - Relational data model, relational DBMS implementation
  - The term “relational database” was invented by E. F. Codd at IBM.
  - E. F. Codd won Turing Award in 1981.

- **1980s:**
  - RDBMS, advanced data models (Object Oriented, deductive, etc.)
  - Application-oriented DBMS (spatial, scientific, engineering, etc.)
  - SQL standard adopted by ISO and ANSI.

Microsoft SQL Server was first released in 1989; MySQL in 1995

- **1990s:**
  - Data mining, data warehousing, multimedia DBs, and Web DBs

Google was founded in 1998; Yahoo! in 1994; Baidu and Yandex in 2000

Evolution of Database Technology (cont.)

THE UNIVERSITY OF WESTERN AUSTRALIA

- **2000s**
  - Stream data management and mining
  - Data mining and its applications
  - Web tech. (XML, data integration) and geographic info. systems
  - Frequently asked question in job interviews:
    - **Did you work on XML related research?**
- **2010s**

Apache Spark was first released in 2014.

  - NoSQL (Graph Databases, Document Stores)
  - Mining from Varieties of Data:
    - Multimedia Data (Images, Audios, and Videos)
    - Natural Language Processing, Social Network Data
  - Machine (Deep) Learning on various data types (e.g. text, videos)
  - Self-Driving Databases, Autonomous Databases
  - Frequently asked question in job interviews:
    - **Did you work on ML related research?**
- **2020s**
  - Leave this homework to you!

From empirical science to data science

- **Before 1600, empirical science**

- **1600-1950s, theoretical science**
  - Each discipline has grown a theoretical component. Theoretical models often motivate experiments and generalise our understanding.

- **1950s-1990s, computational science**
  - Most disciplines have grown a third, *computational* branch
    - e.g. computational ecology, computational physics, comp. linguistics
  - Computational Science traditionally meant simulation. It grew out of our inability to find closed-form solutions for complex mathematical models.

- **1990-now, data science (and data-driven science)**
  - The flood of data from new scientific instruments and simulations
  - The ability to economically store and manage *petabytes* of data online
  - Internet and computing Grid that makes all the data accessible

Outline of Today's Lecture

- Unit Structure and Overview
- A Brief History of Data(base) Technologies
- Storing Data in Databases and Data Warehouses
- What is Data Warehouse
- OLTP vs. OLAP
- Data Mining

Where to Store Data

THE UNIVERSITY OF WESTERN AUSTRALIA

Relational Databases

- Support “delete, insert, update, and query”
- Consistency/integrity is crucial
- Queries are often simple
- Data from single department/organisation

Data Warehouses

- Mainly support “query”
- Queries are more complex

Other distributed file systems

- Hadoop Distributed File Systems (HDFS)

Storing Data in Relational Database

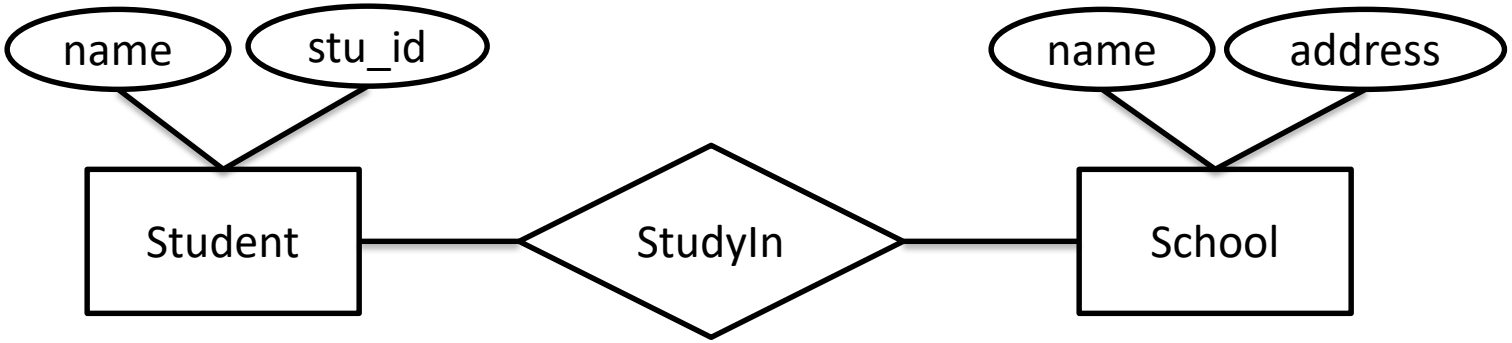
- A relational database is **a collection of tables**, each of which is assigned a unique name.
  - Each table consists of a set of attributes (columns or fields) and usually stores a large set of tuples (records or rows).
  - Each tuple in a relational table represents an object identified by unique key and described by a set of attribute values.

Student_id	Name	Unit_id	School_id	Score
2212201	Jan Smith	CITS3401	006	98
…	…	…	…	….

Storing Data in Relational Database

- A relational database is **a collection of tables**, each of which is assigned a unique name.
  - Each table consists of a set of attributes (columns or fields) and usually stores a large set of tuples (records or rows).
  - Each tuple in a relational table represents an object identified by unique key and described by a set of attribute values.

- A semantic data model, such as the ***Entity Relationship (ER)*** data model, is often constructed for relational databases.
  - An ER data model represents the database as a set of entities and their relationships.



Storing Data in Relational Database (cont.)

- Relational data can be accessed by database queries written in a relational language such as **SQL**.

- A given query is transformed into a set of relational operations such as *join*, *selection* and *projection,* and is then optimised for efficient processing.

- Efficiency of retrieval, efficiency of update and integrity are the key requirements of a good relational database.

An Example - *AllElectronics*

- Four relational tables: *customer*, *item*, *employee* and *branch*.

customer

cust_ID	*name*	*address*	*age*	*income*	*credit_info*	*category*	...
C1	Smith, Sandy	1223 Lake Ave., Chicago, IL	31	$78000	1	3	...
...	...	...	...	...	...	...	...

item

item_ID	*name*	*brand*	*category*	*type*	*price*	*place_made*	*supplier*	*cost*
I3	hi-res-TV	Toshiba	high resolution	TV	$988.00	Japan	NikoX	$600.00
I8	Laptop	Dell	laptop	computer	$1369.00	USA	Dell	$983.00
...	...	...	...	...	...	...	...	...

employee

empl_ID	*name*	*category*	*group*	*salary*	*commission*
E55	Jones, Jane	home entertainment	manager	$118,000	2%
...	...	...	...	...	...

branch

branch_ID	*name*	*address*
B1	City Square	396 Michigan Ave., Chicago, IL
...	...	...

An Example - *AllElectronics*

- Four relational tables: *customer*, *item*, *employee* and *branch*.
- Each relation consists of a set of attributes.

customer

cust_ID	*name*	*address*	*age*	*income*	*credit_info*	*category*	...
C1	Smith, Sandy	1223 Lake Ave., Chicago, IL	31	$78000	1	3	...
...	...	...	...	...	...	...	...

item

item_ID	*name*	*brand*	*category*	*type*	*price*	*place_made*	*supplier*	*cost*
I3	hi-res-TV	Toshiba	high resolution	TV	$988.00	Japan	NikoX	$600.00
I8	Laptop	Dell	laptop	computer	$1369.00	USA	Dell	$983.00
...	...	...	...	...	...	...	...	...

employee

empl_ID	*name*	*category*	*group*	*salary*	*commission*
E55	Jones, Jane	home entertainment	manager	$118,000	2%
...	...	...	...	...	...

branch

branch_ID	*name*	*address*
B1	City Square	396 Michigan Ave., Chicago, IL
...	...	...

purchases

trans_ID	*cust_ID*	*empl_ID*	*date*	*time*	*method_paid*	*amount*
T100	C1	E55	03/21/2005	15:45	Visa	$1357.00
...	...	...	...	...	...	...

items_sold

trans_ID	*item_ID*	*qty*
T100	I3	1
T100	I8	2
...	...	...

works_at

empl_ID	*branch_ID*
E55	B1
...	...

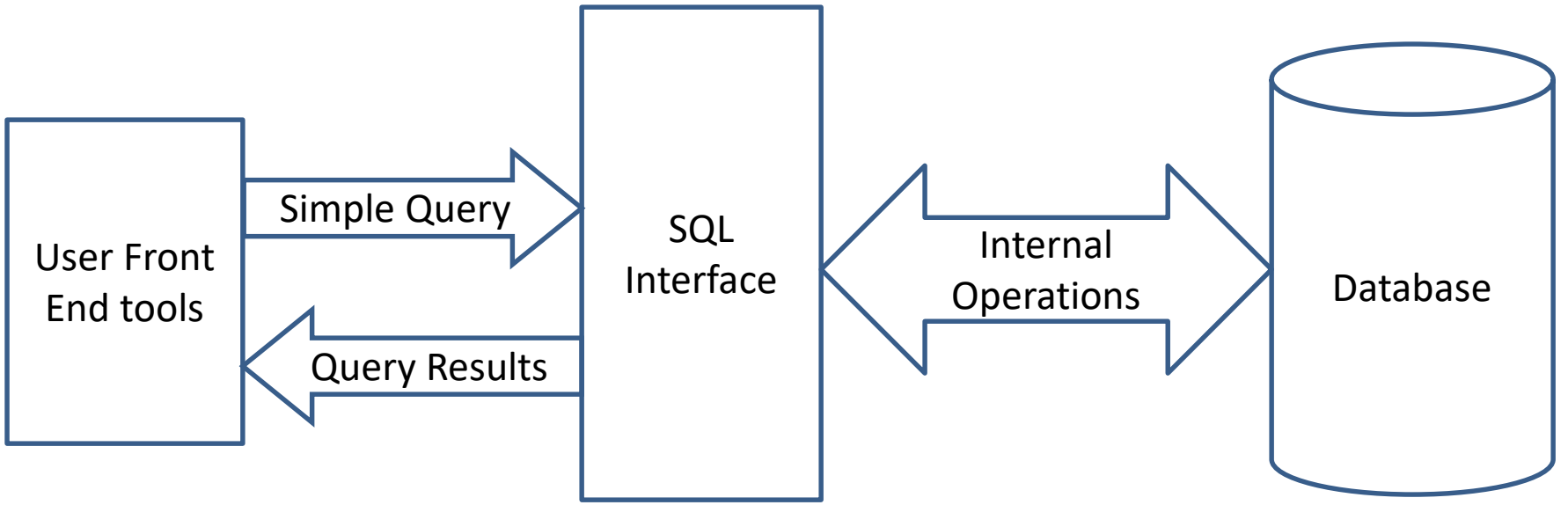
Purpose of Relational Database

- The main purpose of a relational database is to store data **correctly** and retrieve data **on demand**.

- This type of data processing is sometime called Online Transaction Processing (OLTP).

- Relational databases are passive data repositories in the sense that a query only shows you what is stored in the database, but cannot tell you much about the **meaning or trend** of the data.

Query Answering in Relational Database

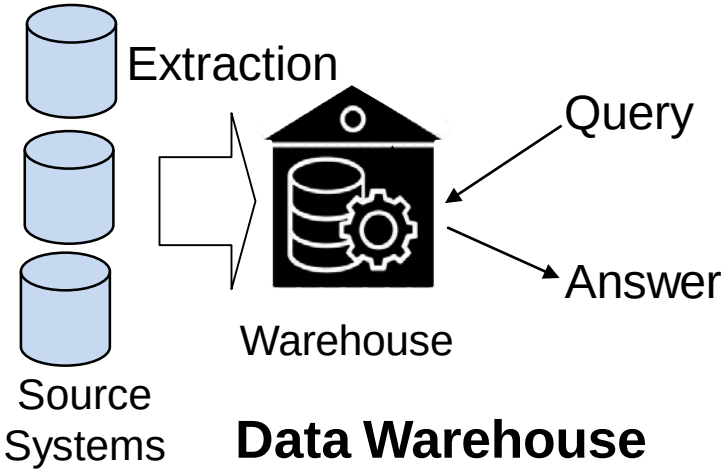


Transactional Database

- A transactional database consists of a file where each record represents a transaction.
- Supports nested relation
- Transaction id: Items, customer name, date…
- Sample Queries**:**
  - Show me all the items purchased by ‘X’
  - How many transactions include item number ‘Y’?
  - market basket data analysis: Which items sold well together? **(Frequent item set)**

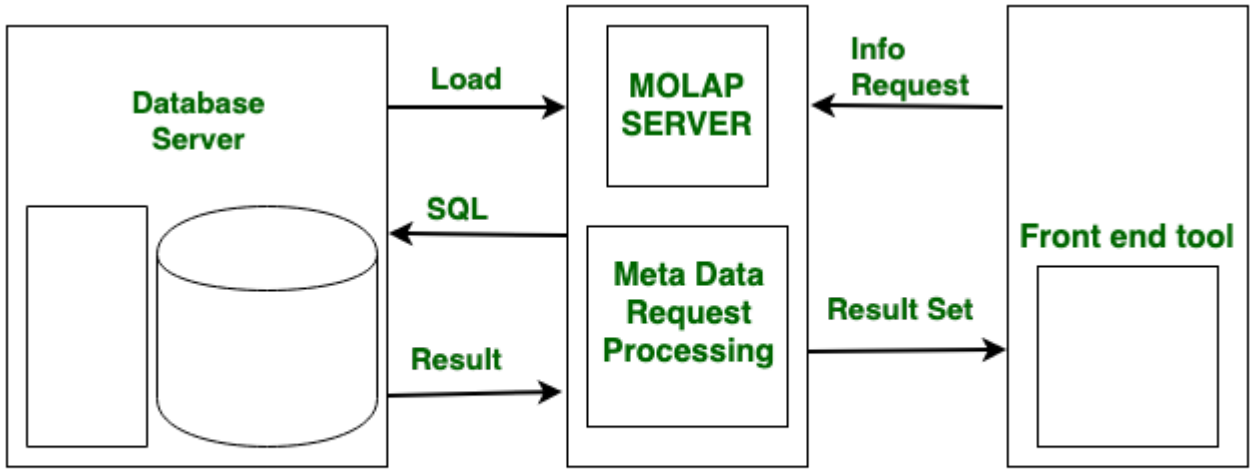
Storing Data in Data Warehouses

- Data are from multiple data sources
  - Relational DB systems, flat files, .csv files, …
- Data are integrated into a data warehouse
- Data are stored in DBMS using tables.



Storing Data in Data Warehouses

- Data are from multiple data sources
  - Relational DB systems, flat files, .csv files, …
- Data are integrated into a data warehouse
- Data are stored in DBMS using tables.
- Query answering process is more complex.

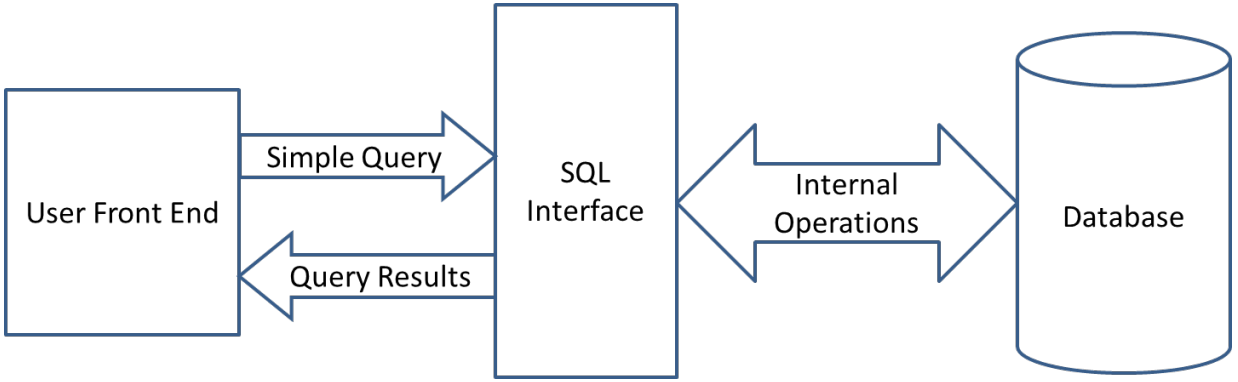


MOLAP Architecture

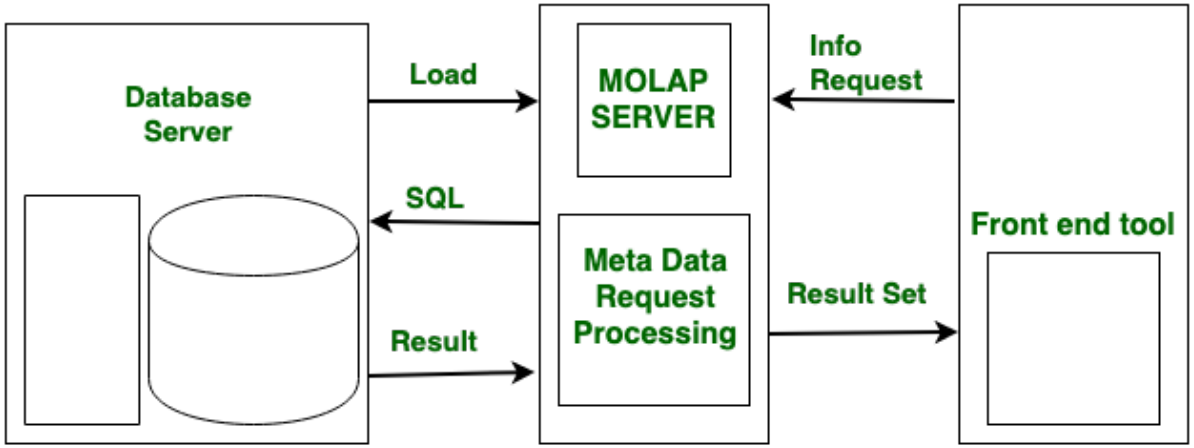
Query Answering Comparison

THE UNIVERSITY OF WESTERN AUSTRALIA

Query answering in Relational Databases



Query answering in Data Warehouses



MOLAP Architecture

Examples of More Complex Queries

THE UNIVERSITY OF WESTERN AUSTRALIA

- Show a list of all items that were sold in the last quarter
- Show the total sales of the last month, grouped by branch
- Which sales person has the highest amount of sales?
- How many sales transactions occurred in September?

Outline of Today's Lecture

- Unit Structure and Overview
- A Brief History of Data(base) Technologies
- Storing Data in Databases and Data Warehouses
- What is Data Warehouse
- OLTP vs. OLAP
- Data Mining

What is Data Warehouse?

- **A data warehouse is a**
  - subject-oriented,
  - integrated,
  - time-variant, and
  - nonvolatile

  collection of data in support of management's decision-making process.

Data Warehouse – Subject Oriented

- Organised around major subjects, such as customer, supplier, product, sales, time.

- Focusing on the modeling and analysis of data for decision makers, not on daily operations or transaction processing.

- Provide a simple and concise view around particular subject issues by excluding data that are not useful in the decision making process.

Data Warehouse - Integrated

- Constructed by integrating multiple, heterogeneous data sources
  - relational databases, flat files, on-line transaction records

- Data cleaning and data integration techniques are applied.
  - Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources

Data Warehouse – Integrated (cont.)

Data Integration is hard.

- Data warehouses combine data from multiple sources
- Data must be translated into a consistent format

- Data integration represents ~80% of effort for a typical data warehouse project!

- Some reasons why it's hard:
  - Metadata is poor or non-existent
  - Data quality is often bad
    - Missing or default values
    - Multiple spellings of the same thing (UWA vs. Uni. of WA. vs. University of Western Australia)
  - Inconsistent semantics
    - Marks across different universities?

Data Warehouse – Time Variant

- The time horizon for the data warehouse is significantly longer than that of operational DB systems.
  - Operational database: current value data.
  - Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)

- Every key structure in the data warehouse
  - Contains an element of time, explicitly or implicitly

Data Warehouse – Non-Volatile

- A physically separate store of data transformed from the operational environment.
- Operational update of data does not occur in the data warehouse environment.
  - Does not require transaction processing, recovery, and concurrency control mechanisms
  - Requires only two operations :
    - *initial loading of data* and *access of data*.

Outline of Today's Lecture

- Unit Structure and Overview
- A Brief History of Data(base) Technologies
- Storing Data in Databases and Data Warehouses
- What is Data Warehouse
- OLTP vs. OLAP
- Data Mining

Key operations of Data Warehouses

Key operations of RDBMS

Data Warehouse (OLAP) vs. Operational DBMS (OLTP)

THE UNIVERSITY OF WESTERN AUSTRALIA

- OLTP (on-line transaction processing)
  - Major task of traditional relational DBMS
  - Day-to-day operations: purchasing, inventory, banking, manufacturing, payroll, registration, accounting, etc.

- OLAP (on-line analytical processing)
  - Major task of data warehouse system
  - Data analysis and decision making
  - Can organise and present data in various forms and combinations

Data Warehouse (OLAP) vs. Operational DBMS (OLTP)

THE UNIVERSITY OF WESTERN AUSTRALIA

- **On-Line Transaction Processing (OLTP)**
  - Many short transactions (queries + updates)
  - Examples:
    - Update account balance
    - Enroll in course
    - Add book to shopping cart
  - Queries touch small amounts of data (e.g. a few records)
  - Updates are frequent
  - Concurrency is biggest performance concern

- **On-Line Analytical Processing (OLAP)**
  - Long transactions, complex queries
  - Examples:
    - Report total sales for each department in each month
    - Identify top-selling books
    - Count classes with < 10 students
  - Queries touch large amounts of data
  - Updates are infrequent
  - Individual queries can require lots of resources

Why OLAP & OLTP don't mix (1)

THE UNIVERSITY OF WESTERN AUSTRALIA

- **Transaction processing (OLTP):**
  - Fast response time important (< 1 second)
  - Data must be up-to-date, consistent at all times
- **Data analysis (OLAP):**
  - Queries can consume lots of resources
  - Can saturate CPUs and disk bandwidth
  - Operating on static "snapshot" of data usually OK
- **OLAP can "crowd out" OLTP transactions**
  - Transactions are slow → unhappy users
- **Example:**
  - Analysis query asks for sum of all sales
  - Acquires lock on sales table for consistency
  - New sales transaction is blocked

Different performance requirements

Why OLAP & OLTP don't mix (2)

THE UNIVERSITY OF WESTERN AUSTRALIA

- **Transaction processing (OLTP):**
  - Normalised schema for consistency
  - Complex data models, many tables
  - Limited number of standardised queries and updates

- **Data analysis (OLAP):**
  - Simplicity of data model is important
    - Allow semi-technical users to formulate ad hoc queries
  - De-normalised schemas are common
    - Fewer joins → improved query performance
    - Fewer tables → schema is easier to understand

Different data modeling requirements

Why OLAP & OLTP don't mix (3)

THE UNIVERSITY OF WESTERN AUSTRALIA

- **An OLTP system targets one specific process**
  - For example: ordering from an online store
- **OLAP integrates data from different processes**
  - Combine sales, inventory, and purchasing data
  - Analyse experiments conducted by different labs
- **OLAP often makes use of historical data**
  - Identify long-term patterns
  - Notice changes in behavior over time
- **Terminology, schemas vary across data sources**
  - Integrating data from disparate sources is a challenge

Analysis requires data from many sources

Why separate data warehouse?

THE UNIVERSITY OF WESTERN AUSTRALIA

- Doing OLTP and OLAP in the same database system is often impractical
  - Different performance requirements
  - Different data modelling requirements
  - Analysis queries require data from many sources

- Different functions and different data:
  - missing data: Decision support requires historical data which operational DBs do not typically maintain
  - data consolidation: DW requires consolidation (aggregation, summarisation) of data from heterogeneous sources
  - data quality: different sources typically use inconsistent data representations, codes and formats which have to be reconciled

Comparison of OLTP and OLAP

THE UNIVERSITY OF WESTERN AUSTRALIA

Table 4.1 Comparison of OLTP and OLAP Systems

Feature	*OLTP*	*OLAP*
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA, database professional	knowledge worker (e.g., manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements decision support
DB design	ER-based, application-oriented	star/snowflake, subject-oriented
Data	current, guaranteed up-to-date	historic, accuracy maintained over time
Summarization	primitive, highly detailed	summarized, consolidated
View	detailed, flat relational	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read
Focus	data in	information out
Operations	index/hash on primary key	lots of scans
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	GB to high-order GB	≥ TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

Note: Table is partially based on Chaudhuri and Dayal [CD97].

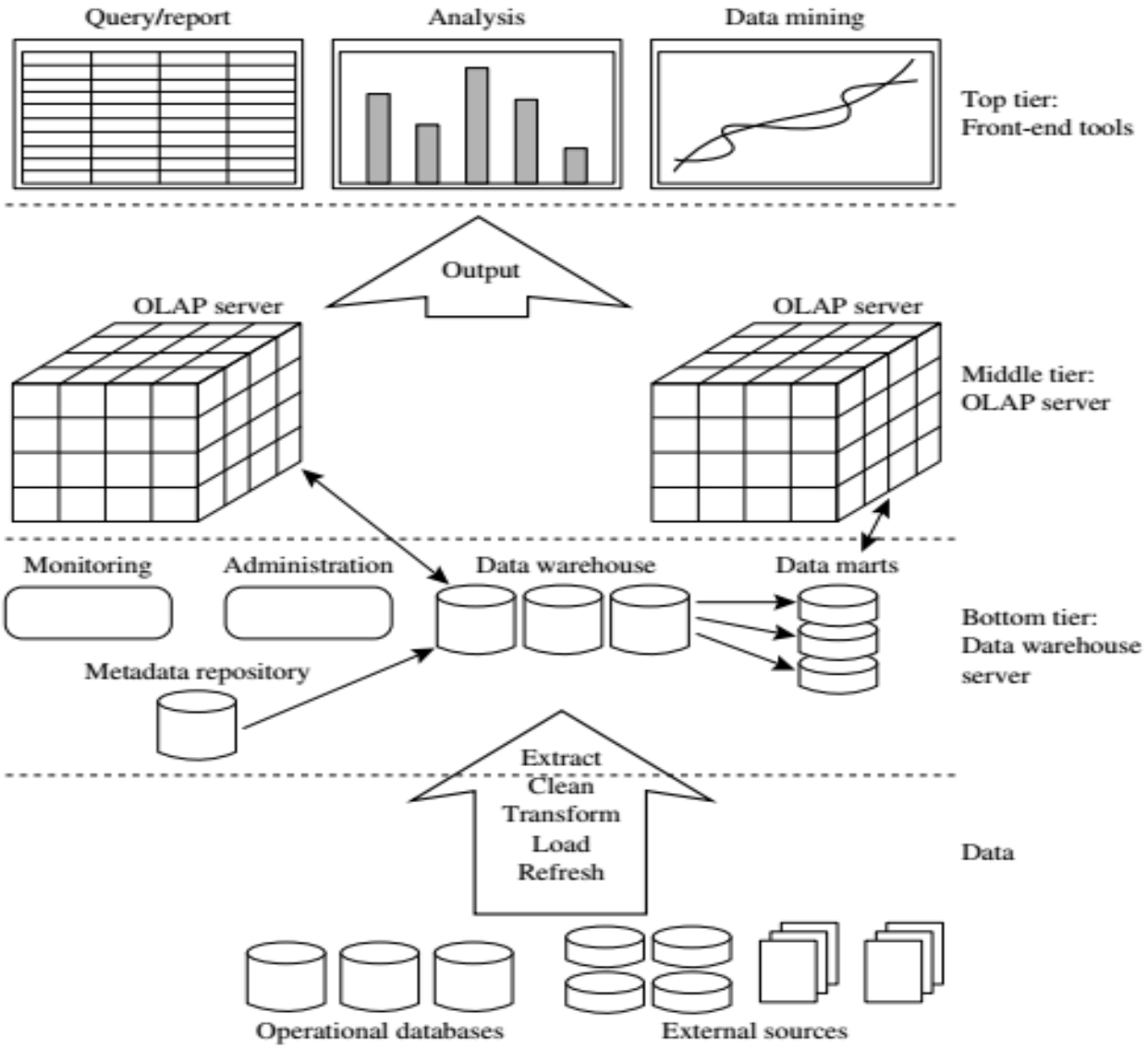
Building a Data Warehouse along with OLTP Systems

THE UNIVERSITY OF WESTERN AUSTRALIA

- Solution: Build a “data warehouse”
  - Copy data from various OLTP systems
  - Optimise data organisation, system tuning for OLAP
  - Transactions aren’t slowed by big analysis queries
  - Periodically refresh the data in the warehouse

- High performance for both systems
  - DBMS— tuned for OLTP: access methods, indexing, concurrency control, recovery
  - Warehouse—tuned for OLAP: complex OLAP queries, multidimensional view, consolidation.

A three-tier data warehousing architecture



Data Warehouse Models

THE UNIVERSITY OF WESTERN AUSTRALIA

- **Enterprise warehouse:**
  - An enterprise warehouse collects all of the information about subjects spanning the entire organisation.
  - Contains both detailed and summerised data

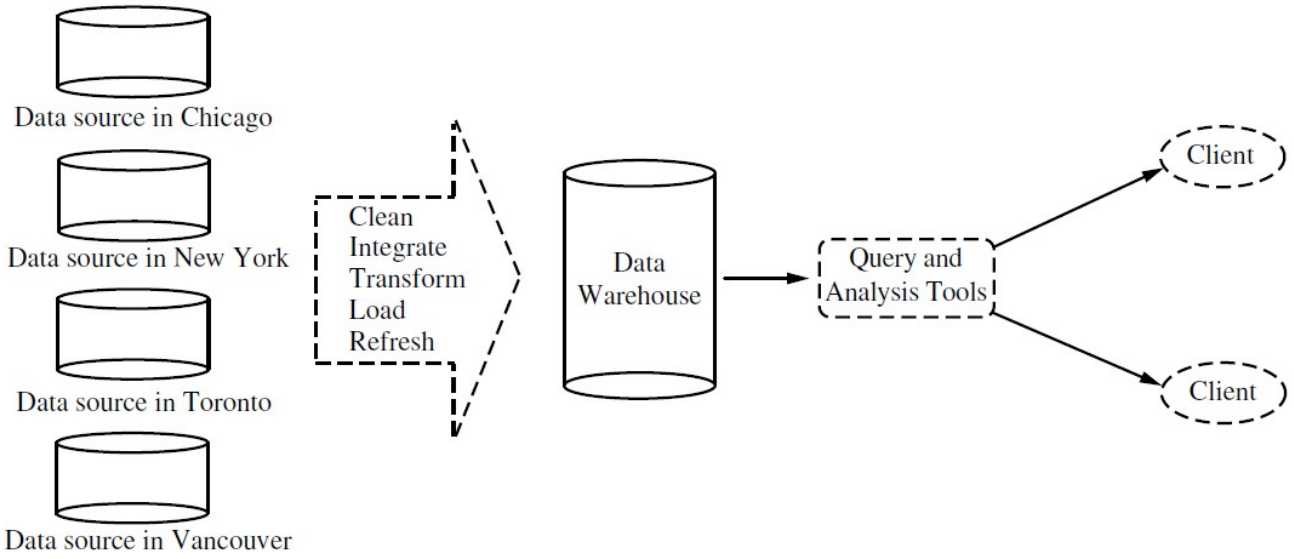
- **Data Mart**
  - A data mart contains a subset of corporate-wide data that is of value to a *specific group of users*.
  - The scope is confined to specific selected subjects, e.g. marketing data mart (customer, item, and sales)
  - Summerised (sometimes due to privacy concerns)

- **Virtual warehouse:**
  - A virtual warehouse is a set of views over operational databases.
  - For efficient query processing, only some of the possible summary views may be materialised.

Data Warehouse of AllElectronics

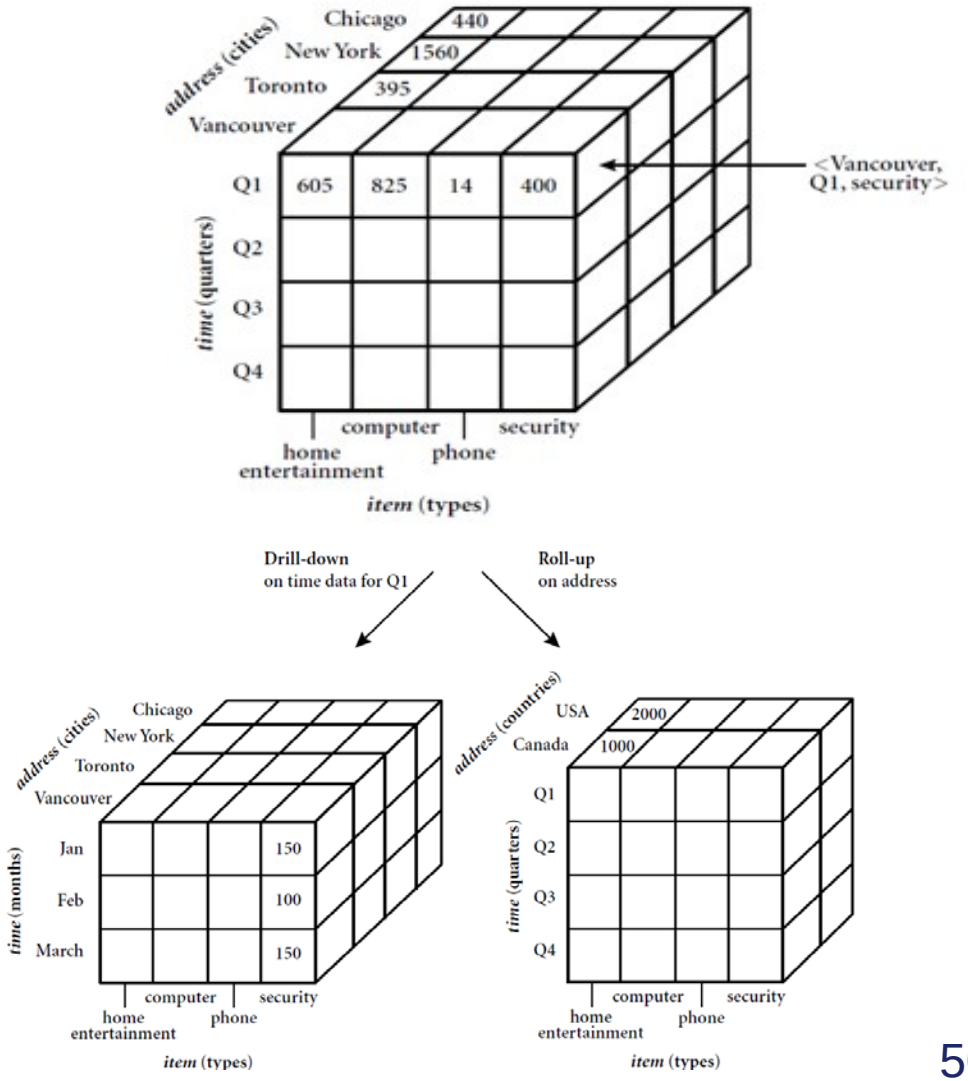
- A data warehouse is a ***repository*** of information collected from multiple sources, stored under a ***unified schema***, and that usually resides at a single site.

- The need is to provide an analysis of the company's sales per item type per branch for a specified period.



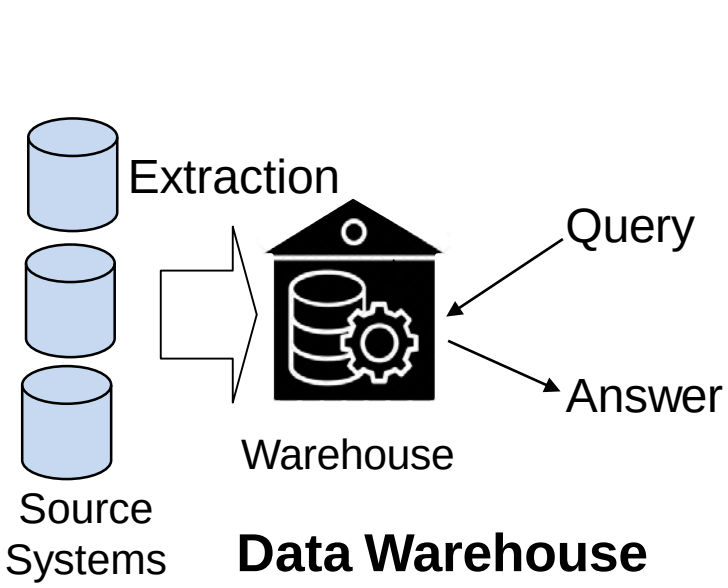
Data Warehouse

- The data warehouse may store a summary of the transactions per item type for each store or, summarised to a higher level, for each sales region.

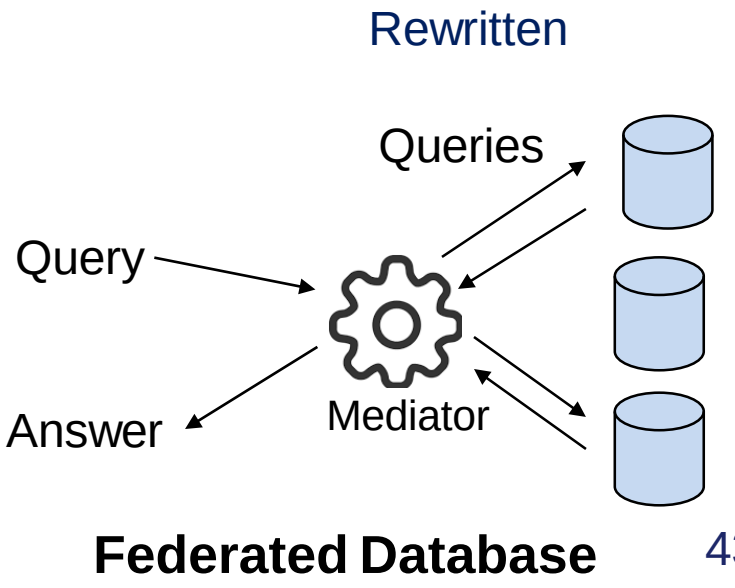


Federated Databases

- An alternative to data warehouses
- Data warehouse
  - Create a copy of all the data
  - Execute queries against the copy
- Federated database
  - Pull data from source systems as needed to answer queries
- “lazy” vs. “eager” data integration



Data Warehouse



Federated Database

Warehouse vs. Federation

THE UNIVERSITY OF WESTERN AUSTRALIA

- Advantages of federated databases:
  - No redundant copying of data
  - Queries see “real-time” view of evolving data
  - More flexible security policy
- Disadvantages of federated databases:
  - Analysis queries place extra load on transactional systems
  - Query optimisation is hard to do well
  - Historical data may not be available
  - Complex “wrappers” needed to mediate between analysis server and source systems
- Data warehouses are much more common in practice
  - Better performance
  - Lower complexity
  - Slightly out-of-date data is acceptable

3 kinds of data warehouse applications

THE UNIVERSITY OF WESTERN AUSTRALIA

- **Information processing**
  - supports querying, basic statistical analysis, and reporting using crosstabs, tables, charts and graphs
- **Analytical processing**
  - multidimensional analysis of data warehouse data
  - supports basic OLAP operations, slice-dice, drilling, pivoting
- **Data mining**
  - knowledge discovery from hidden patterns
  - supports associations, constructing analytical models, performing classification and prediction, and presenting the mining results using visualisation tools.

References

- Some slides are adapted from
  - http://web.stanford.edu/class/cs345/
  - https://hanj.cs.illinois.edu/bk3/bk3_slidesindex.htm

- Readings
  - Chapter 1.1, 1.2 and 4.1 of Han et al.'s book
  - Evolution of Sciences (page 4 to 8).
  - Relational DB v.s. Transational DB

DATA SCIENCE WITH DANIEL

BUILD. SHARE. LEARN.

We seek to build a community of Data Scientists, so that we can share our passion and learn together. We do this by bringing everything and everyone together in one place; Data Science with Daniel.

STUDENTS

We support anyone on their Data Science journey by providing an environment where they can ask questions, find answers and connect with others.

INDUSTRY

We engage with industry to understand the Data Science landscape and ensure that the next generation of Data Scientists develop the skills to succeed in their career.

ACADEMICS

We promote studying Data Science to bring new people into the field and provide feedback to improve these studies.

Join our Slack Community through the blog today!
www.datasciencewithdaniel.com.au