+

Lecture Notes
Week 07

INFS3200 Advanced Database Systems

Semester 1, 2021

Data Integration and Linkage

Professor Xue Li

The University of Queensland Australia

+ Outline

- Database integration
  - Local-Global Schemas Mapping, Data Mapping, Data Fusion
  - Concepts w.r.t. **Global Information Systems**
    - Distributed Databases (DDB)
    - Data Warehouses (DW)
    - Federated databases (FDB)
    - Multi-Databases (MDB)
    - Interoperable Information Systems
  - Issues on different mapping techniques

- Data linkage
  - The problem of data linkage
  - Computing data similarity
  - Applicability of different similarities

+ Other Issues

- Data Integration
  - Combine data from different structured or unstructured data sources
- Information Fusion
  - Extract information from different data sources
- Data Cleaning
  - Remove noise in original data
  - Remove noise in integrated data
    - Inconsistency and redundancy
- Data Quality
  - Data augmentation, data constraints, and data provenance
- Data Privacy
  - Share data with the assurance that "private" information cannot be derived.

+ Data Integration vs. Information Fusion

- Both are designed to integrate and organize data from multiple sources in order to present a unified view of data to derive **actionable insights**.
- Data integration focuses on combining data to create a bigger and consistent data set.
- Information Fusion involves "fusing" data at higher abstraction level and less uncertainty to see a "big picture" of a theme.
- Information Fusion, unlike Data Integration, focuses on deriving insight from real-time streaming data with semantic context from other Big Data sources.
- Most advanced, mission-critical, analytical applications start looking to Information Fusion to add real-time value on data.

https://objectivity.com/information-fusion-and-data-integration-fast-vs-batch/#:~:text=So%2C%20while%20data%20integration%20focuses,narrow%20set%20of%20application%20workloads.

+ Global Information Systems

- The three dimensions:
  - Distribution
  - Homogeneity
  - Autonomy
- The two approaches:
  - Top-down
  - Bottom-up
- What we have discussed so far:
  - Distributed database systems (Top-down)
  - Data warehousing systems (Bottom-up)

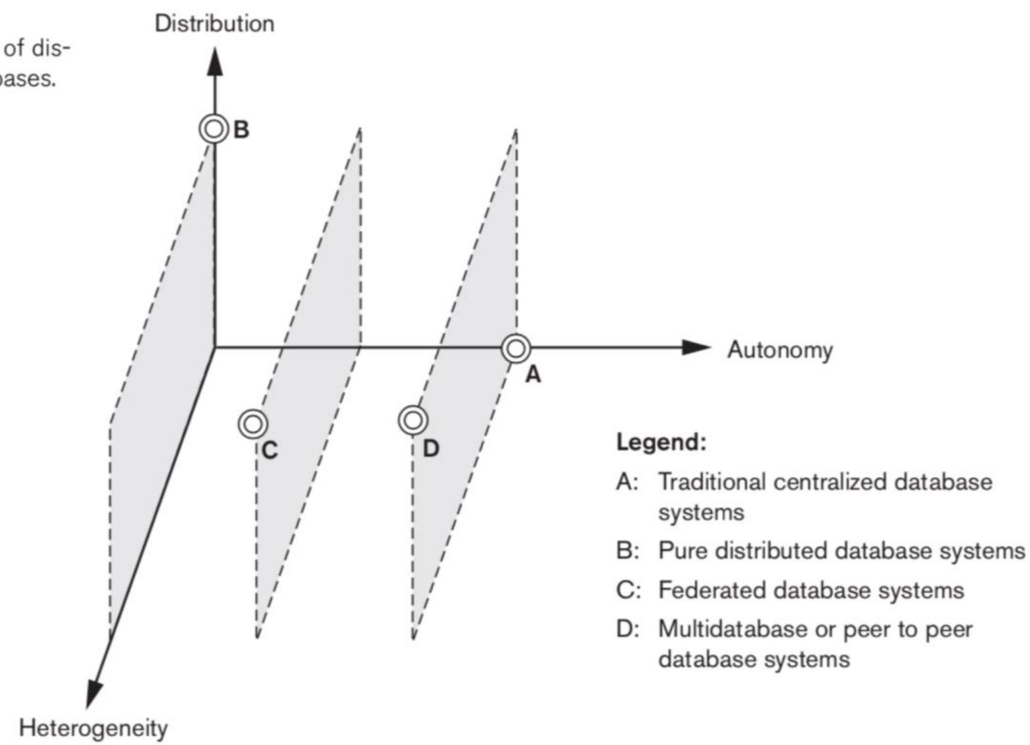
+ Global Information Systems

- Federated databases (FDB)
  - Semi-autonomous database systems, a global view is provided
- Multi-databases (MDB)
  - Autonomous database systems, no or limited global view is provided
- Interoperable information systems
  - No global virtual view
  - Only mechanisms (APIs) provided to communicate with different databases

+ Global Information Systems

	Global Interface	Local node types	Full global DB function?	Integration method
DDB	*Internal DBMS Functions*	*Databases*	*Yes*	*Global Schema*
MDB/FDB	*DBMS User Interface*	*Databases*	*Partial*	*Partial Global Schema*
Interoperable IS	*APIs on top of the DBMS*	*Any data source*	*No*	*No Global Integration*

Classification of distributed databases.



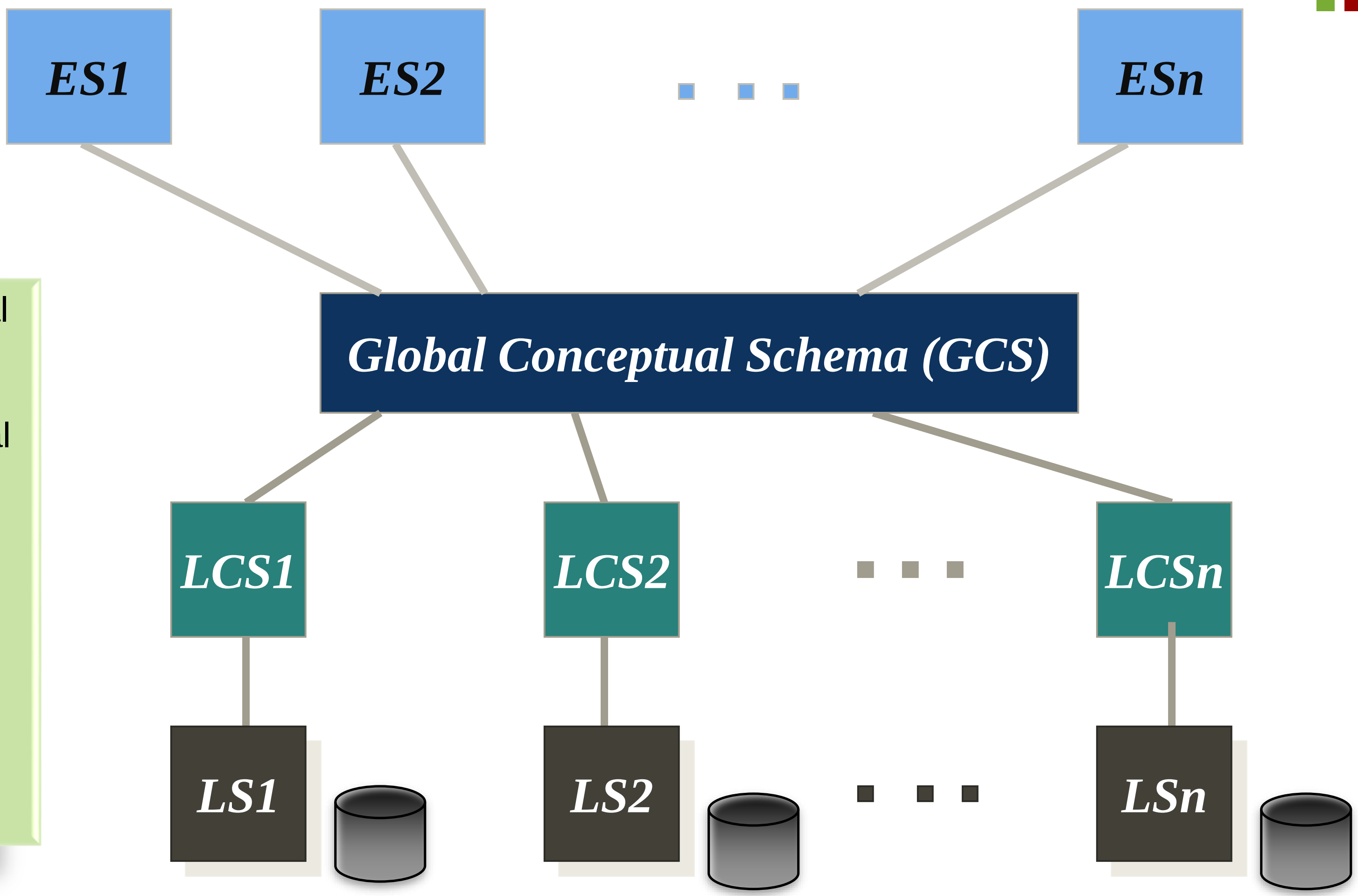
- Different levels of **integration**: DDB, FDB/MDB, to interoperable systems
- **Views** are used as a main mechanism for integration
- Compared with **semantic differences,** system/structure differences are easier to deal with.

P. 896, Elmasri & Navathe, 7th Ed.; p884, , Elmasri & Navathe, 6th Ed.
https://www.brainkart.com/article/Types-of-Distributed-Database-Systems_11591/

+ WiKi Queensland

Youtube Link: https://www.youtube.com/watch?v=NrK2hiH3q9I

+ Federated Databases Illustrated



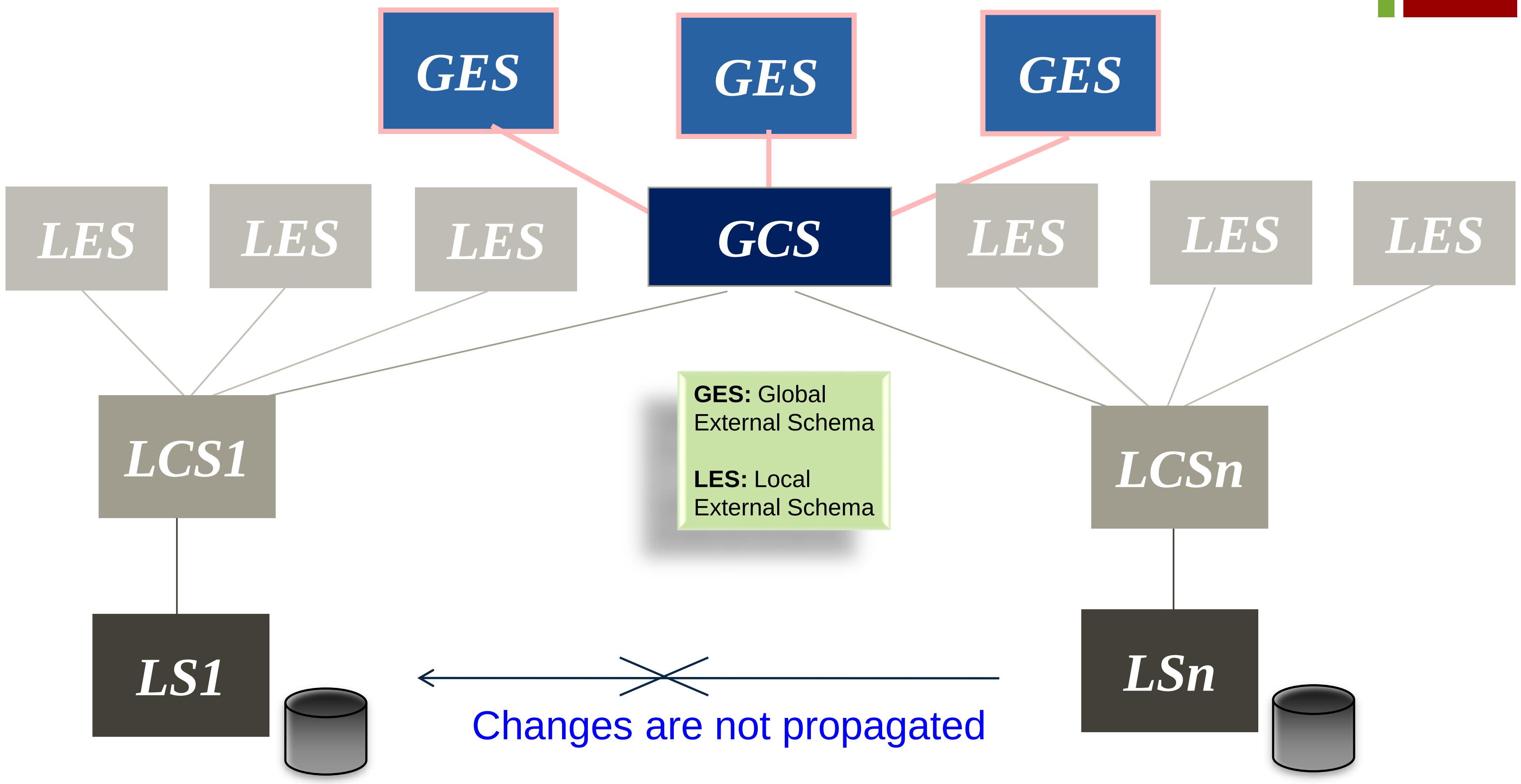
ES: External Schema

GCS: Global Conceptual Scheme

LCS: Local Conceptual Schema

LS: Local Schema

+ Multi-Databases Illustrated



P36, Ozsu & Valduriez: Principles of Distributed Database Systems, 3rd Ed.

+ Interoperable Systems

- Interoperability
  - Ability for an application to access multiple distinct systems
  - Not necessarily for databases only
- Interoperable systems
  - Exchange messages and requests
  - Receive services and operate as a unit towards a common goal
- Different types of interoperability
  - Syntactic: languages and data formats
  - Semantic: meanings
  - System: machines, networks, database systems
  - Structural: data structures and data models

+ In Distributed Databases

- In distributed systems we assume that the entire project is under control of a single organization
- Choices as to fragmentation, replication and tasks for sub-transactions are made on engineering considerations, assuming information availability
  - for allocation of fragments and replicas to sites
  - for system catalog supporting query optimization
  - for resource locking and commit protocols…
- So, building a DDB is a “white box” engineering problem.

+ Now, we have a Different Scenario

- In this learning module, different parts of the system are controlled by different organizations or organizational units
  - Technological boundaries, organizational boundaries and political boundaries
  - These organizations/units are taken to be autonomous
    - No one can tell another what to do
    - No organization/unit is required to expose the internals of their systems, including their system catalogs
- We have a "black box", or possibly "grey box" (e.g., some participants may reveal some information) problem.

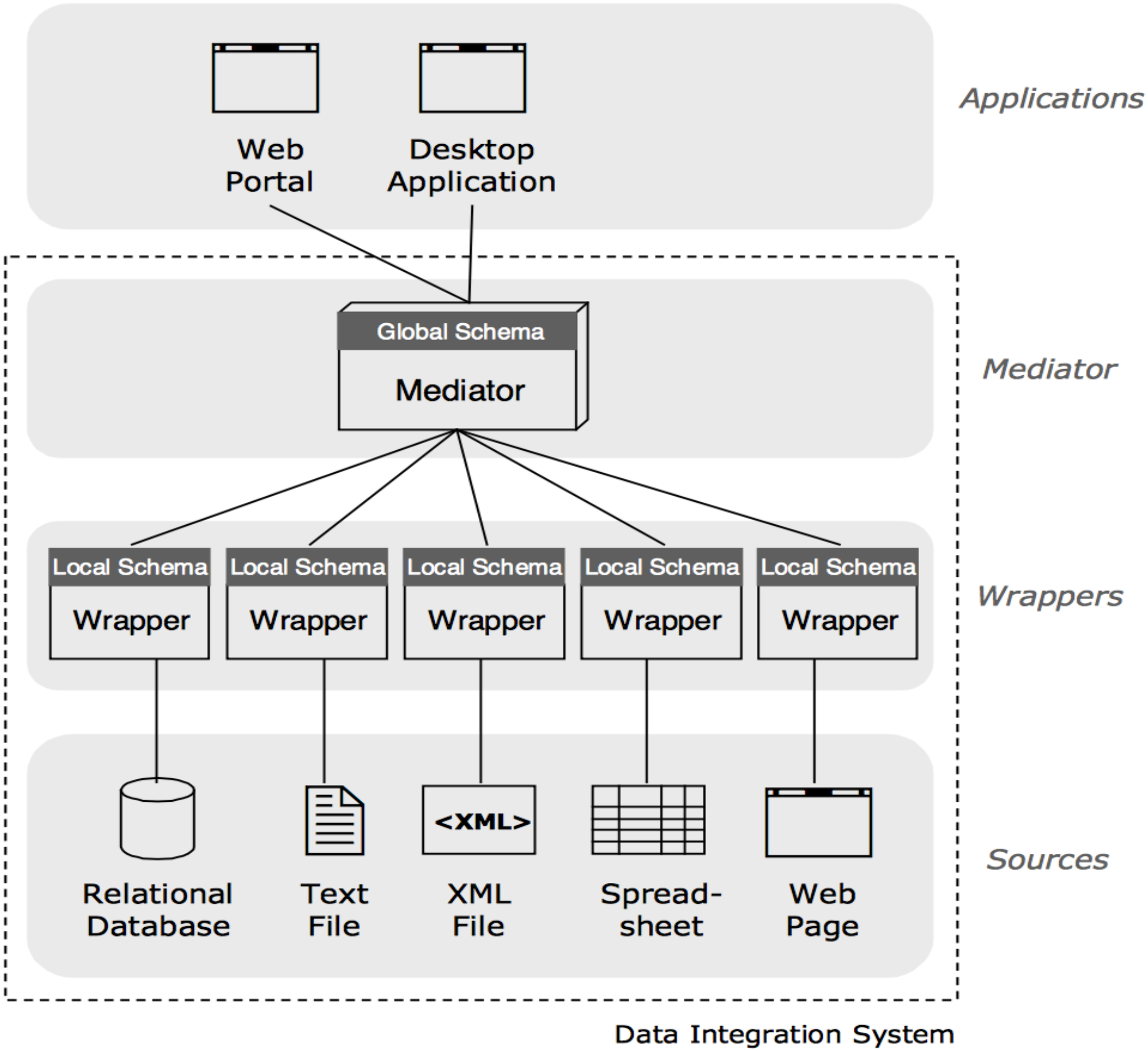
+ Why Database Integration?

- Scenarios:
  - Want to combine databases when two companies merge
  - Want to enhance information using data from different sources
  - Want to access data in legacy databases
- Examples
  - Telstra claims to have over 1,000 information systems
  - Health Connect is an Australian Government initiative intended to integrate hospitals, medical practitioners, pathology laboratories, the Health Insurance Commission, health funds, and more (→ My Health Record)
  - Supply chain management integrates retailers, wholesalers, manufacturers and suppliers
  - E-commerce exchanges allow electronically mediated interaction among many thousands of businesses

+ # Example of data integration: Mediator Wrapper Architecture

- Integrating information over many data sources (e.g. websites), which are dynamically joining and dropping and may have radically different computing platforms
  - Wrapper
    - Exports information about the source schema, and query processing capabilities
  - Mediator
    - Centralizes the information provided by the wrappers in a unified view of all available data (maintaining a GDD)
    - Decomposes user queries, and gathers partial results to compute query results

+ A General Illustration on Mediator Wrapper Architecture



+ Integrated Database Systems

- Building a virtual database system that acts as a front end to multiple local DBs via a bottom-up approach
  - This is different from data warehouses (which is a physical database that is loosely coupled with local DBs)
- The system provides full database functionality and interacts with local systems at their external user interface
  - Local systems maintain their local autonomy
  - The global system provides some means of resolving the differences in data representations
  - A global user can access information from multiple sources with a single relatively simple request , as if they were accessing a single centralized database

+ Challenges in DB Integration

- Each database could be in a different type of DBMS
  - Relational, semi-structured, NoSQL…
- Schema heterogeneity
  - S1: **Employee**(*ID, name, address, position, salary, from, until*)
  - S2: **Worker**(*EID, name, address*); **Position**(*EID, PID, salary, from, until*)
  - S3: **Name**(*EID, address, salary, startingDate*)
- Data type heterogeneity
  - Employee ID could be a string or an integer
- Value heterogeneity
  - The "*cashier*" position could be called "*associate*" in another system
- Semantic heterogeneity
  - Salary is *hourly salary*, or is *weekly salary* with allowances

+ Three Steps for DB Integration

- Schema mapping: mapping of structures
- Data mapping: matching based on content
- Data fusion: reconciliation of mismatching content

We will cover more about data mapping and data fusion in data quality management part

+ Mappings

- Need to have an integrated representation
  - Naming conflicts
  - Format differences (domain, scale, precision…)
    - Local to global transformation can be simple but the inverse can be very complex
  - Structural differences
    - Data value versus attribute
  - Missing or conflicting data
  - Conflicts among constraints

- Examples
  - Schema mapping: e.g. *Name = Title*
  - Domain mapping: e.g. *Integer = String*
  - Value mapping: e.g. *'UK' = 'United Kingdom'*

+ Example of Structural Difference

- Consider two companies data models :

  All records are stored in one table:

  Emp(<u>Emp#</u>, Fname, Lname, Bdate, Dept#, Rank, Salary)

  Another uses one for each department:

  DeptXX(<u>S-id</u>, Fname, Sname, Position, Phone#, e-mail, URL)

- Build an integrated schema

  Employee(<u>EmpID, DeptID</u>, Fname, Lname)

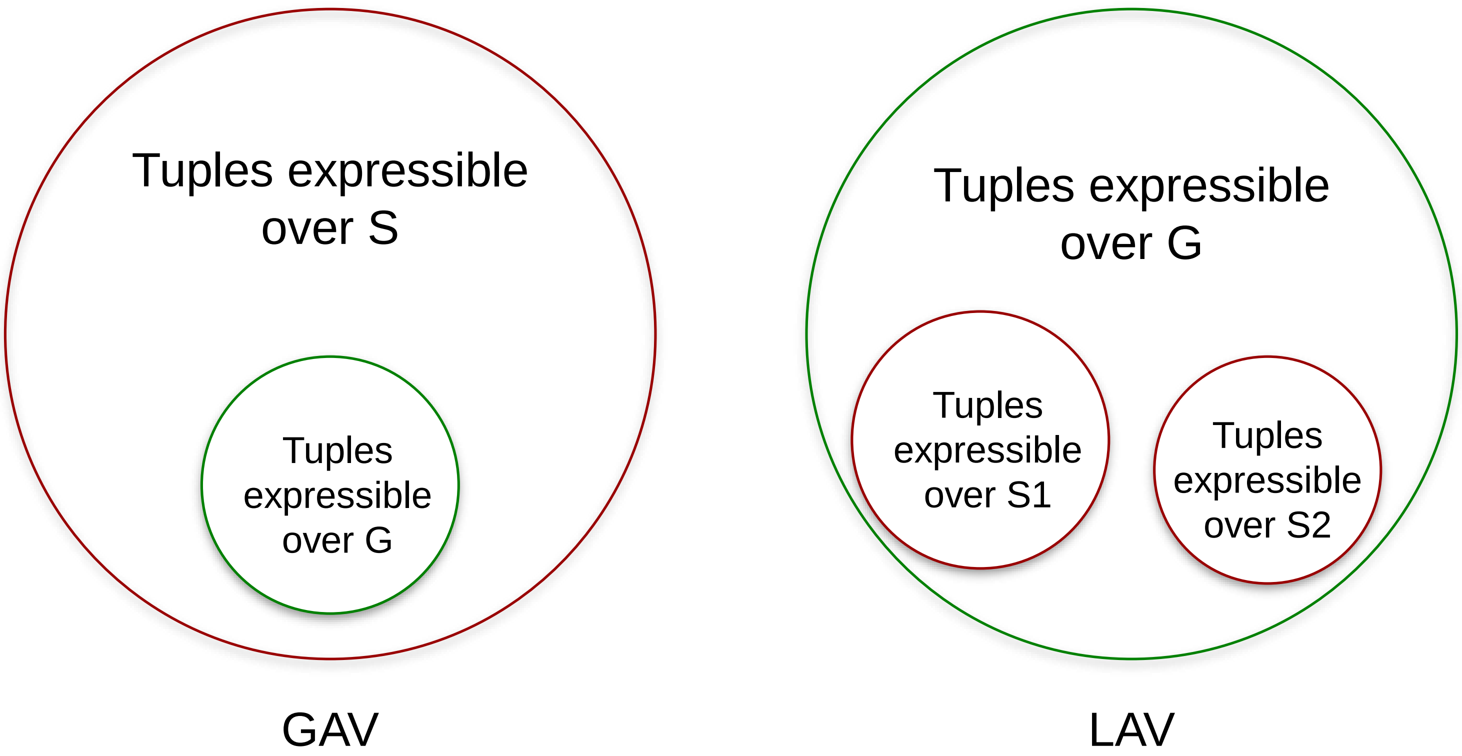
+ View-Based Database Integration

- Problem definition: <*G*, *S*, *M*>
  - *G*: the global schema
  - *S*: a set of local schemas
  - *M*: the mapping to translate queries between *G* and *S*
- A global query is issued over *G* and processed over *S*
- Two popular ways of mapping
  - Global as View (GAV): *G* is a set of views over *S*
    - *M* associates each element in *G* as a query over *S*
    - **Employee.EmpID** ← Emp.Emp# || DeptXX.S-id (G ← S)
  - Local as View (LAV): *S* is a set of views over *G*
    - *M* associates each element in *S* as a query over *G*
    - Emp.Emp# ← **Employee.EmpID** (S ← G)

P.133, Ch4.1, Ozsu & Valduriez: Principles of Distributed Database Systems, 3rd Ed.

A Survey Paper: https://www.inf.unibz.it/~calvanese/papers/calv-lemb-lenz-D2I-D1R5-2001.pdf

+ A Graphical View



- In **GAV**, the set of possible tuples is defined on source *S.* While the set of tuples expressible over the sources *S* may be much larger and richer.
- In **LAV**, the set of possible tuples for each source S_i is defined on G. While the set of tuples expressible over *G* can be much larger (thus, LAV must deal with incomplete answers).

P.134, Ch4.1, Fig.4.2, Ozsu & Valduriez: Principles of Distributed Database Systems, 3rd Ed.

https://www.researchgate.net/publication/2844372_Combining_the_Best_of_Global-as-View_and_Local-as-View_for_Data_Integration

+ Limitations of Views

- Views are for structures, not semantics
  - Views in general are not possible to address data integration problems due to semantic heterogeneity where similar terms could mean different things in different systems
- Semantic heterogeneity is less of a problem where organizations do business together
  - To do business, the organizations must agree on the terms involved
  - Integrated systems require a global schema (ontology) developed before the application, and the participating systems must give up some of their autonomy to commit to the ontology

+ Semantic Issues

Consider this application to have an integrated Student table

- A consortium of universities has a global schema with two tables
  - Student(<u>ID</u>, PublicID, StudentStatus, VisaStatus)
  - Services(<u>StudentStatus, VisaStatus</u>, ServicesApplying)
- A distributed system might have Student table at each university
  - StudentU(<u>ID</u>, PublicID, StudentStatus)
- And a separate server for
  - StudentV(<u>ID</u>, PublicID, VisaStatus)
- A query linking students with services needs to navigate both StudentU and StudentV to get the key for Services

+ The Case of Bottom-Up

- StudentU is operated by a consortium of universities
- StudentV is operated by the Department of Immigration
- Services operated by someone else
- Need to consider
  - Agreement on identification of instances
  - Coverage of instances
  - How these affect queries

+ Agreement on Instance IDs

- StudentU(<u>IDu</u>, PublicIDu, StudentStatus) @Universities
- StudentV(<u>IDv</u>, PublicIDv, VisaStatus) @Immigration Dept
- How do we get association between student and status fields?
  - No reason to suppose IDu and IDv are related
  - PublicIDu could be name and date of birth
  - PublicIDv could be thumbprint
- Join is impossible, *even if* the two systems have information on exactly the same people
- Organizations must agree on IDs and at least one must gather more data and maintain a correspondence between two sets of IDs

+ Coverage of Instances

- Even with an agreement on IDs, in practice the two systems could cover different populations
  - StudentU may include domestic as well as overseas students
  - StudentV may include all sorts of student visa holders, not just university students
- What does it tell us?
  - If a person is linked to both StudentStatus and VisaStatus by this system
  - If a person is linked to one but not the other
- Depends on how reliably the two systems are updated, and on how frequently
  - Need agreements on quality of service (QoS)

+ Ontological Commitment

- Building a multi-database bottom-up by schema and population integration can be problematic
- A global schema can only be created by agreement (ontology)
- Each participant must commit to the ontology
  - Create views, often modify schemas
  - Often introduce global identifiers like ISBN and establish correspondences between local and global identifiers
- This is actually a *top-down* approach, no longer a *bottom-up* approach

An **ontology** is a formal naming and definition of the types, properties, and interrelationships of the entities that really or fundamentally exist for a particular domain of discourse. It is thus a practical application of philosophical **ontology**, with a taxonomy.

+ Semantics/Ontology and Standards

- A very large community of researchers is working on improving understanding of data (schema and value) semantics
  - **Knowledge graph**
- Many influential efforts towards providing standards for data exchange
- Motivation resides in the ability to allow disparate systems to interoperate seamlessly

The goal of the Semantic Web is to make Internet data machine-readable.

For Reference on Semantic Web, see: www.w3.org/DesignIssues/Semantic.html

+ Meta Data – The Binding Force

- Metadata is defined as the data providing information about one or more aspects of the data:
  - Means of creation of the data
  - Purpose of the data
  - Time and location of creation
  - Creator or owner of the data, and data lineage
  - Standards used
- Metadata can be used to address:
  - Schema differences
  - Structural differences
  - Value differences

Data Linkage

+ What is Data Linkage?

- An operation to identify records referring to the same real-world entity
  - a.k.a. data cleaning, data scrubbing, record linking, de-duplication, fuzzy matching, entity resolution, merge/purge, reference reconciliation, data hardening, entity disambiguation…
- An essential pre-step for data integration and data mining
  - For data integration
  - For duplication detection
- Increased interest recently for automatic entity resolution, because of increasing complexity in:
  - Data volumes
  - Data diversity
  - Data usage

+ What Data Linkage is Not

- Data linkage is related but different from:
  - *System heterogeneity* (to link data stored in different systems)
  - *Structural heterogeneity* (e.g., address vs (street#, street_name, city…)
  - *Mirror detection* (e.g., near-duplicate docs or web pages)
  - *Co-reference resolution* (e.g., *I graduated from the University of Queensland. It is one of the best universities in the world*)
- Typically, semantic issues are not considered (too hard)
  - E.g., “oz” = “Australian”
- Records with a well-defined ID are not considered (too easy)

+ Application: Data Integration

- Company acquisition and merge
- From organizational-unit-focused information systems to customer-focused information systems
- The whole-of-X approach
  - X: government, health care, water systems…
- Data mining and data warehousing

ID	Given_name	Surname	DOB	Gender	Address	Loan_type	Balance
6723	peter	robert	20.06.72	M	16 Main Street 2617	Mortgage	230,000
8345	smith	roberts	11.10.79	M	645 Reader Ave 2602	Personal	8,100
9241	amelia	millar	06.01.74	F	49E Applecross Rd 2415	Mortgage	320,750

Bank database

Health database

PID	Last_name	First_name	Age	Address	Sex	Pressure	Stress	Reason
P1209	roberts	peter	41	16 Main St 2617	m	140/90	high	chest pain
P4204	miller	amelia	39	49 Aplecross Road 2415	f	120/80	high	headache
P4894	sieman	jeff	30	123 Norcross Blvd 2602	m	110/80	normal	checkup

+ Application: Duplicate Detection

- 90% data cleaning work is associated with de-duplication when archiving data by inputting files to data warehouse
  - [National security] [www.dailykos.com/story/2006/1/5/85158/32663]: Some innocent people included in “No Fly Watch List”
  - [Statistics]: One patient has several diagnosis records
  - [Marketing]: One customer has several records - sending extra catalogues

[Win06] Overview of record linkage and current research directions, W. E. Winkler, Research report series (Statistics #2006-2), Statistical Research Division, U.S. Census Bureau.

+ The Causes of Differences

- Typological errors
- Missing or uncertain values
- Phonetic issues
- Numeric issues
- Inconsistent abbreviations
- Some ‘natural’ causes:
  - Context-related variations
  - Dynamic nature of data (variations over time, regions, disciplines)
- …

+ Why Difficult?

- Beyond string similarity
  - The same real-world object can be represented as different strings
  - The same string can represent different real-world objects
- Quadratic complexity to data sizes
  - With very high complexity for ‘unit’ operations
- Often no black-and-white answers – probabilistic answers
- Need to consider privacy issues too

+ Some Examples

- Now let's see some examples, to have a better understanding of the problems we are going to address

Name	School
Smith, William	The School of Info. Tech. and Elec. Eng.
W. A. Smith	ITEE
Bill Smith	School of ITEE
Wiliam Smith	ITEE
Bill Smyth	Department of Computer Science

Name	Diagnosis	Address	Age
John Williams	Skin Cancer	9 Hamptons Blvd NW Calgary, AB T3A 5S2	50
John Williams	Skin Cancer	130 Savannah Dr Moncton, NB E1A 6T7	55
John Williams	Diabetes	130 Savannah Dr Moncton, NB E1A 6T7	55

+ Linking with Different Granularity

- Data matching methods
  - Field-level: for two given attributes, to decide if they are identical (e.g., two names, or two addresses)
    - This is the most basic form the entity linking
  - Record-level: for two database records, to check if they are about the same entity (a.k.a data augmentation)
    - With more contextual information the linking accuracy can be improved
  - Group-level: to check if two groups of records are about the same composite entity (e.g., two families with each record for one family member)
    - One-to-one mapping within the composite entity can help to improve linking accuracy

+ Field Matching

- Find similarity for two given text strings
  - A basic operation for more complex similarity measures
- Main problem to address: typographical issues
  - Spelling Errors, e.g. “Jhn” instead of “John”
  - Incorrect Input, e.g. “AIMS Bank” instead of “AIMS Finance”

[NAD04] Flexible string matching against large databases in practice (2004). N. Koudas, A. Marathe and D. Srivastava

+ Similarity by Edit Distance

- The edit distance between two strings is the minimum number of operations to transform one string to another
  - Operations: delete, insert or substitute one character
- What's the edit distance?
  - 'John', 'Jon'
  - 'John' , 'Jhn'
  - 'John', 'Josh'

There are many types of edit distance, the one with insertion, deletion, substitution is called Levenshtein distance. Other well-known edit distance types include Hamming distance, Longest Common Subsequence (LCS) distance.

+ Using Edit Distance

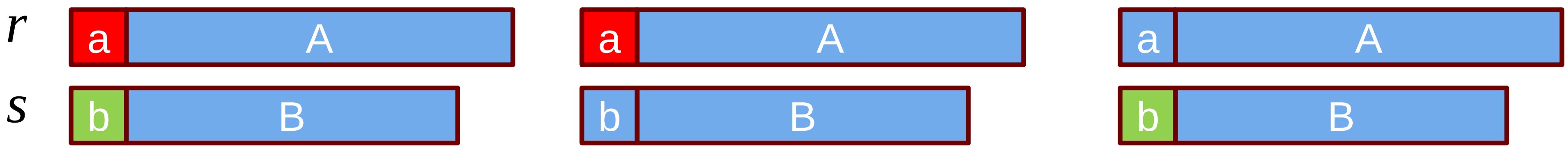
- Two strings are considered identical if their ED is less than a pre-defined threshold
  - Normalization is recommended

$$sim(a,b) = 1 - \frac{ED(a,b)}{\max(|a|,|b|)}$$

- Usually asked questions about a distance:
  - When is *ED(a, b)=0*?
  - *ED(a, b)* vs. *ED(b, a)*?
  - *ED(a, c)* vs. (*ED(a, b) + ED(b, c))*?
  - *|a| = m* and *|b| = n*, what are the min and max *ED(a, b)*?
  - How to compute ED for two given strings?
  - How this can be done for two large data sets?

+ Edit Distance Computing

- Dynamic programming algorithm
  - Solve the current problem with the known results of sub-problems
  - Keep reducing the problem size recursively



d(A, B)+1 d(A,"b"+B)+1 d("a"+A, B)+1

r = "a" + "A"
s = "b" + "B"

- Equation

$$ED(r,s)=\begin{cases} n, & m=0 \\ m, & n=0 \\ \min\begin{Bmatrix} ED\big(sub(r), sub(s)\big)+subcost, \\ ED\big(sub(r), s\big)+1, \\ ED\big(r, sub(s)\big)+1 \end{Bmatrix}, & otherwise \end{cases}$$

$$subcost=\begin{cases} 0, & head(r)=head(s) \\ 1, & otherwise \end{cases}$$

+ An Example

- Use (|r|+1)×(|s|+1) matrix E
- Start from E[0,0], and **E[| r |,| s |] is the edit distance**
- E[i,j] = *[i-1, j-1]* if $r_i = s_j$, or

  min*([i, j-1]+1, [i-1, j-1]+1,*

  [i-1, j]+1) if $r_i \neq s_j$
- Complexity is O(|r|×|s|)

j=1, 2, 3, 4

i=1, 2, 3

		j	**o**	**h**	**n**
	0	**1**	**2**	**3**	**4**
j	**1**	**0**	**1**	**2**	**3**
h	**2**	**1**	**1**	**1**	**2**
n	**3**	**2**	**2**	**2**	**1**

Result

G. Navarro, “A Guided Tour to Approximate String Matching,” ACM Computing Survey 2001

+ Another Example

- ED("Dubios", "Dubose") ?
  - E[i,j] = [i-1, j-1] **if $r_i = s_j$**
  - E[i,j] = min([i, j-1]+1, [i-1, j-1]+1, [i-1, j]+1) **if $r_i \neq s_j$**

		D	U	B	O	S	E
	0	1	2	3	4	5	6
D	1	0	1	2	3	4	5
U	2	1	0	1	2	3	4
B	3	2	1	0	1	2	3
I	4	3	2	1	1	2	3
O	5	4	3	2	1	2	3
S	6	5	4	3	2	1	2

+ Comments on Edit Distance

- The ED discussed so far is known as Levenshtein Metric (1965)
- Observations
  - A costly operation for large strings
  - Suitable for common typing mistakes
    - “Comprehensive” vs “Comprenhensive”
  - Problematic for specific domains or abbreviations
    - “AT&T Corporation” vs “AT&T Corp”
    - “IBM Corporation” vs “AT&T Corporation”
    - “ITEE” vs “IEEE” vs “School of Information Technology and Electrical Engineering”

+ Similarity by Tokenization (q-grams)

- Varying semantics of 'term'
  - Words in a field
    - 'AT&T Corporation' -> 'AT&T' , 'Corporation'
  - q-grams (sequence of q-characters in a field, a.k.a. *n-grams*)
    - *{'AT&','T&T','&T_', 'T_C','_Co', 'Cor','orp','rpo','por','ora','rat','ati','tio','ion'}* **3-grams**
    - Can add '##A', '#AT' and 'on#', 'n##' to the set for the ends of sequence
- Calculate similarity by manipulating **sets** of terms
- Question
  - For a string of *n* characters, how many *q*-grams does it contain?

+ q-Gram and Jaccard Coefficient

- Idea: if two strings share many q-grams, they can be considered as similar
- Given two sets of terms S, T
  - Jaccard coefficient: **Jaccard(S,T) = |S∩T|/|S∪T|**
- A common technique used in language processing
  - Text recognition, spelling checking
  - Insensitive to word orders: “**University of Queensland**” vs “**Queensland University**”
  - Can be computationally efficient
- Problem
  - “School of ITEE” vs. “ITEE” vs. “School of Art”

+ Similarity Measure with TF/IDF

- Term frequency (**tf**) inverse document frequency (**idf**)
  - tf : # of times 'term' appears in a document
  - idf : number of documents (N) / number of documents containing 'term' (n)
  - Term score: tf*idf
- Widely used in traditional IR (Information Retrival) approaches
  - Intuitively: a term which appears often 'locally', but rare 'globally' is more important

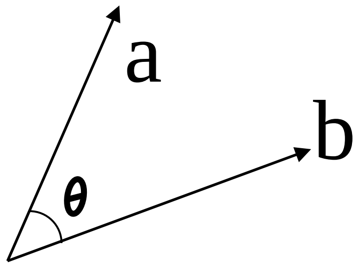
$$weight_{string}(token) = \log(tf_{token} + 1) * \log(idf_{token})$$

  - "School of ITEE": w_1 * "school", w_2 * "of", w_3 * "ITEE"
    - $w_3 > w_1 > w_2$

[Ww98] Integration of Heterogeneous databases without common domains using queries based on textual similarity, sigmod 1998

+ Cosine Similarity

- Each field value is transformed via **tf/idf** weighting to a vector in *d* high dimensional
- Let *a*,*b* be two field values and *Va*, *Vb* be the set of tf/idf scores for terms in *a* and *b*



$$sim(a,b) = \frac{\sum_{i=1..d} V_a(i) \bullet V_b(i)}{\| V_a \|_2 \bullet \| V_b \|_2}$$

+ Numeric Similarity

- Numbers with similar values but presentations are dissimilar
  - For example: 500 is similar to 499

- Two straightforward ways
  - Treat a number as a word
    - Similarity of two numbers is measured by edit distance
    - Problem: e.g. 500 versus {499, 800}
  - Numbers *a*, *b* are similar if *a*, *b* in a range *T*, that is, $|a-b| < T$
    - Problem: what is a proper range?

[NAD04] Flexible string matching against large databases in practise (2004). N. Koudas, A. Marathe and D. Srivastava

+ Phonetic Similarity

- A phonetic algorithm that indexes names by their sounds when pronounced in English
- Soundex consists of the first letter of the name followed by three numbers. Numbers encode similar sounding consonants
  - Retain the first letter and drop all other occurrences of a, e, i, o, u, y, h, w (all vowels + w, h)
  - Replace consonants with digits as follows (after the first letter):

    b, f, p, v → 1   c, g, j, k, q, s, x, z → 2   d, t → 3

    l → 4   m, n → 5   r → 6
  - **Concatenate first letter of string with first 3 numerals**
- Exp1: “great” and “grate” become G6EA3 and G6A3E and then G630
- Exp2: “Robert” and “Rupert” both become R163

[Rus18] US1,261,167 (PDF version) (1918-04-02) R. C. Russell

+ Record Matching (3-1)

- Multi-attribute similarity measures
  - Q: combine all attributes together into a long string?
  - Problems with string concatenation

Name	Address
RAM Finance	16, Finance St, South Bank, QLD

"RAM Finance 16, Finance St, South Bank, QLD"

- Finance is universal in *name* field, less discrimination power
- Finance is rare in *address* field, more discrimination power

+ Record Matching (3-2)

- Weighted-Sum
- Measure the distance between individual fields, and then compute the weighted distance between records.
  - Static Weight
    - e.g. sim = sim(name)* w + sim(address)* (1-w)
    - Easy to implement but a good value of ***w*** is not obvious
  - Dynamic Weight [NAD04]
    - Give more weight to the longer field so as to unify the influence of field length to the similarity

[NAD04] Flexible string matching against large databases in practise (2004). N. Koudas, A. Marathe and D. Srivastava

+ Record Matching (3-3)

- Rule-Based Approaches
- Equation between records can be inferred by specified equation theory
  - Equation theory dictates the logic of domain equivalence, not simple value or string equivalence [HS95]

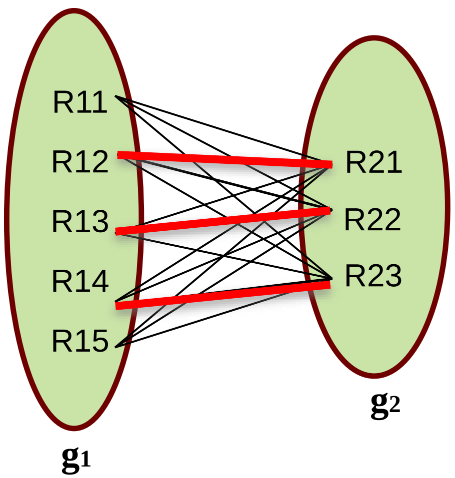
Given two records, r_1 and r_2
IF the ***last name*** of r_1 equals the ***last name*** of r_2,
AND the ***first names*** differ slightly,
THEN
$r_1 = r_2$

[HS95] M.A. Hernandez and S.J. Stolfo, The merge/purge problem for large databases, Sigmod 1995

+ Group Matching

- Discover two groups of records referring to same entity, e.g. a family
  - For two groups (g_1, g_2), ($|g_1| = m_1$, $|g_2| = m_2$), similarity between all pairs of records are computed as *BM*
  - Construct a bipartite graph (**$|M|$ = *number of matched pairs***)
  - Find the maximum weight matching from the graph
    - 1:1 matching between records in two groups (***sim(r_{1i}, r_{2j})***)
    - Summation of similarity is maximized

$$BM = \frac{\sum_{(r_{1i}, r_{2j} \in M)} (sim(r_{1i}, r_{2j}))}{m_1 + m_2 - |M|}$$



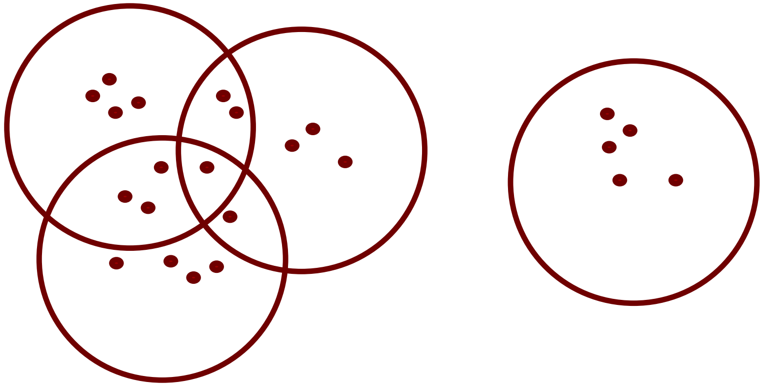
[BND07] B.W. On, N. Koudas, D. Lee, D. Srivastava, Group Linkage, ICDE 2007

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4221698

+ Canopies

- Two steps:
  - **Step 1:** Data are divided (by inexpensive distance measurement) into overlapping subsets, called canopies
    - *E.g., Patients with a common diagnosis fall in same canopy.*
    - *E.g., Publications with one same author fall in same canopy*
    - *E.g., Using inverted index in a search engine: two documents having a certain number of common words fall in same canopy.*
  - **Step 2:** Expensive distance measurement made among points within a same canopy.

Points not appearing in any common canopy are not possibly be in the same cluster.



An inverted index is a sparse matrix representation in which, for each word, we can directly access the list of documents containing that word.

[MKL00] MA. McCallum, K. Nigam, L. H. Ungar, Efficient clustering of high-dimensional data sets with application to reference matching, KDD (2000)
https://dl.acm.org/doi/pdf/10.1145/347090.347123

+ Summary

- Database integration
  - Top-down vs bottom-up approaches
  - DDB/DW, to FDB, MDB and interoperable systems
  - Views are extensively used in DB integration
- Data linkage
  - The problem of data linkage (i.e., computing the same real-world entity)
  - Different types of data similarity measures, including
    - Edit distance, Jaccard coefficient, and cosine similarity
- Both DB integration and data linkage can be application-dependent, thus no off-the-shelf solutions

Next week: Data Quality

+ Reading Materials

- Ch1.7, Ch4 & Ch9**,** Ozsu & Valduriez: Principles of Distributed Database Systems, 3rd Ed.
- A PhD Thesis **Data Integration against Multiple Evolving Autonomous Schemata**: https://cds.cern.ch/record/1387966/files/CERN-THESIS-2001-036.pdf
- Ch25: Distributed Database Systems, Elmasi & Navathe, 6th Ed.
- Ch23: Distributed databases, NOSQL Systems, and Big Data, Elmasi & Navathe, 7th Ed.
- A Survey Paper: https://www.inf.unibz.it/~calvanese/papers/calv-lemb-lenz-D2I-D1R5-2001.pdf
- Additional readings:
  - Phil Bernstein and Laura Haas, "Information Integration in the Enterprise", *Communications of the ACM* 2008
  - Ahmed K. Elmagarmid et al, "Duplicate Record Detection: A Survey", *IEEE Transactions on Knowledge and Data Engineering,* 2007

Elmasri & Navathe 7th Ed., https://www.auhd.site/upfiles/elibrary/Azal2020-01-22-12-28-11-76901.pdf

Elmasri & Navathe 6th Ed. https://seu1.org/files/level6/IT344/Fundamentals_of_Database_Systems,_6th_Edition.pdf

Ozsu & Valduriez book: https://link.springer.com/book/10.1007/978-1-4419-8834-8