Data Integration:
Querying Heterogeneous Information Sources Using Source Descriptions & Data Integration: The Teenage Years

CPSC 534P

Rachel Pottinger

September 19, 2011

Administrative Notes

- Homework 1 due… now
- I'll get grades on your first reviews back ASAP
  - If you got a 2, it almost certainly means that you need more analysis/synthesis
  - Try to ask more questions that you think would be good for discussion (partially my fault)
  - It's good to think of weaknesses, but remember that your work's not perfect either
- Project proposals due next Monday

For today’s class, I’ll wear three hats

- The presenter’s hat
- The discussion leader’s hat
- The “me” hat

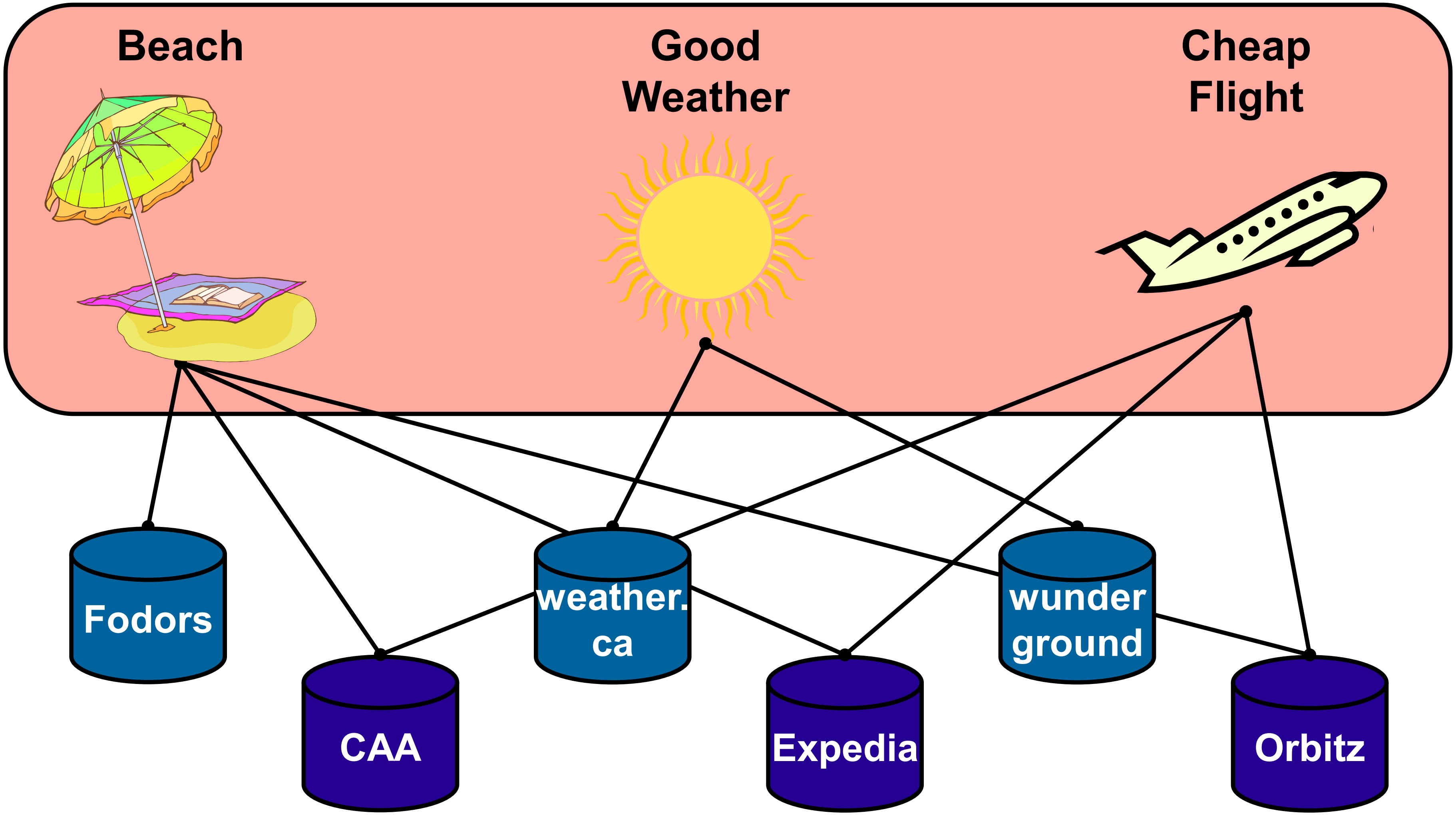
I’ll try to make it clear which is which, but if you get confused, let me know

Data Integration

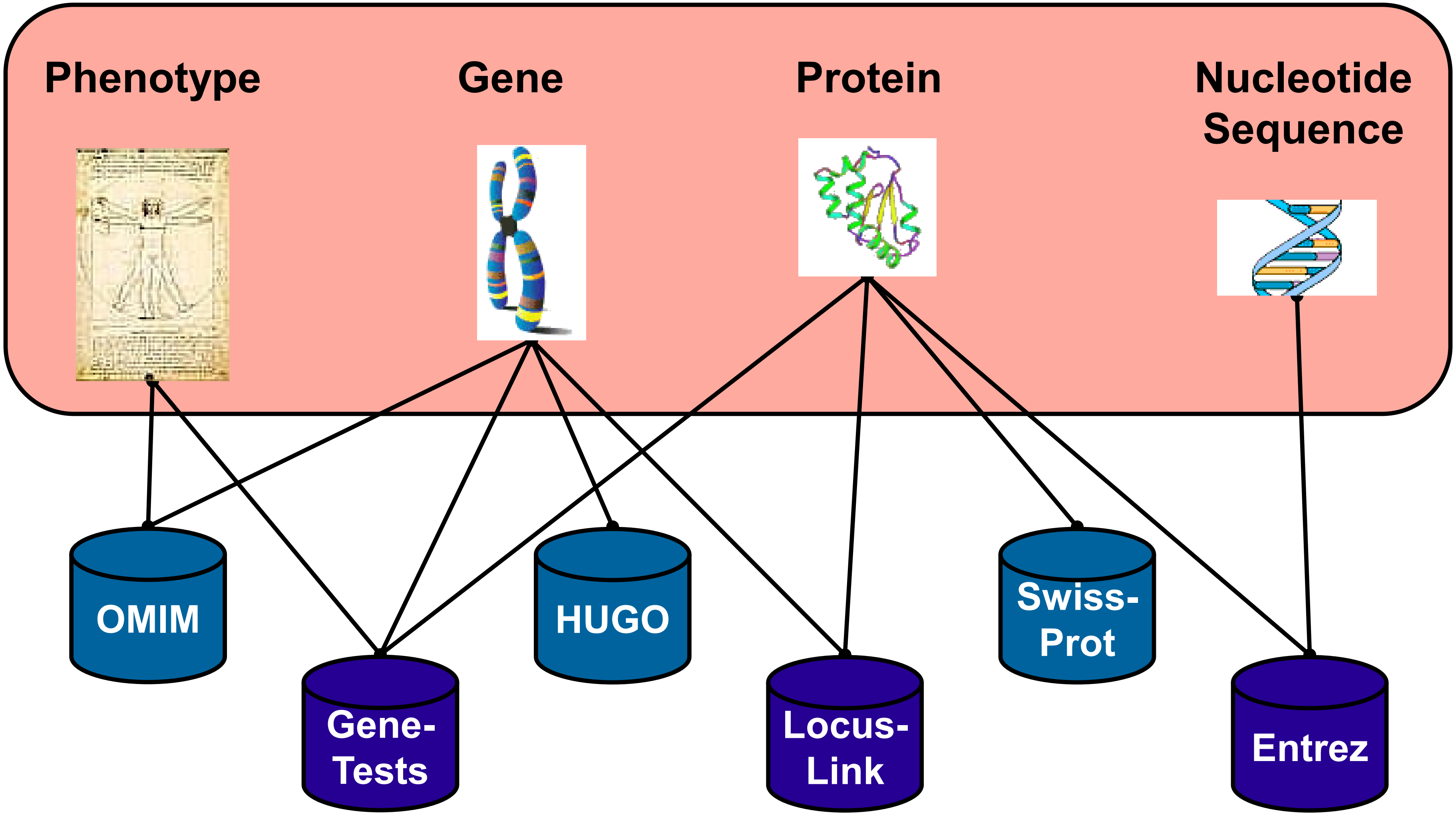
- Up until now: one database – one schema
- Queries programmed by experts

  General users issue pre-programmed queries
- Interaction between databases
  - Not very common
  - Extremely manually intensive to set up
    - Expensive
    - Time consuming
    - Hard to change

Planning a Beach Vacation



BioMedical Research



Phenotype

Gene

Protein

Nucleotide Sequence

OMIM

Gene-Tests

HUGO

Locus-Link

Swiss-Prot

Entrez

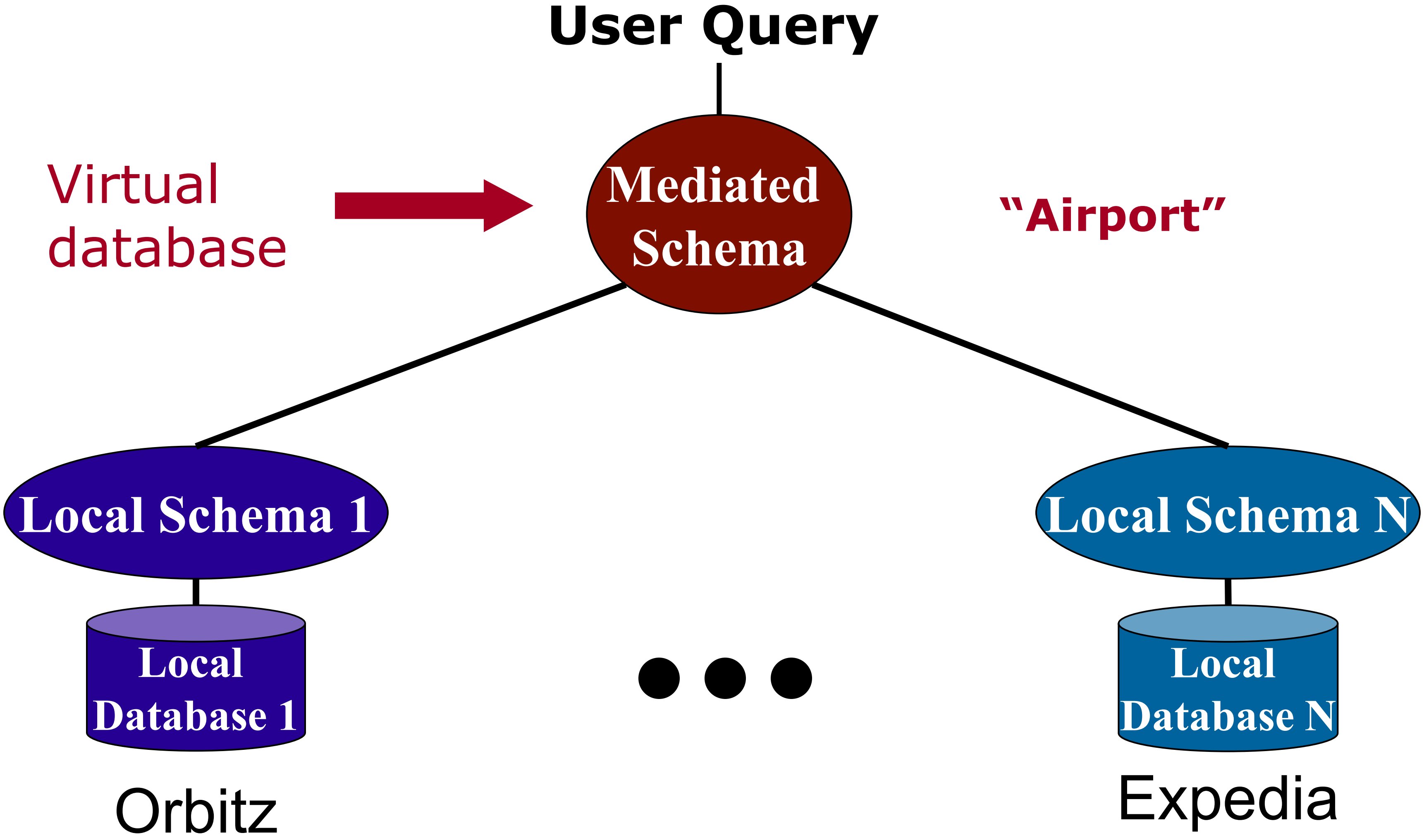
Modern Data Management

- Many overlapping databases
- Vast user base
- Users want data from multiple sources

Users want to combine data from many databases without knowing where it comes from

The catch? They all have different *schemas*

Data Integration Systems



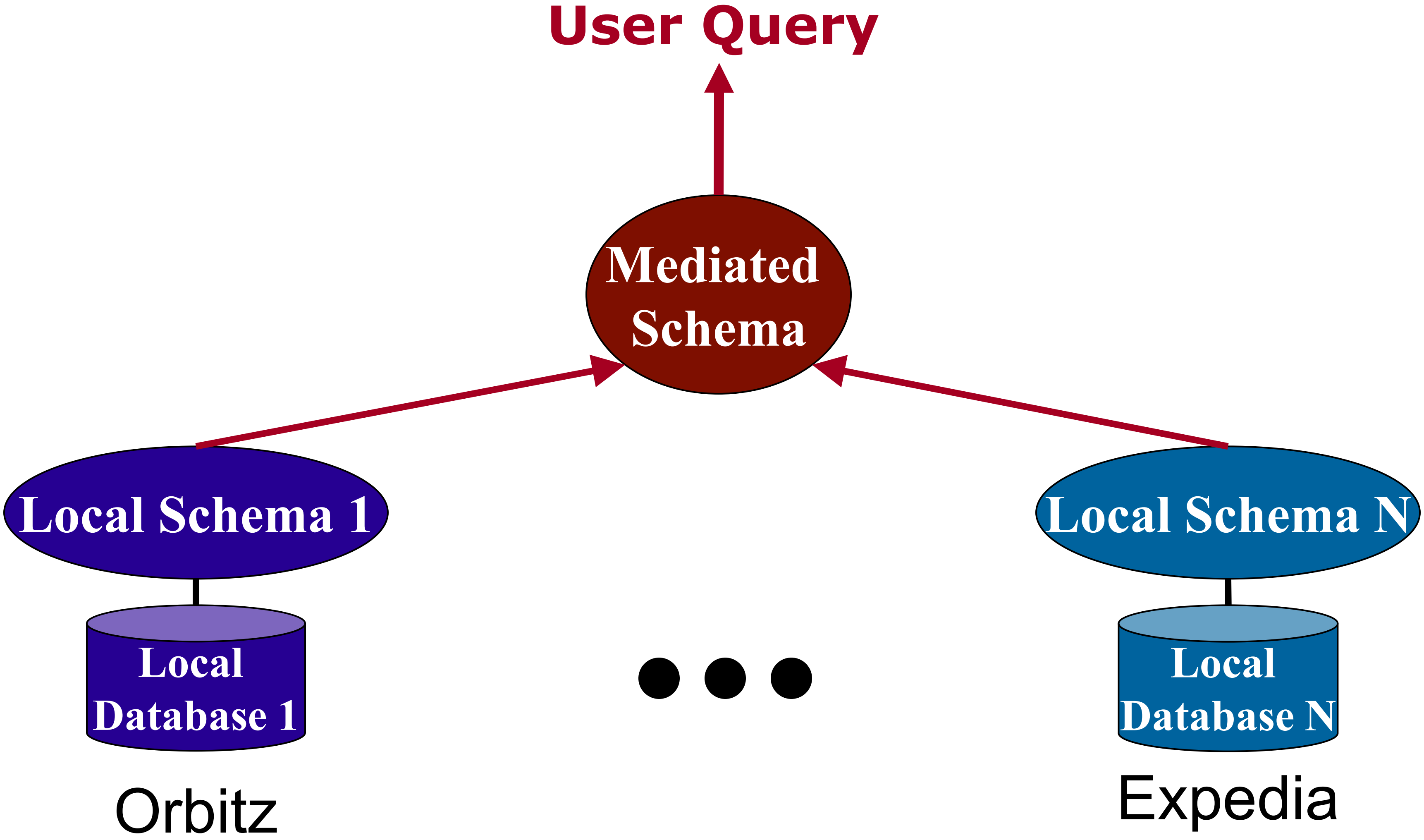
Discussion question

- Where do you think this mediated schema comes from? What kinds of information should be taken into account when building one?

How can we relate concepts in one schema to concepts in another?

- Views, glorious views! (I told you they were handy)
- In a *materialized view*, we compute what the answers are and save the result

Previous Data Integration Architecture: Global-As-View (GAV)

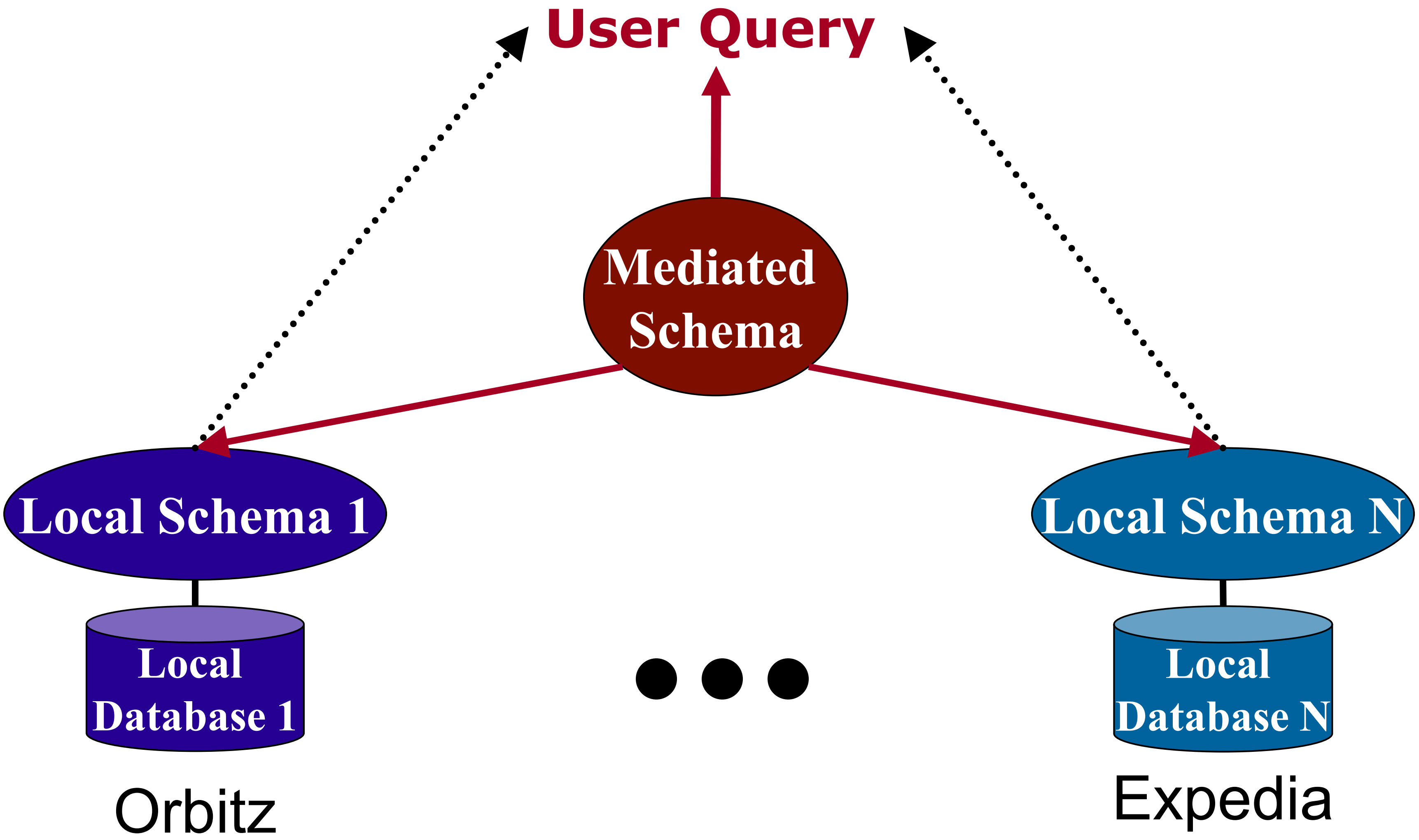


Global sources are views on source schemas

Example of Global-As-View (GAV)

- Mediated schema:
  Airport(code, city)
  Feature(city, attraction)
- Source schemas:
  Expedia-Air(aircode,postalcode)
  CanadaPost(postalcode, city)
- Mapping:
  Airport(code, city) :- Expedia-Air(code, postcode),
  CanadaPost(postalcode,city)
- How do you answer a query?
- What if you want to add OrbitzA(code,postcode)?

Information Manifold Data Integration Architecture: Local-As-View (LAV)



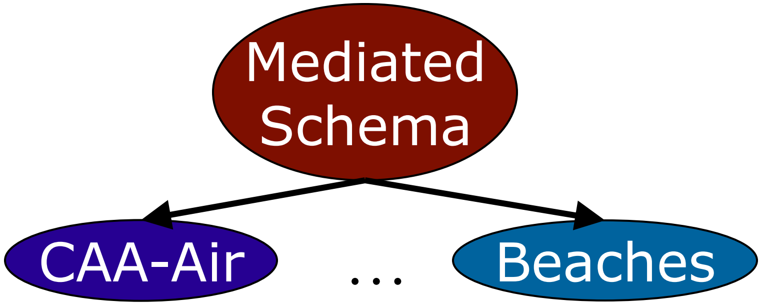
Local sources are views on mediated schema

Local As View (LAV)

A *view* is a named query

LAV: local source is *materialized view* over mediated schema

Mediated Schema:
  Airport(code, city)
  Feature(city, attraction)



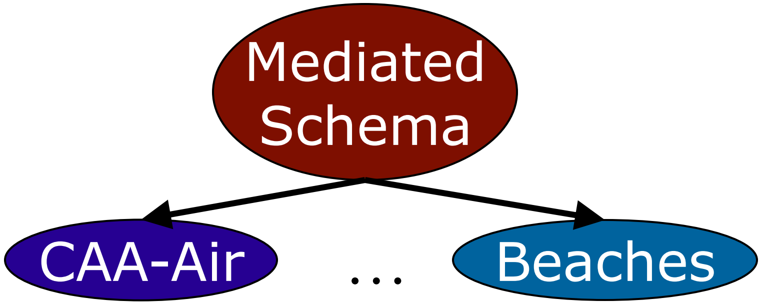
Local Sources/Views:
  CAA-Air(code, city) :- Airport(code, city)
  Beaches(code) :- Airport(code, city), Feature(city, “Beach”)

Local As View (LAV)

A *view* is a named query

LAV: local source is *materialized view* over mediated schema

Mediated Schema:
  Airport(code, city)
  Feature(city, attraction)



Local Sources/Views:
  CAA-Air(code, city) :- Airport(code, city)
  Beaches(code) :- Airport(code, city), Feature(city, "Beach")

- 👍 Adding new sources is easy
- 👎 Rewriting queries is NP-complete

Answering Queries Using Views

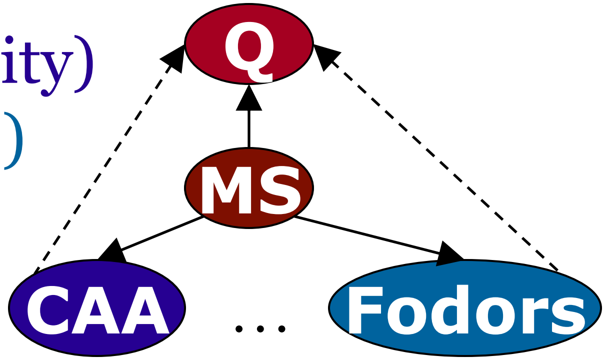
Query:

Dest(code) :- Airport(code, city), Feature(city, "Beach")

Sources/Views:

CAA-Air(code, city) :- Airport(code, city)

Fodors(city, POI) :- Feature(city, POI)



Rewriting:

Dest(code):-CAA-Air(code, city), Fodors(city, "Beach")

Maximally Contained Rewriting: all answers to Query are a subset of those of Rewriting, and Rewriting contains all possible answers given local sources

Answering Queries Using Views

Query:

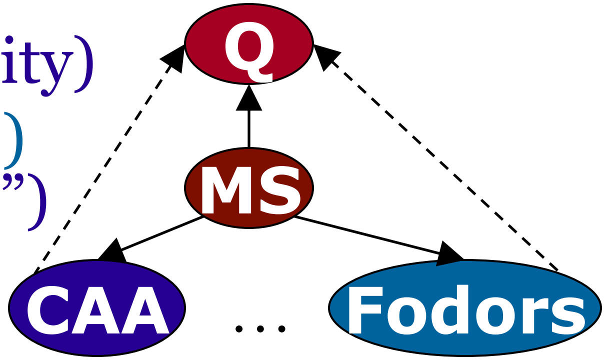
Dest(code) :- Airport(code, city), Feature(city, “Beach”)

Sources/Views:

CAA-Air(code, city) :- Airport(code, city)

Fodors(city, POI) :- Feature(city, POI)

Sun-Surf(city) :- Feature(city, “Beach”)



Rewriting:

Dest(code):-CAA-Air(code, city), Fodors(city, “Beach”) ∪
Dest(code):-CAA-Air(code, city), Sun-Surf(city)

Maximally Contained Rewriting: all answers to Query are a subset of those of Rewriting, and Rewriting contains all possible answers given local sources

Containment, what is it?

- For two queries, Q_1 and Q_2, if all answers to Q_1 are a subset of those for Q_2 for all databases, then Q_1 is *contained in* Q_2.
- Denoted as $Q_1 \subseteq Q_2$.
- For example, if
  $Q_1(x,x)$:-e1(x,x)
  $Q_2(y,z)$:-e1(y,z)
  $Q_1 \subseteq Q_2$.

Equivalent queries

- $Q_1 \equiv Q_2$ if they return the same answers for all databases. This is the same as $Q_1 \subseteq Q_2$ and $Q_2 \subseteq Q_1$
- For example, if

  $Q_1(X,Y)$:- $e_1(X,Z),e_2(Z,Y),e_1(X,W)$

  $Q_2(X,Y)$:-$e_1(X,Z)$, $e_2(Z,Y)$

  $Q_1 \equiv Q_2$.

How do you prove containment?

- There are a number of different ways, but don’t worry about it. The key thing is that even for conjunctive queries, it’s still NP-complete in the number of subgoals in the query.

So what’s a maximally contained rewriting then?

- It’s a rewriting where the rewritten query is *contained* in the original query, but it has as many answers as possible given the sources.
- Like the example above
- So how do you compute them?

Bucket Algorithm: Populating buckets

For each subgoal in the query, place relevant views in the subgoal's bucket

Inputs:

$Q(x)$:- $r_1(x,y)$ & $r_2(y,x)$

$V_1(a)$:-$r_1(a,b)$

$V_2(d)$:-$r_2(c,d)$

$V_3(f)$:- $r_1(f,g)$ & $r_2(g,f)$

Buckets:

$r_1(x,y)$	$r_2(y,x)$
$V_1(x)$,$V_3(x)$	$V_2(x)$, $V_3(x)$

Combining Buckets

For every combination in the Cartesian products from the buckets, check containment in the query

$Q(x) \text{:- } r_1(x,y)\ \&\ r_2(y,x)$

$V_1(a)\text{:-}r_1(a,b)$

$V_2(d)\text{:-}r_2(c,d)$

$V_3(f)\text{:- } r_1(f,g)\ \&\ r_2(g,f)$

Bucket Algorithm checks all possible combinations

Buckets:

$r_1(x,y)$	$r_2(y,x)$
$V_1(x),V_3(x)$	$V_2(x), V_3(x)$

Candidate rewritings:

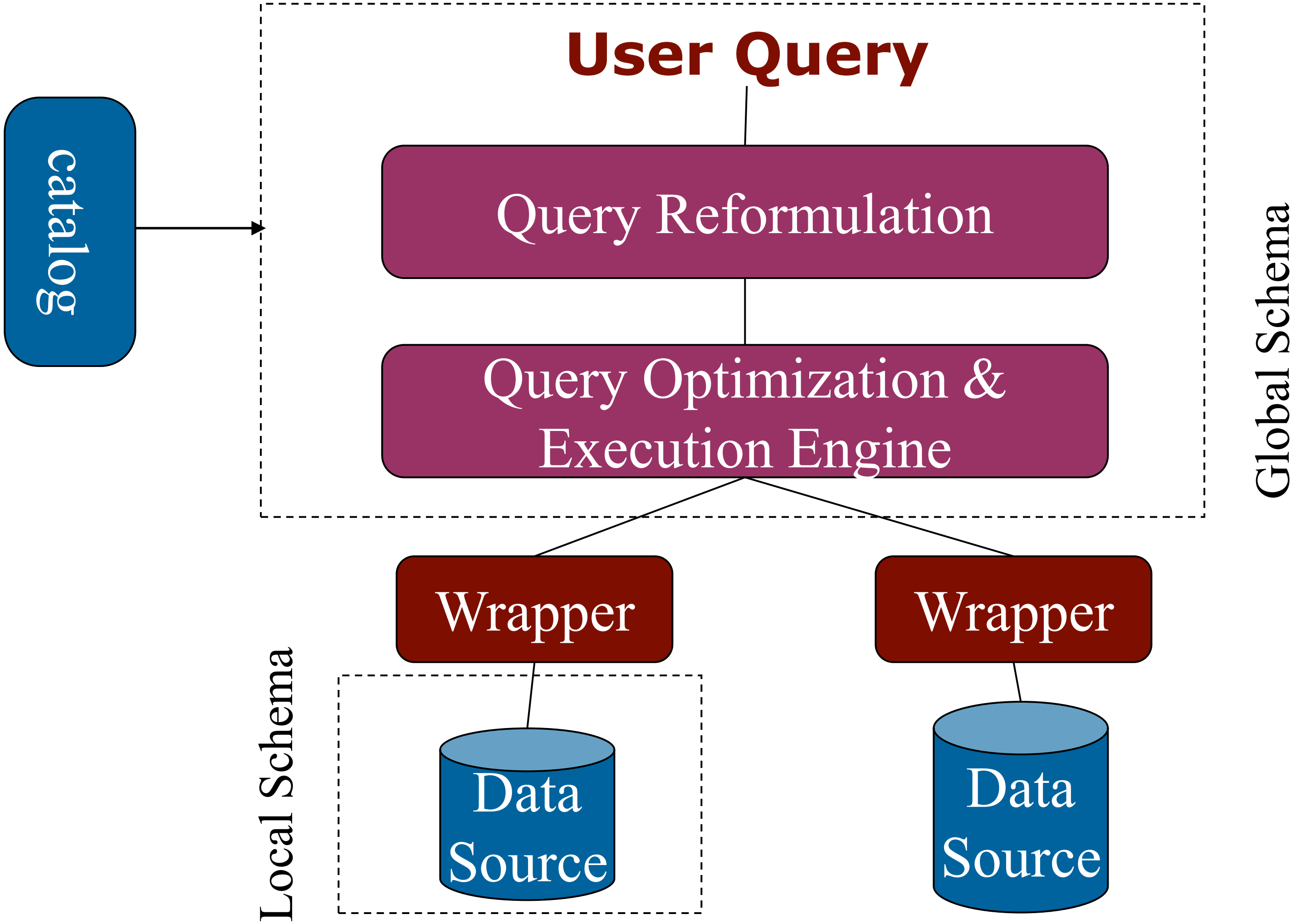
$Q'_1(x) \text{ :- } V_1(x)\ \&\ V_2(x)$ ✗

$Q'_2(x) \text{ :- } V_1(x)\ \&\ V_3(x)$ ✗

$Q'_3(x) \text{ :- } V_3(x)\ \&\ V_2(x)$ ✗

$Q'_4(x) \text{ :- } \underbrace{V_3(x)}_{r_1(x,y)}\ \&\ \underbrace{V_3(x)}_{r_2(y,x)}$ ✓

Sample Data Integration Architecture



Discussion

- This paper won the 10 year test of time award. Why do you think that the committee chose it?

So that’s the initial data integration paper. What happened then?

Schema mappings (coming up a bit in a few weeks)

- Where do those mappings come from? What do they look like?

Peer Data Management Systems (coming up Wednesday)

- Rather than have a centralized authority, make things distributed

Model Management

- Most metadata applications are redone from scratch every time.
- It would be nice to have an algebra (like relational algebra) only on the schema level so that these algorithms could be reused

Data Spaces (coming up next Monday)

- Pay as you go data integration

Discussion

- Which of these topics would you most want to work on and why?

Industry: Data Integration → Enterprise Information Integration

- Challenges:
  - Scale up and performance
  - Horizontal (general) vs. vertical (solving entire problem)
  - Integration with EAI and other middleware
- But did make it

Discussion

- The second paper was a result of a 10 year “test of time award”. As such it was not subject to rigorous peer review. What should we expect to be different about such papers from normal ones? What should we expect to be the same?

Any questions about what I expect?

- Things to keep in mind for the presenters:
  - It is not necessary to present the entire paper (I’ll give you a list of things not to skip)
  - You do not need to understand every last detail of the paper
- Things to keep in mind for the discussion leaders
  - Make sure you don’t leave all the discussion until the end
  - If you have trouble calling on people, I will help.