Information Integration

Mediators

Warehousing

Answering Queries Using Views

Example Applications

1. Enterprise Information Integration: making separate DB's, all owned by one company, work together.
2. Scientific DB's, e.g., genome DB's.
3. Catalog integration: combining product information from all your suppliers.

Challenges

1. *Legacy databases* : DB's get used for many applications.
   - You can't change its structure for the sake of one application, because it will cause others to break.
2. *Incompatibilities* : Two, supposedly similar databases, will mismatch in many ways.

Examples: Incompatibilities

- *Lexical* : `addr` in one DB is `address` in another.
- *Value mismatches* : is a "red" car the same color in each DB? Is 20 degrees Fahrenheit or Centigrade?
- *Semantic* : are "employees" in each database the same? What about consultants? Retirees? Contractors?

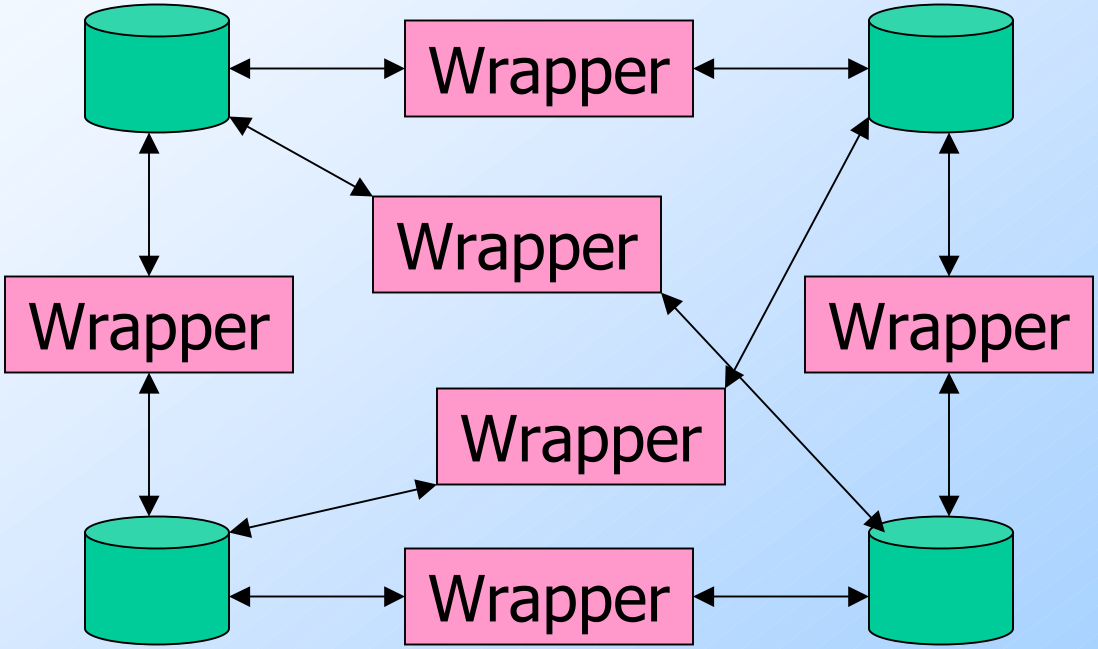
What Do You Do About It?

- Grubby, handwritten translation at each interface.
  - Some research on automatic inference of relationships.
- *Wrapper* (aka "adapter") translates incoming queries and outgoing answers.

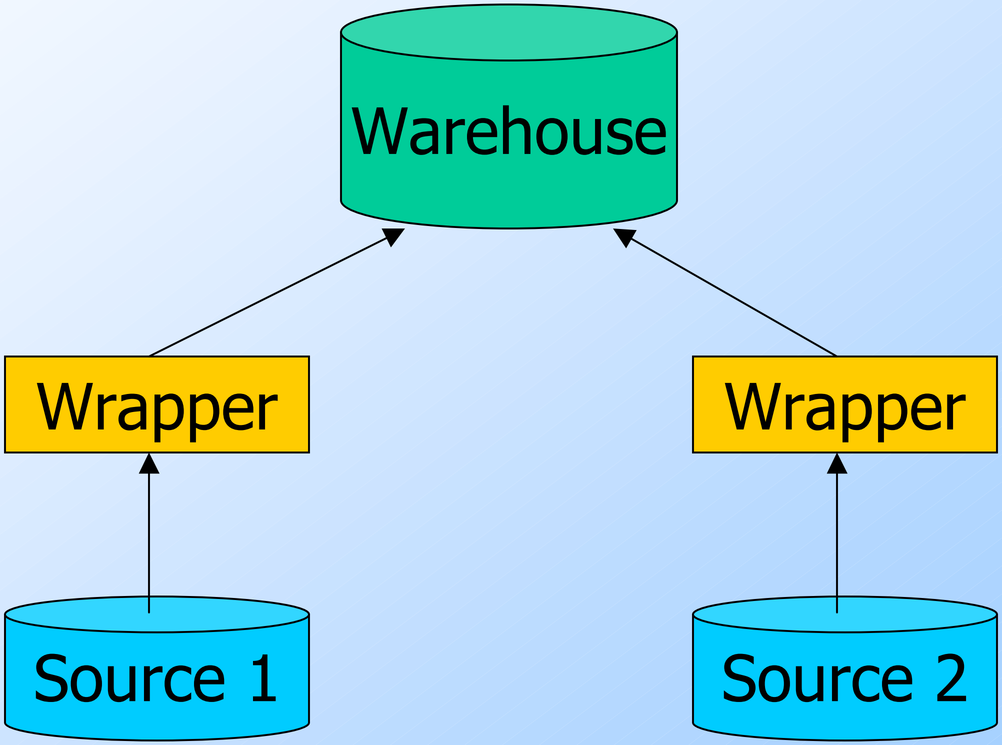
Integration Architectures

1. *Federation* : everybody talks directly to everyone else.
2. *Warehouse* : Sources are translated from their local schema to a global schema and copied to a central DB.
3. *Mediator* : Virtual warehouse --- turns a user query into a sequence of source queries.

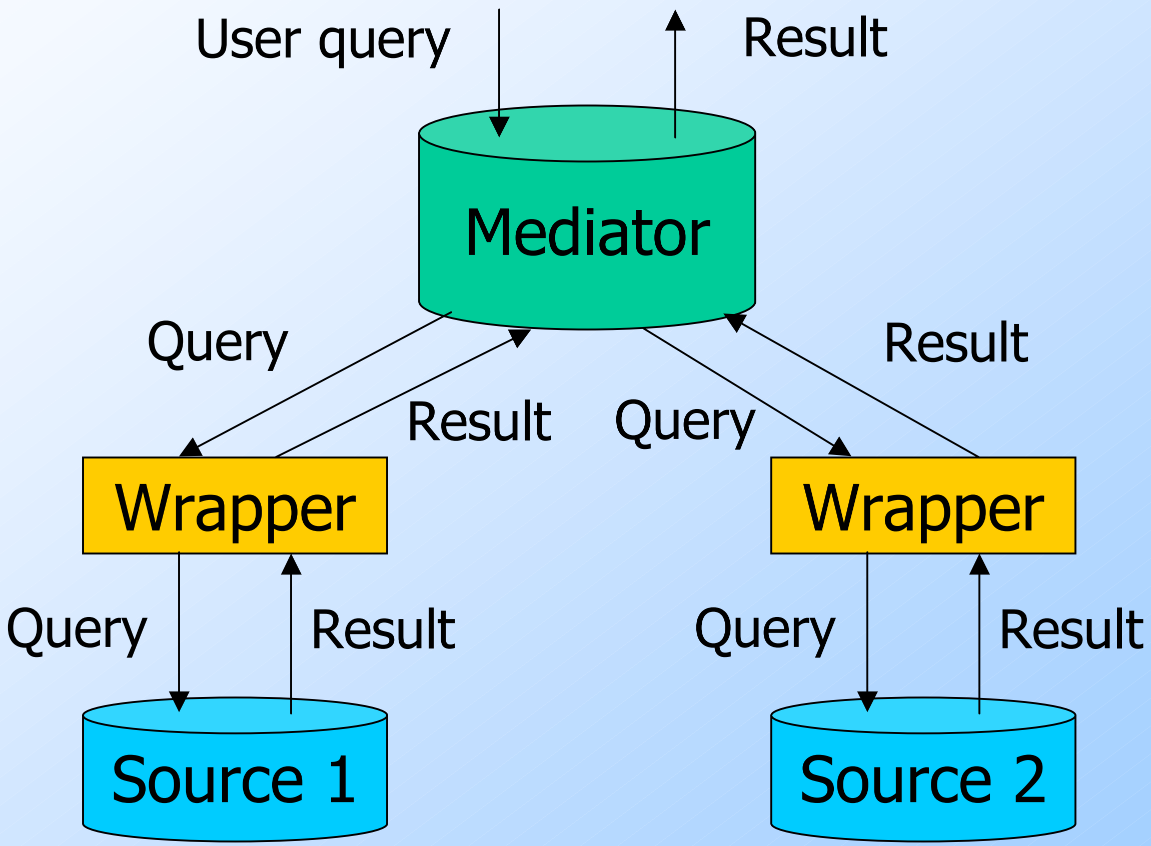
Federations



Warehouse Diagram



A Mediator



User query
Result
Mediator
Query
Result
Query
Result
Wrapper
Wrapper
Query
Result
Query
Result
Source 1
Source 2

Two Mediation Approaches

1. *Global as View* : Mediator processes queries into steps executed at sources.
2. *Local as View* : Sources are defined in terms of global relations; mediator finds all ways to build query from views.

Example: Catalog Integration

- Suppose Dell wants to buy a bus and a disk that share the same protocol.
- Global schema:

```
Buses(manf,model,protocol)
Disks(manf,model,protocol)
```

- Local schemas: each bus or disk manufacturer has a (model,protocol) relation --- manf is implied.

Example: Global-as-View

- Mediator might start by querying each bus manufacturer for model-protocol pairs.
  - The wrapper would turn them into triples by adding the manf component.
- Then, for each protocol returned, mediator queries disk manufacturers for disks with that protocol.
  - Again, wrapper adds manf component.

Example: Local-as-View

- Sources' capabilities are defined in terms of the global predicates.
  - E.g.,Quantum's disk database could be defined by QuantumView(M,P) = Disks('Quantum',M,P).
- Mediator discovers all combinations of a bus and disk "view," equijoined on the protocol components.

A Harder LAV Case

- The mediator supports a par(c,p) relation (which doesn't really exist, but can be queried).
- Sources can support views that are complex expressions of par.
- A logic is needed to work with queries and view definitions.
  - Datalog is a good choice.

Example: Some Local Views

- Source 1 provides some parent facts.

  V1(c,p) <- par(c,p)

- Source 2, run by the “Society of Grandparents,” supports only grandparent facts.

  V2(c,g) <- par(c,p) AND par(p,g)

Example – (2)

- Query (great-grandparents):

ggp(c,x) <- par(c,u) AND par(u,v) AND par(v,x)

- How can the sources provide solutions that provide all available answers?

Example – (3)

Sol1(c,x) <- V1(c,u) AND V1(u,v) AND V1(u,x)

Sol2(c,x) <- V1(c,u) AND V2(u,x)

Sol3(c,x) <- V2(c,v) AND V1(v,x)

- No other queries involving the views can provide more ggp facts.
- Deep theory needed to explain.

Comparison: LAV Vs. GAV

- GAV is simpler to implement.
  - Lets you control what the mediator does.
- LAV is more extensible.
  - Add a new source simply by defining what it contributes as a view of the global schema.
  - Can get some use from grandparent info., even if par(c,p) is the only mediator data.

Course Plug

- In the Spring 07-08, Alon Halevy (Google) is teaching CS345C *Information Integration*.
- It will cover this technology and many others.