Data integration and transformation

Paolo Atzeni

Dipartimento di Informatica e Automazione
Università Roma Tre
7/10/2009

Integrazione e trasformazione di dati

- Obiettivi formativi
  - Acquisire familiarità con i complessi problemi derivanti dall'utilizzo di dati in formati diversi provenienti da fonti diverse.
    Studio delle proposte scientifiche recentemente formulate, con riferimento ad aree quali le basi di dati federate, l'integrazione di basi di dati, il data exchange, il model management e i dataspace
- Approccio:
  - Corso seminariale, basato su letteratura scientifica, in parte illustrata dal docente in parte approfondita individualmente

A ten-year goal for database research

- The “Asilomar report”
  (Bernstein et al. Sigmod Record 1999 www.acm.org/sigmod):
  - ***The information utility:***
    make it easy for everyone to store, organize, access, and analyze the majority of human information online

- A lot of interesting work has been done but …
- …integration, translation, exchange are still difficult…
- **… 2009 has come… we are late!**

A general framework: cooperation

- "The capacity of a system to interact (effectively) with other systems, possibly managed by different organizations"

Forms of cooperation

- **Process-centered cooperation**:
  – the systems offer **services** one another, by exchanging messages, information or documents, or by triggering activities, without making remote data explicitly visible
- **Data-centered cooperation**:
  – the systems offer **data** one another; data is distributed, heterogeneous and autonomous, and accessible from remote locations according to some co-operation agreement

Databases in the Internet era

- Databases before the Internet
  - An internal infrastructure, a precious resource, but usually hidden, with some controlled cooperation
- Internet changes the requirements
  - More users (not only humans), more diverse cooperating systems (distributed, heterogeneous, autonomous), more types of data
- "Future" Internet changes more
  - New devices (embedded everywhere), even more users (many “per person”), real mobility, need for personalization and adaptation

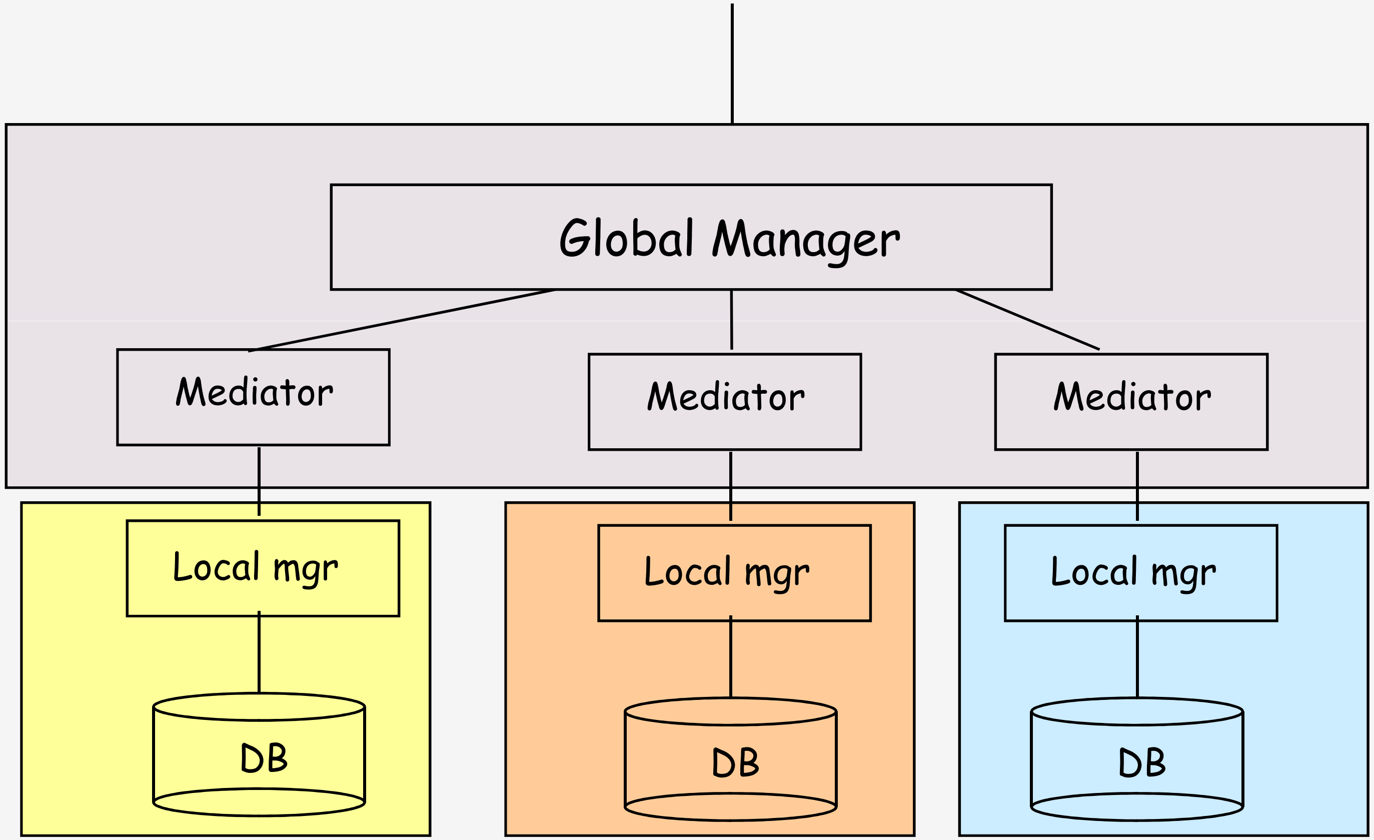
The most studied form of data-centered cooperation: integration

- We are interested in data-centered cooperation, often referred to as integration

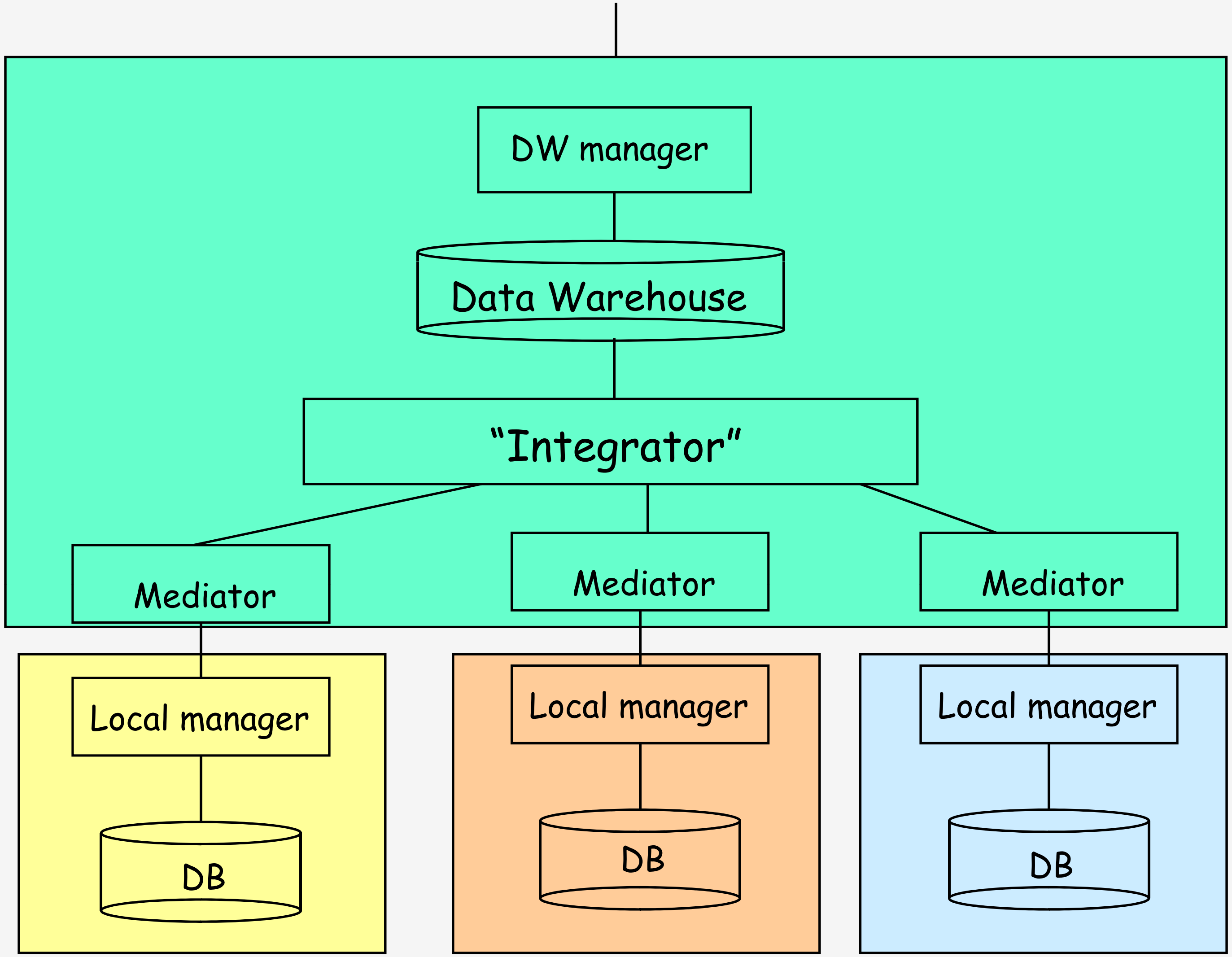
  "The unification of related, heterogeneous data from disparate sources, for example, to enable collaboration" (Hammer & Stonebraker 2005)

- Some "paradigms" …

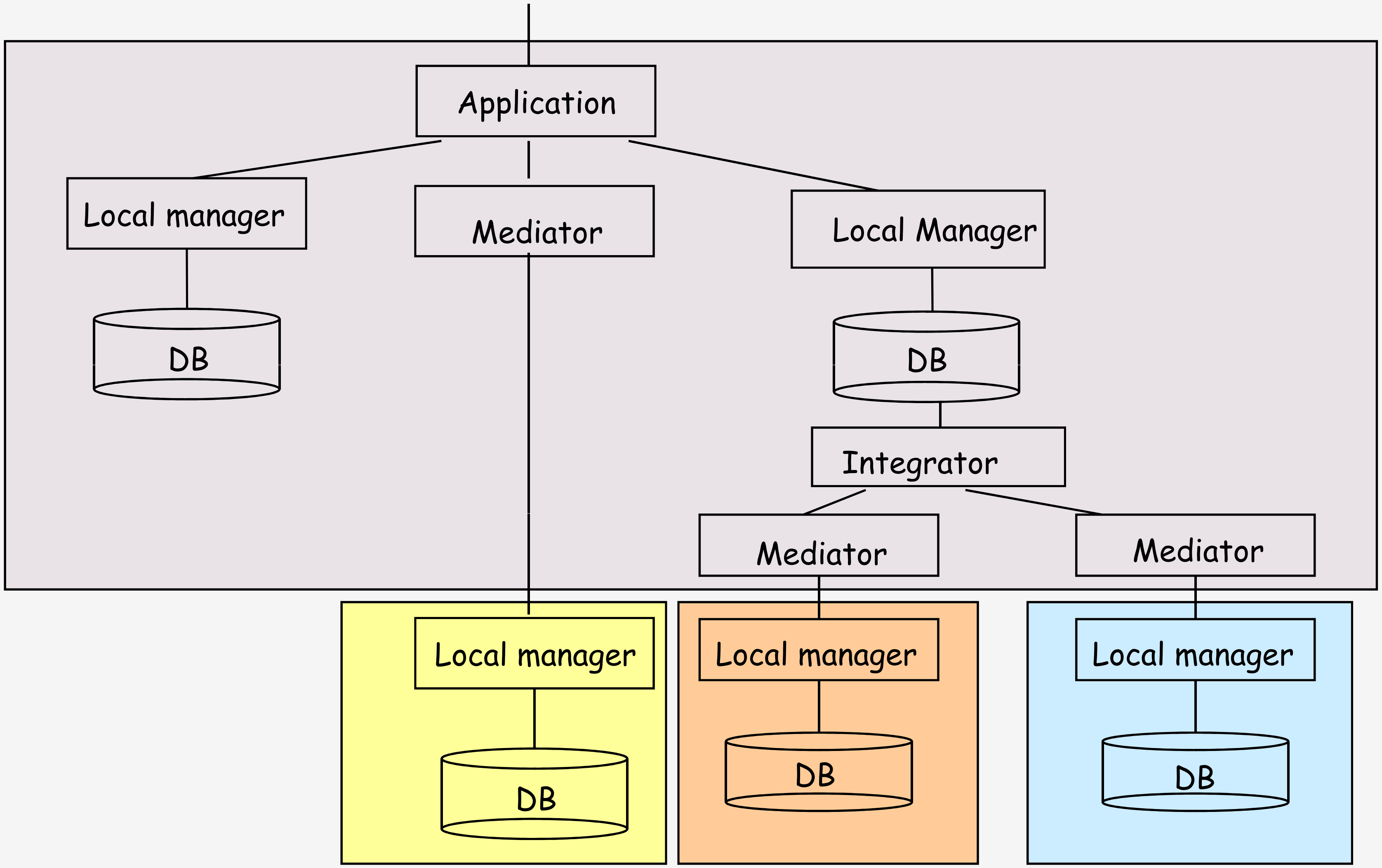
Multidatabase



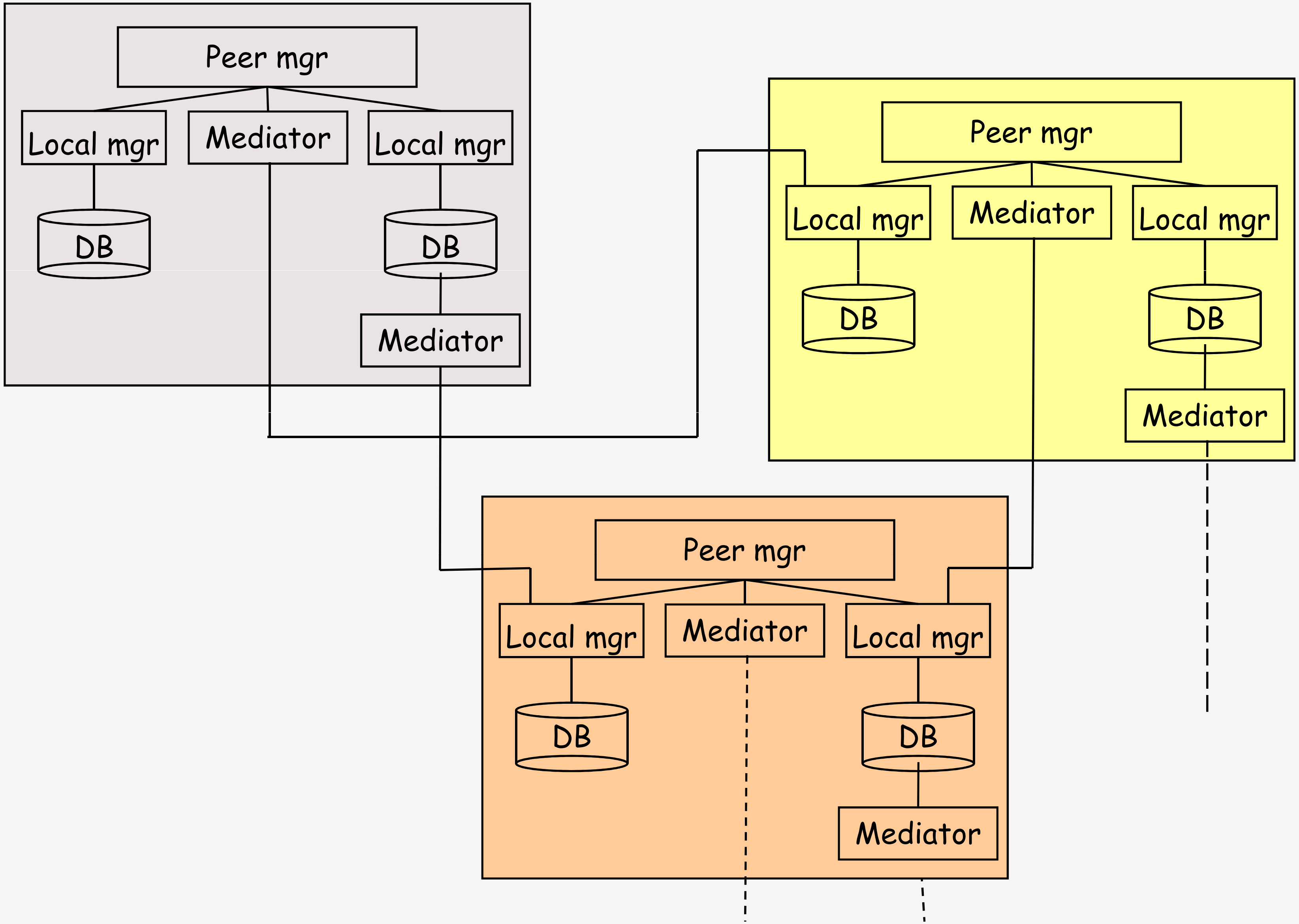
Data Warehousing System



Intermediate solutions in practice



Peer-based architecture

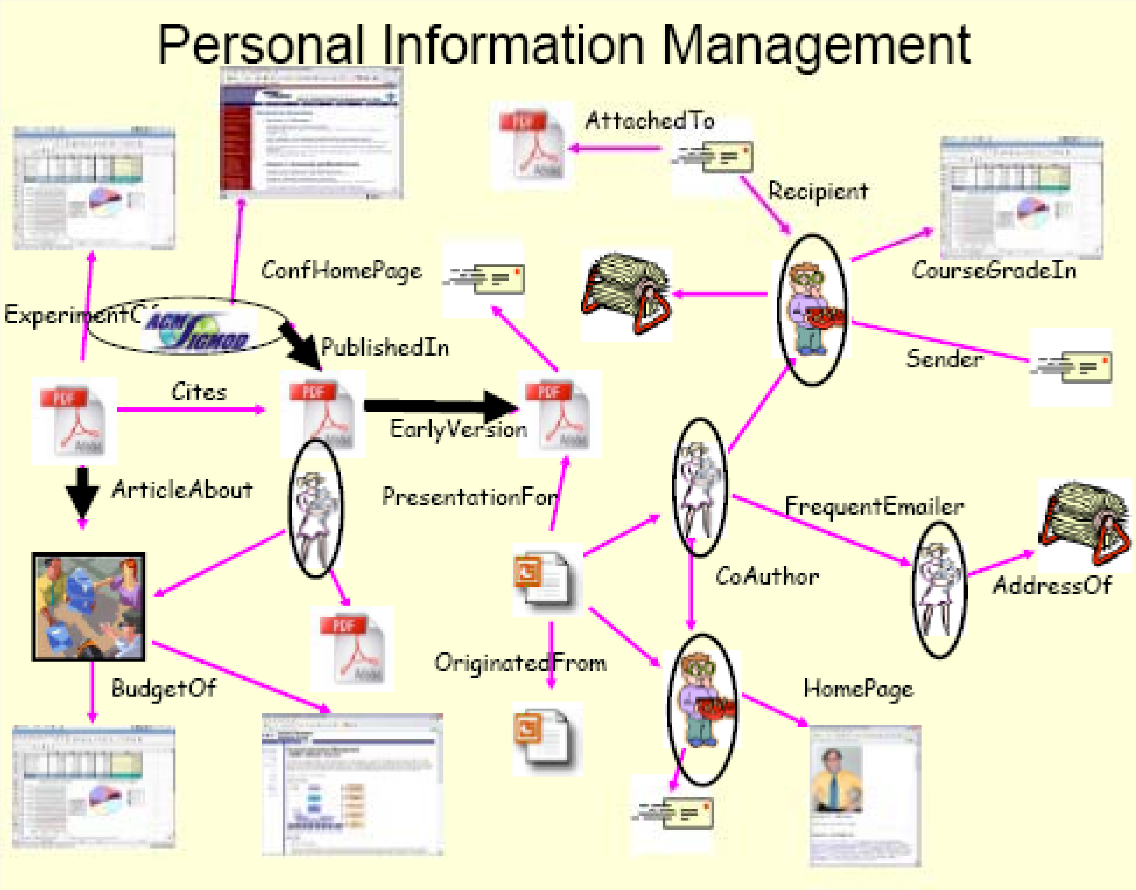


Data is not just in databases

- Mail messages
- Web pages
- Spreadsheets
- Textual documents
- Palmtop devices, mobile phones
- Multimedia annotations (e.g., in digital photos)
- XML documents

Data spaces

- The information and data is often unstructured and not preprocessed



The same data in the same form?

- Adaptivity:
  - Personalization: content adapted to the user
    - upon system's decision
    - upon user's request
  - Customization: structure adapted to the user
    - according to the user's role
    - upon user's request
  - Context dependence
    - User, Device, Network, Place, Time, Rate

A general need

- We have data at various places, and data has to be
  - exchanged
  - replicated
  - migrated
  - integrated
  - adapted

A major difficulty

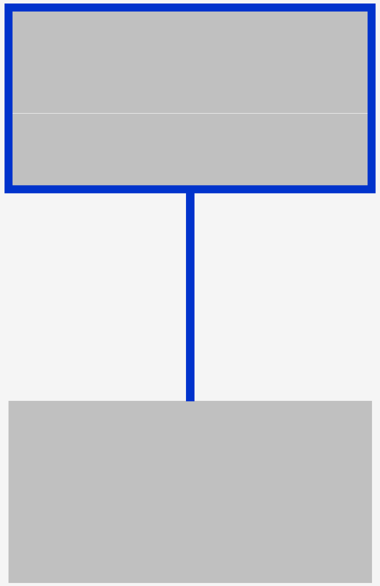
- Heterogeneity
  - System level
  - Model level
  - Structural (different structure for similar data)
  - Semantic (different meaning for the same structure)
- Many efforts, but current techniques are mostly manual and *ad hoc*

Three problems

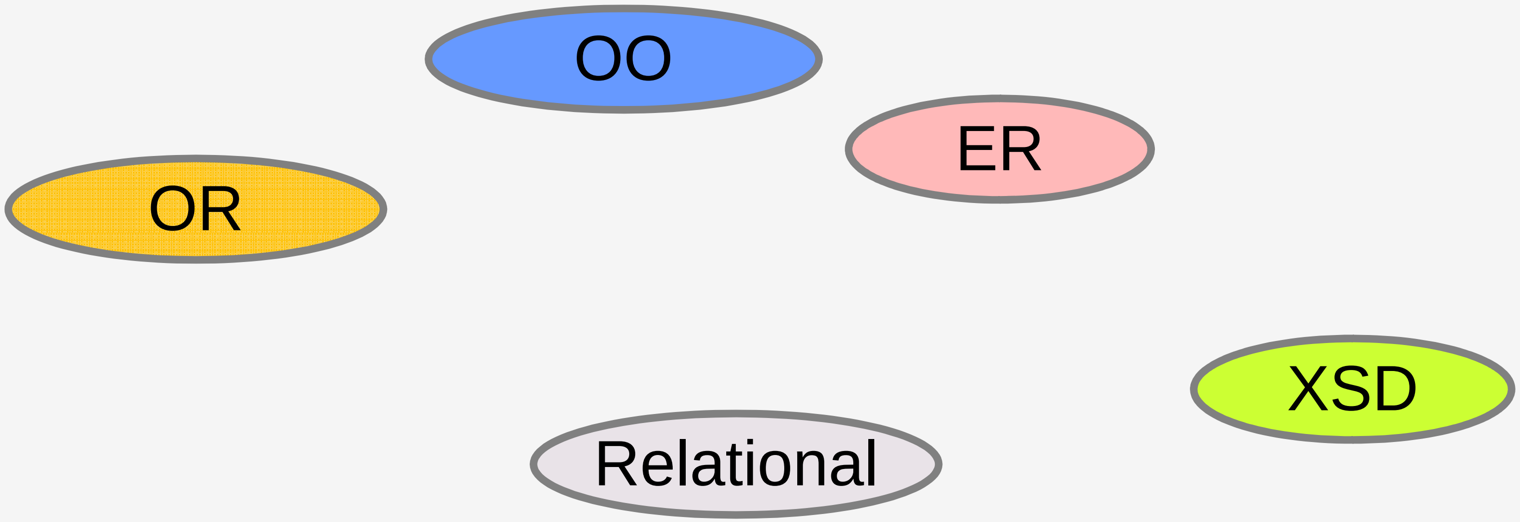
- Schema and data translation
- Schema and data integration
- Data exchange

Schema and data translation

- Given a schema find another one with respect to some specific goal (better quality, another model, …)



Many different models



…

Many different models (and variants …)

OR w/ PK, gen, ref, FK

OR w/ PK, gen, ref

OR w/ PK, gen, FK

OR w/ PK, ref, FK

OR w/ gen, ref

OR w/ PK, ref

OR w/ PK, FK

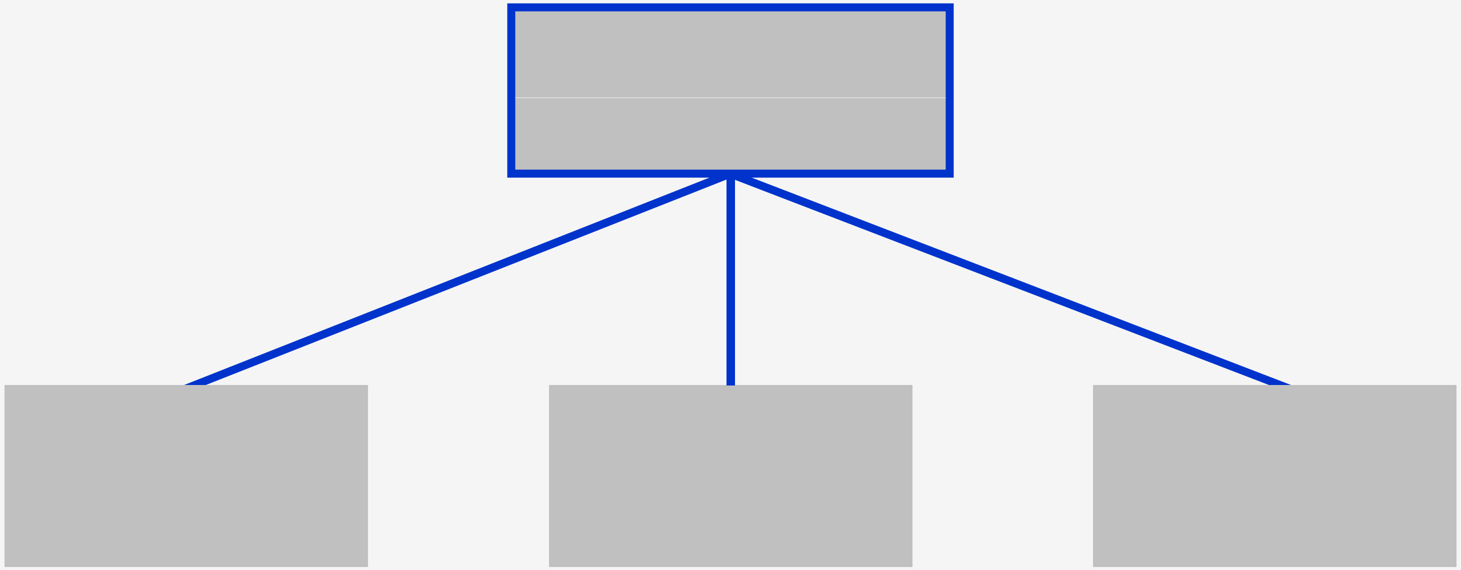
OR w/ ref

…

Relational

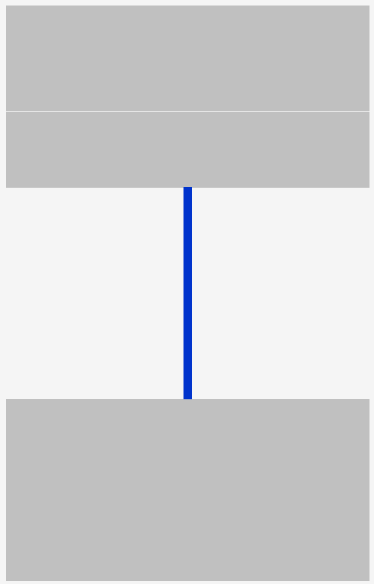
Schema and data integration

- Given two or more sources, build an integrated schema or database

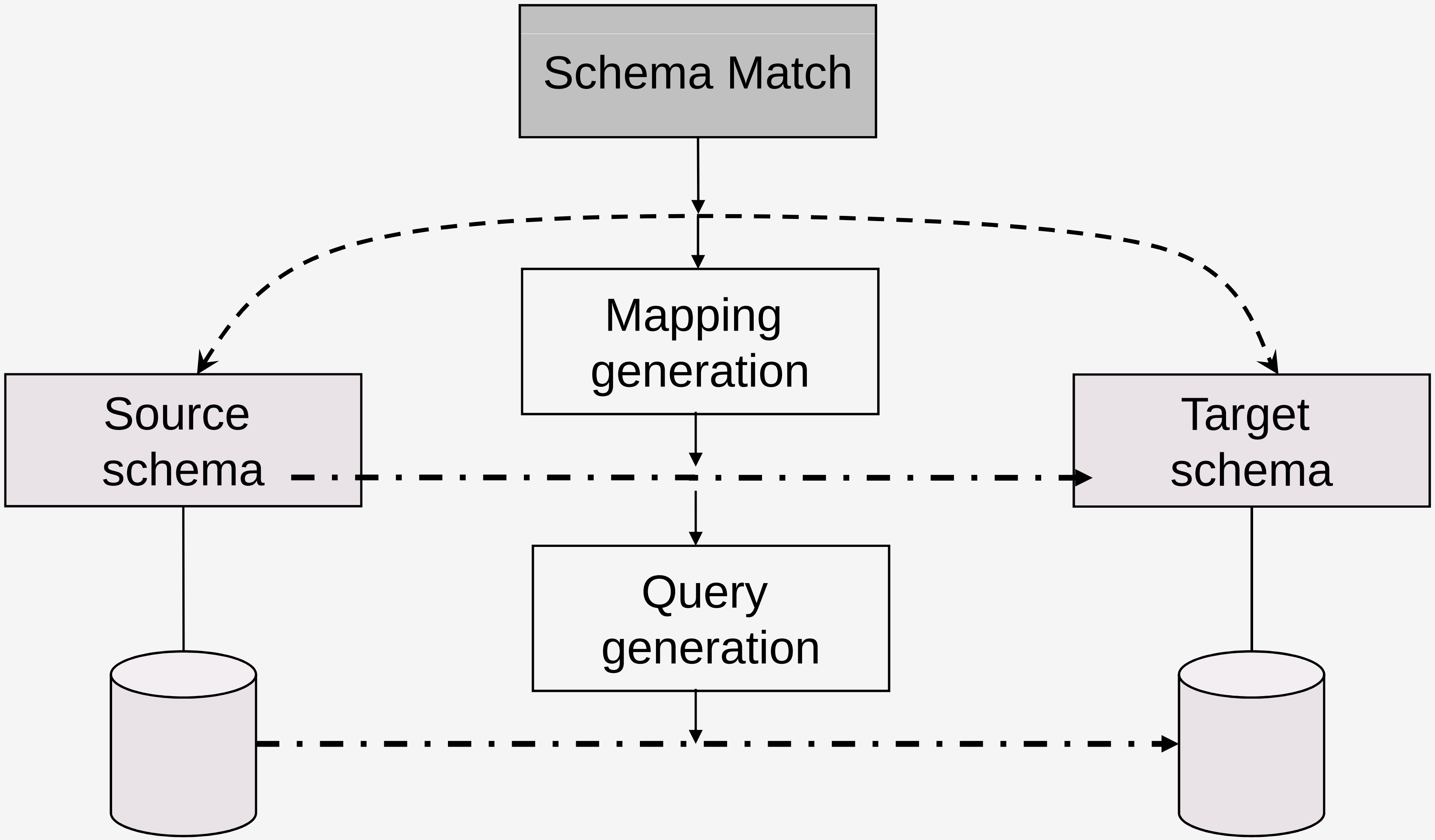


Data exchange

- Given a source and a target schema, find a transformation from the former to the latter



Data exchange, a typical approach (the Clio project)



Data exchange, example

Address (Id Addr)

Professor (Id Name Sal)

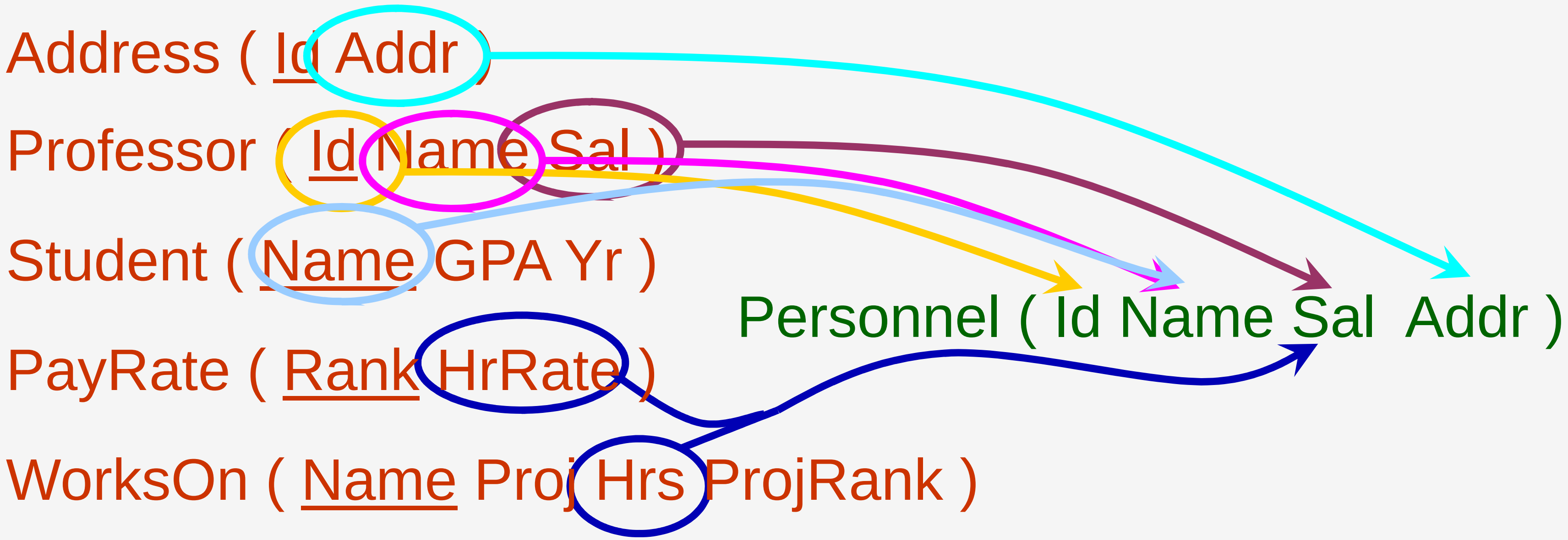
Student (Name GPA Yr)

PayRate (Rank HrRate)

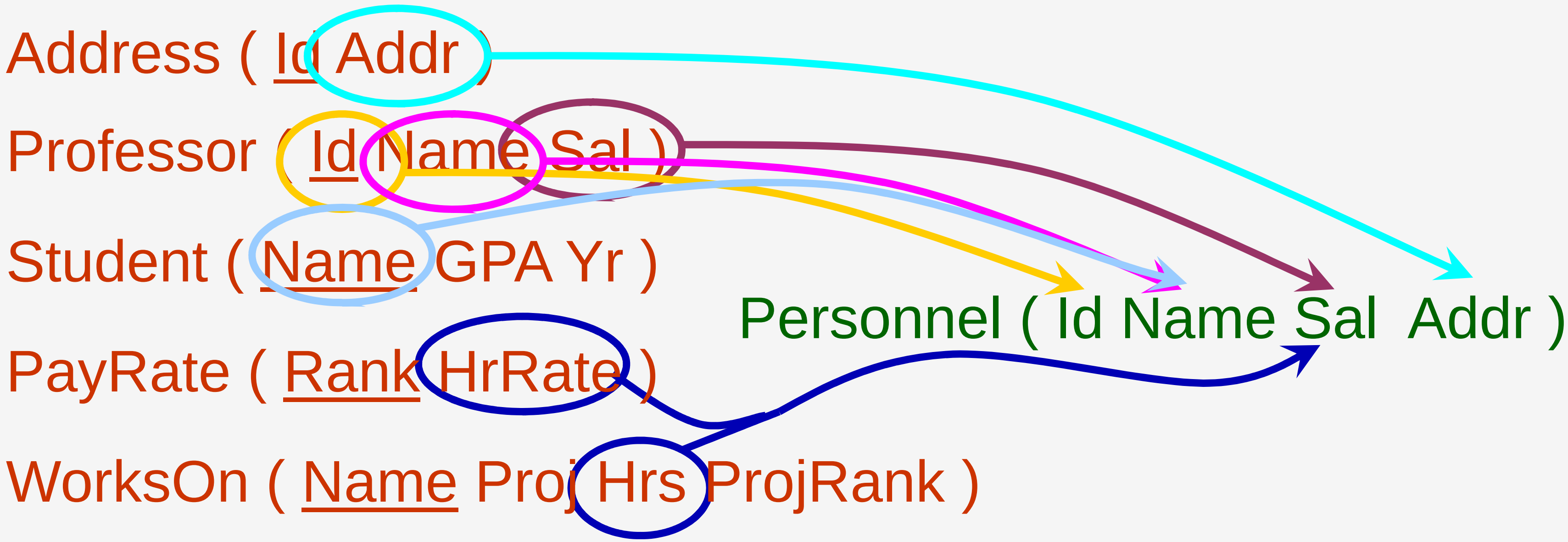
WorksOn (Name Proj Hrs ProjRank)

Personnel (Id Name Sal Addr)

Data exchange, example



The process, example




```
SELECT P.Id, P.Name, P.Sal, A.Addr
FROM Professor P, Address A
WHERE A.Id = P.Id
UNION ALL
SELECT NULL AS Id, S.Name, p.HrRate * W.Hrs, NULL AS Addr
FROM PayRate P, Student S, WorksOn W
WHERE W.Name = S.Name AND S.Yr = P.Rank
```

A direction for the solutions

- Be **general**!
  - *ad hoc* solution could work in-the-small, but they
    - are repetitive and time consuming
    - do not scale
    - are not maintainable

- Historical notes:
  - W. C. McGee: Generalization: Key to Successful Electronic Data Processing. J. ACM 1959
- Indeed, databases are the result of generalization applied to secondary storage management!

Generality requires …

- … high-level descriptions of problems within the family of interest:
  - **Metadata**:
    - “data about data”
    - (formal or informal) description of structures and meaning
- General solutions leverage on metadata (and then operate on data as a consequence)

A wider perspective

- **(Generic) Model Management**:
  - A proposal by Bernstein et al (2000 +)
  - Includes a set of operators on
    - schemas and
    - mappings between schemas

Terminology: a warning

Model Mgmt people	Traditional DB people
Meta-metamodel	Metamodel
Metamodel	Model
Model	Schema

Schemas and mappings

- More on the issue later
- For the time being:
  - Schema:
    - a set of elements, related in some way to one another
  - Mapping:
    - a set of correspondences (pair of elements) or
    - its reification, a third schema related to the other two via two sets of correspondences

Model mgmt operators, a first set

- map = **Match** (S1, S2)
- S3 = **Merge** (S1, S2, map)
- S2 = **Diff** (S1, map)
- and more
  - map3 = Compose (map1, map2)
  - S2 = Select (S1, pred)
  - Apply (S, f)
  - list = Enumerate (S)
  - S2 = Copy (S1)
  - …

Match

- map = **Match** (S1, S2)
  - given
    - two schemas S1, S2
  - returns
    - a mapping between them
- the “classical” initial step in data integration:
  - find the common elements of two schemas and the correspondences between them

Merge

- S3 = **Merge** (S1, S2, map)
  - given
    - two schemas and a mapping between them
  - returns
    - a third schema (and two mappings)
- the “classical” second step in data integration:
  - given the correspondences, find a way to obtain one schema out of two

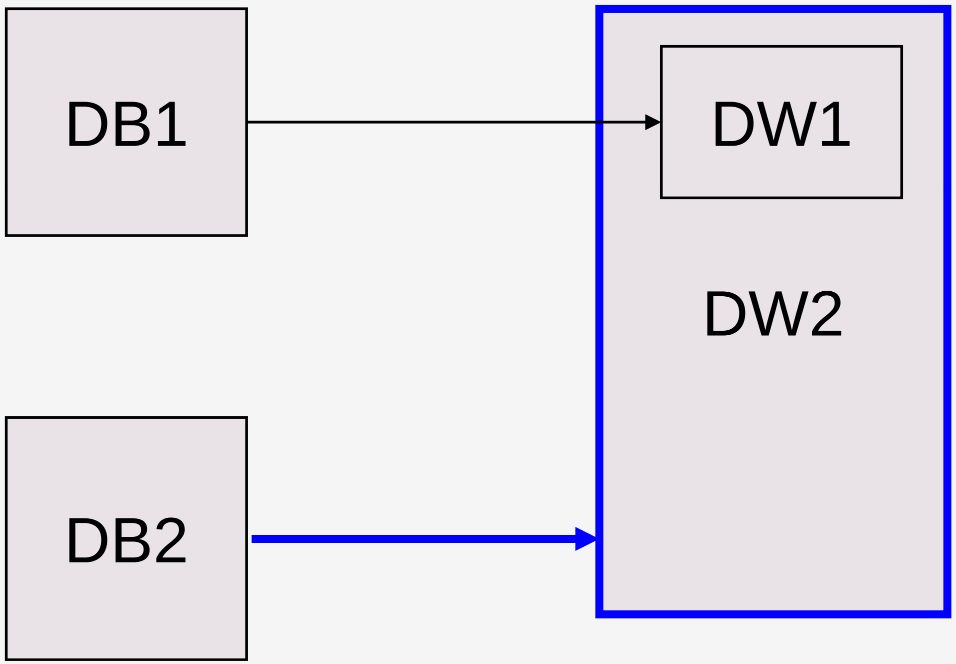
Diff

- S2 = **Diff** (S1, map)
  - given
    - a schema and a mapping from it (to some other schema, not relevant)
  - returns
    - a (sub-)schema, with the elements that do not participate in the mapping

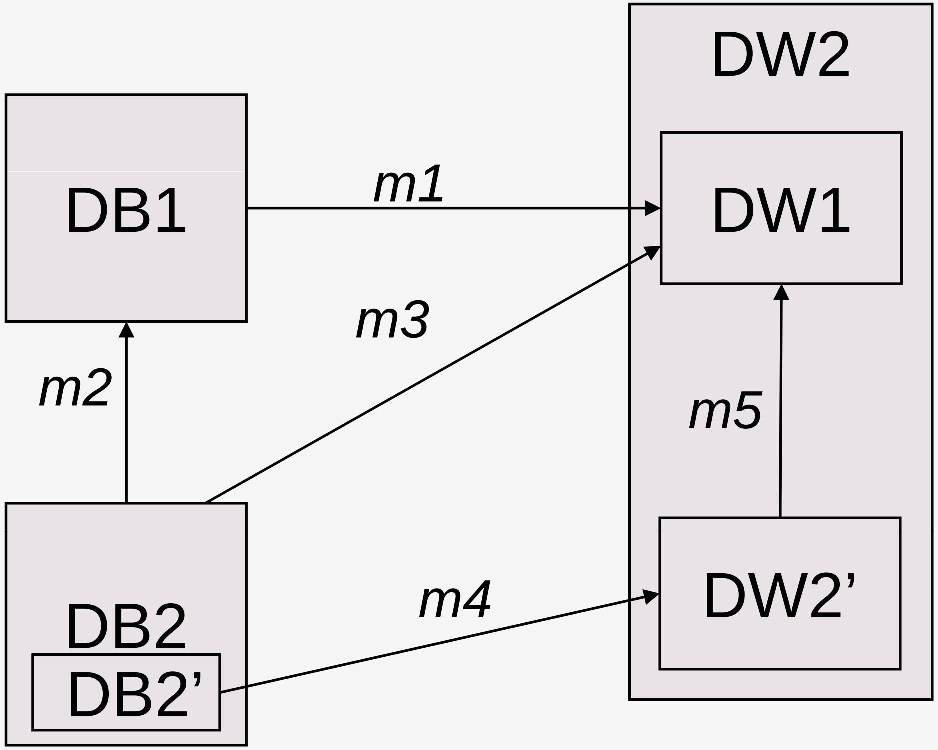
Example

(Bernstein and Rahm, ER 2000)

- A database (a “source”), a data warehouse and a mapping between the two
- We want to add a source, with some similarity to the first one
- and update the DW



Example, the "solution"



m2 = Match(DB1,DB2)

m3= Compose(m2,m1)

DB2'=Diff(DB2,m3)

DW2', m4 user defined

m5 = Match(DW1,DW2')

DW2 = Merge(DW,DW2',m5)

Magic does not exist

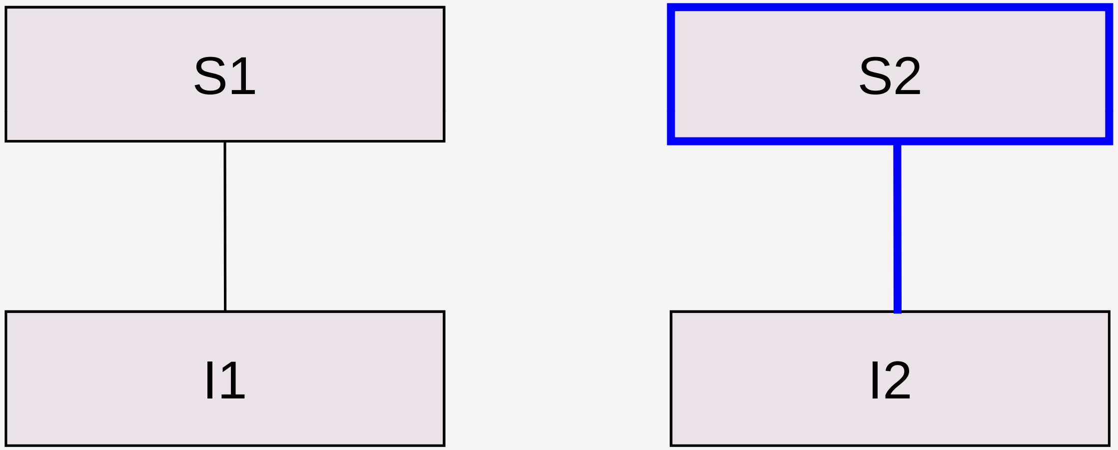
- Operators might require human intervention:
  - Match is the main case
- Scripts involving operators might require human intervention as well (or at least benefit from it):
  - a full implementation of each operator might not always available
  - a mapping might require manual specification
  - incomparable alternatives might exist

The “data level”

- The major operators have also an extended version that operates on data, and not only on schemas
- Especially apparent for
  - Merge

We also have heterogeneity

- Round trip engineering (Bernstein, CIDR 2003)
  - A specification, an implementation
  - then a change to the implementation: want to revise the specification
- We need a translation from the implementation model to the specification one



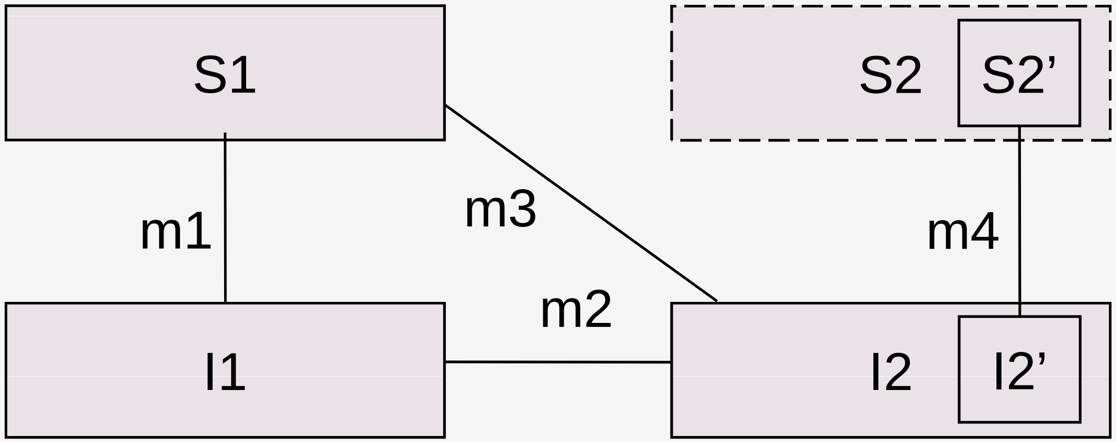
Model management with heterogeneity

- The previous operators have to be “model generic” (capable of working on different models)
- We need a “translation” operator
  - <S2, map12> = **ModelGen** (S1)

ModelGen, an additional operator

- <S2, map12> = **ModelGen** (S1)
  - given
    - a schema (in a model)
  - returns
    - a schema (in a different data model) and a mapping between the two
- A “translation” from a model to another
- I should call it “SchemaGen” …
- We should better write
  - <S2, map12> = **ModelGen** (S1,mod2)

Round trip engineering



m2 = Match (I1,I2)
m3 = Compose (m1,m2)
I2’= Diff(I2,m3)
<S2’,m4 > = Modelgen(I2’)
… Match, Merge

Summary

- data integration
- schema and data translation
- data exchange
- model management
- dataspaces