



Global as View Mediator Systems

Craig Knoblock
University of Southern California



Global as View

- Global integration model
- Each term in the global model is described as a view on the sources
- I.e., `person(name, age, salary) :-
 personnel_DB(name,age)
 payroll_DB(name, salary)`



Advantages

- Simple to process queries on the global model
 - Mapping to queries is specified in the axioms
- Terms in the global model can be composed of arbitrarily complex queries on the sources
- Easier to express and exploit conflict resolution strategies



Disadvantages

- Any change to a source could result in many changes to the definitions of the global terms
- Addition and removal of sources could require significant reengineering efforts
- Mapping to sources is fixed making it difficult to respond to unavailable sources or missing data

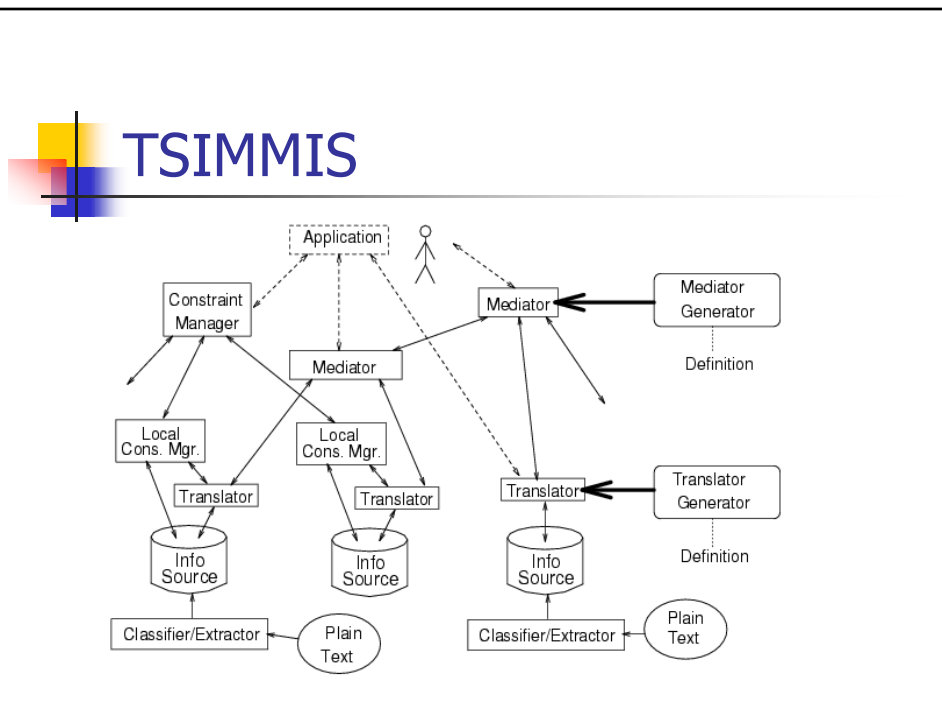
Example Systems

■ TSIMMIS

- [Chawathe, Garcia-Milina, Hammer, Ireland, Papakonstantinou, Ullman, Widom, 1994]
- Focused on using a self-describing object model to provide the unifying framework

■ HERMES

- [Subramanian et al., 1995]
- Focused on handling very heterogeneous types of data sources
 - E.g., relation, text documents, video, etc.





Translators

- Translator (or wrapper) logically converts the underlying data objects to a common model
 - Converts queries to the global model into requests the sources can execute
 - Converts data returned by the sources into the global model



Object Exchange Model (OEM)

- Self-describing object model
- Objects have labels that describe their meaning (similar to XML)
- Object representing Fahrenheit temperature of 80 degrees:
<temp-in-Fahrenheit, int, 80>
- Set of temperatures:
<set-of-temps, set, {cmp1,cmp2}>
 cmp1: <temp-in-Fahrenheit, int, 80>
 cmp2: <temp-in-Celsius, int, 20>



Mediators

- Mediator embeds the knowledge for processing a specific type of information
- Combine and process data from multiple translators
- Support requests in the OEM-QL language
- Can do additional processing on the data, such as normalizing dates, or just pass the entire objects



Example bibliographic source: Biblio

```
(bib, set, {doc1, doc2, ..., docn})
  doc1: (doc, set, {au1, top1, cn1})
    au1: (authors, set, {au11})
      au11: (author-ln, str, "Ullman")
    top1: (topic, str, "Databases")
    cn1: (local-call#, integer, 25)
  doc2: (doc, set, {au2, top2, cn2})
    au2: (authors, set, {au21, au22, au23})
      au21: (author-ln, str, "Aho")
      au22: (author-ln, str, "Hopcroft")
      au23: (author-ln, str, "Ullman")
    top2: (topic, str, "Algorithms")
    cn2: (dewey-decimal, str, "BR273")
  :
  docn: (doc, set, {aun, topn, cnn})
    aun: (one-author, str, "Michael Crichton")
    topn: (topic, str, "Dinosaurs")
    cnn: (fiction-call#, int, 95)
```



Example Query

- Query:

```
Select bib.doc.topic
```

```
From Biblio
```

```
Where bib.doc.authors.author-ln = "Ullman"
```

- Result:

```
<answer, set, {o1,o2}>
```

```
o1: <topic, str, "Databases">
```

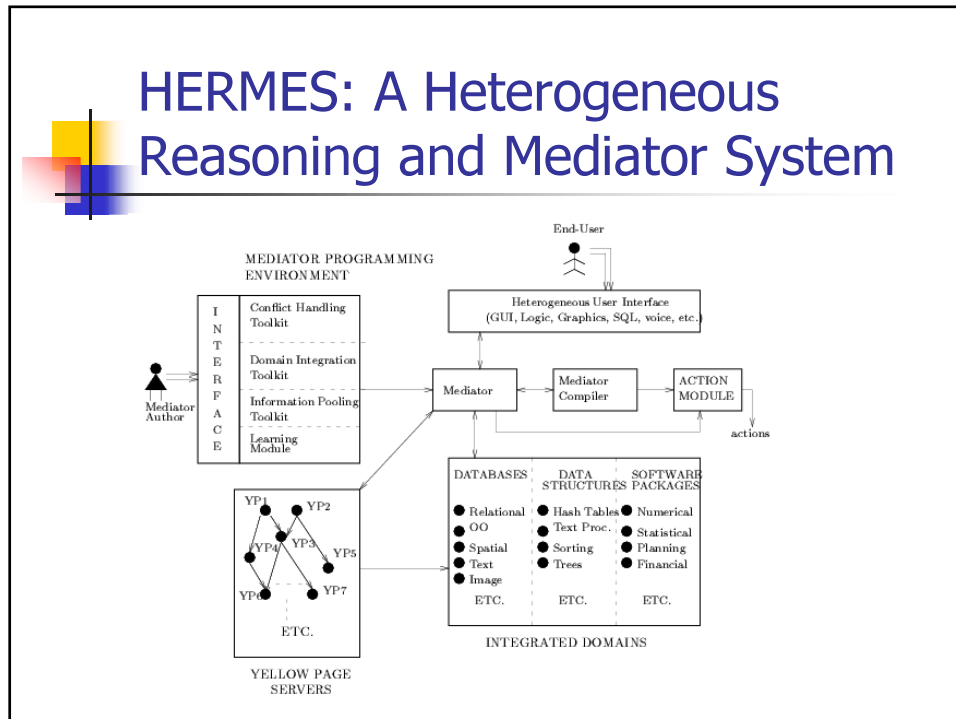
```
o2: <topic, str, "Algorithms">
```



Summary

- Translators convert all data into object model
- Ability to combine and manipulate results in the mediator in arbitrary ways
- The hard part of the is defining the mediators
- The integration plan is up to the mediator designer

HERMES: A Heterogeneous Reasoning and Mediator System



Types of Data Supported

- Relational data
- Spatial data
- Text documents
- Pictorial data
- Object-oriented databases
- Path planning software



Mediator Language

- (=) binary equality – succeeds if both arguments are identical
- (in) succeeds if the first argument is one of the elements of the second
 - $\text{In}(P, \text{PARADOX:project}(\text{'parts'}, \text{color})) \ \& \ \text{=(P.color, "green")}$
 - Succeeds if the parts database contains one green object
- (is) binary equality over variables and sets
 - $\text{Is}(\{\text{"green"}\}, \text{PARADOX:project}(\text{'parts'}, \text{color}))$
 - Succeeds if the parts database contains only green objects



Example Query

- Retrieve any supplier that lies within 50 unites of distance from a given factory and the supplier has enough parts to provide a given quantity of the component.

```
query1(Supplier, Part, Quantity, Factory) : [1, R] ←  
in(Supplier, PARADOX : project(select_>('db1', "qty", Quantity), "name")) &  
in(Loc1, DBASE : project(select_=( 'db2', "name", Supplier), "location")) &  
in(Loc2, DBASE : project(select_=( 'db2', "name", Factory), "location")) &  
in(Loc1, SPATIAL : RANGE('db3', Loc2.x, Loc2.y, 50)).
```



Example Pictoral Query

- Find a picture with both George Bush and the spouse of a person whose tax dealings have been reported in the newspaper

```
p(Person, Rank, Picture) : [1, R] ←  
  in("George Bush", PICTUREDB : feature(File)) &  
  in(OtherPerson, PICTUREDB : feature(File)) &  
  in(Spouse, PARADOX : project(select=( 'spouse', Spouse1, OtherPerson), Spouse)) &  
  news(Spouse, Article) : [1, R] &  
  news("taxes", Article).
```



Conflict Resolution Strategies

- Latest data preference strategy
 - All data is time stamped
 - Preference given to most recent data
- Predicate preference strategy
 - Specified preference between two relations when there is a conflict
- Object preference strategy
 - Specified preference over an entire object
- Value-based preference strategy
 - Preference based on the actual values
 - Can prefer the highest of two numeric fields
- Criticality number strategy
 - Number represents reliability and in the case of conflicts prefer the more reliable source



Summary

- Hermes provides a toolkit for constructing mediators out of heterogeneous source types
- Integration occurs in the mediator predicates and queries
- Strengths
 - Wide variety of source types
 - Good tools for resolving inconsistencies



Local-as-View Mediator Systems

- Coming next week...