On the integration of biomedical knowledge bases: problems and solutions

Luca Vassalli

lucanl@star.dist.unige.it

Systems and Technologies for Automated Reasoning laboratory, DIST, University of Genoa
A collaboration between:

- Systems and Technologies for Automated Reasoning laboratory, DIST, University of Genoa
- Bioengineering and Bioimages laboratory (Biolab), DIST, University of Genoa

Brief introduction to the problem

Our research goal

The different possible solutions

BioGIS (Bioinformatic GAV Integration System)

- Rewriting rules
- Front end
- Internal structure

Conclusions
Data Sources Integration

“The user should be able to focus on what he is looking for rather than thinking how to obtain it” (A. Levy)

- Issues:
  - Overlapping and mismatching
  - Syntactic difference between sources
  - Different layout of the sources (chart based, text based, etc.)
  - Lacking of a common exchange format
  - Unknown data source internal structure
  - Internet is not a stable environment
  - Sometimes hard identifying the same element in different systems
BioGIS

The goal:
- Integration of the human metabolic pathways

The sources:
- KEGG (M. Kanehisa et al., 2002)
- Reactome (G. Joshi-Tope et al., 2005)

The user:
- Biolab portal (http://grid.bio.dist.unige.it)
Modelling the data sources

Global as view (Garcia-Molina et al., 1997)

- Two data sources:
  - DB1 (Pathway_Name, Pathway_ID1, Description, Molecule)
  - DB2 (Pathway_ID2, Pathway_Name, Organism)

- Mediated schema relations:
  - Pathway (Pathway_Name, Description, Organism) :-
    DB1(Pathway_Name,Pathway_ID1, Description, Molecule),
    DB2(Pathway_ID2, Pathway_Name, Organism)
  - Connection_Molecule (Pathway_Name, Molecule) :-
    DB1(Pathway_Name,Pathway_ID1, Description, Molecule)
Modelling the data sources

Local as view (O. Duschka et al., 1997)

- DB1 (Pathway_Name, Pathway_ID1, Description, Molecule) :-

  Pathway (Pathway_Name, Description, Organism, Pathway_ID1, Pathway_ID2),
  Connection_Molecule (Pathway_Name, Molecule, Class),
  Class = “genes”

- DB2 (Pathway_ID2, Pathway_Name, Organism) :-

  Pathway (Pathway_Name, Description, Organism, Pathway_ID1, Pathway_ID2),
  Organism = “homo sapient”
A Comparison

- **GAV**
  - Does not require containment checking (fast and reliable)
  - Somehow awkward modelling the system
  - Difficult to extend

- **LAV**
  - Easy to extend
  - Useless details in the model of the system
  - Requires containment checking (slow)
  - The algorithm may be even intractable

- **GLAV (M Friedman et al., 1999)**
  - Same complexity than LAV
  - Solved some drawbacks in the modelling phase
BioGIS

- **Front end or ad hoc methods**
- **Execution engine which iteratively calls the wrappers**
- **A wrapper for each data source**
- **Integration engine**

Diagram:
- Ad hoc method call
- Front end
- Execution engine
  - Reactome wrapper
  - KEGG wrapper
  - Integration engine
    - Reactome WS
    - KEGG WS
    - Query Answer
The information extracted

- Two ad hoc family of methods:
  - getMoleculesForPathway
  - getPathwayForMolecules

- Three global schema relations:
  - Pathway
  - Connection_Molecule
  - Reaction
Front End

Queries have to follow a precise grammar

Examples:
- PATHWAY { GOTerm = "alanine metabolism" } END
- PATHWAY { ReactomePathwayID = "109606" } , CONNECTION MOLECULE { ReactomePathwayID = "109606" } END
- CONNECTION MOLECULE { UniqueID = "Q92934" } END
Internal structure

- **Execution engine:**
  - Simple unfolding of the queries according to the GAV methodology
  - Ad hoc methods: concurrent threads which query in parallel the wrappers

- **Wrappers:**
  - A class for every different data source relation. The information is retrieved from the sources and structured into objects.

- **Integration engine:**
  - Pathways merged using the pathway names and the Gene Ontology terms
  - Molecules merged using the UniProt and COMPOUND ids
Performances

- Vary according to several factors:
  - The number of hits of the query
    - “Retrieve all the genes that take part to a pathway which matches the keyword “pyruvate””: around 65 hits – 1 minute
    - “Retrieve all the genes that take part to a pathway which matches the keyword “metabolism””: thousands of hits – half an hour
  - The state of the Reactome cache
  - The network latency

- Better to be used in a chain of web services than as a standalone service available through a browser
Conclusions

- **GAV approach:**
  - Yet possible easy extensions of the wrappers thanks to the modelling of the same knowledge base as more relations
  - Good approach in case of few stable sources and limited extension

- **Web service approach**

- **Future work:**
  - Extension to allow a more expressive grammar
  - Extension to another data source (BioCyc)
  - Extension to take advantage also XML format together with web services
Thanks for your kind attention
Any question?

Contact me: lucanl@star.dist.unige.it

Other contacts:
E. Giunchiglia: giunchiglia@unige.it
I. Porro: pivan@dist.unige.it
M. Fato: fantomas@dist.unige.it
The grammar

- **goal** → relations END
- **relations** → relation rel'
- **Rel'** → , relation rel
  | ε
- **relation** → namerelation { bindings }
- **Namerelation** → PATHWAY
  | CONNECTION MOLECULE
  | REACTION
- **bindings** → binding bin'
- **bin'** → , binding bin'
  | ε
- **binding** → string = “ string ”
- **string** → [azA-Z0-9[ ] +, ]
The global schema: Pathway

- Pathway (PathName, KEGGPathwayID, ReactomePathwayID, Description, Organism, GOTerm) :- KEGG1 (PathName, KEGGPathwayID, Organism), Reactome1 (PathName, ReactomePathwayID, Description, Organism, GOTerm)

- Pathway (PathName, KEGGPathwayID, ReactomePathwayID, Description, Organism, GOTerm) :- KEGG1 (PathName, KEGGPathwayID, Organism),

- Pathway (PathName, KEGGPathwayID, ReactomePathwayID, Description, Organism, GOTerm) :- Reactome1 (PathName, ReactomePathwayID, Description, Organism, GOTerm)
Connection_Molecule (ReactomePathwayID, KEGGPathwayID, ReactomeMoleculeID, MoleculeNameR, KEGGMoleculeID, MoleculeNameK, UniqueID, Database, Definition, Class, Description) :- Reactome3 (ReactomePathwayID, ReactomeMoleculeID, MoleculeNameR, UniqueID, Database), KEGG2 (KEGGMoleculeID, MoleculeNameK, UniqueID, Definition, Class, Description), KEGG3 (KEGGPathwayID, KEGGMoleculeID, Class)

Connection_Molecule (ReactomePathwayID, KEGGPathwayID, ReactomeMoleculeID, MoleculeNameR, KEGGMoleculeID, MoleculeNameK, UniqueID, Database, Definition, Class, Description) :- Reactome3 (ReactomePathwayID, ReactomeMoleculeID, MoleculeNameR, UniqueID, Database)

Connection_Molecule (ReactomePathwayID, KEGGPathwayID, ReactomeMoleculeID, MoleculeNameR, KEGGMoleculeID, MoleculeNameK, UniqueID, Database, Definition, Class, Description) :- KEGG2 (KEGGMoleculeID, MoleculeNameK, UniqueID, Definition, Class, Description), KEGG3 (KEGGPathwayID, KEGGMoleculeID, Class)
The global schema: Reaction

Reaction (PathName, ReactomePathwayID, Reaction) :-
Reactome1 (PathName, ReactomePathwayID, Description, Organism, GOTerm),
Reactome2 (ReactomePathwayID, Reaction)