



Data Access & Integration in the ISPIDER Proteomics Grid

N. Martin – A. Poulouvassilis – L. Zamboulis

{[nigel](mailto:nigel@dc.s.bbk.ac.uk),[ap](mailto:ap@dc.s.bbk.ac.uk),[lucas](mailto:lucas@dc.s.bbk.ac.uk)@dc.s.bbk.ac.uk}



Project Details

- Members
 - Birkbeck College
 - European Bioinformatics Institute
 - University of Manchester
 - University College London

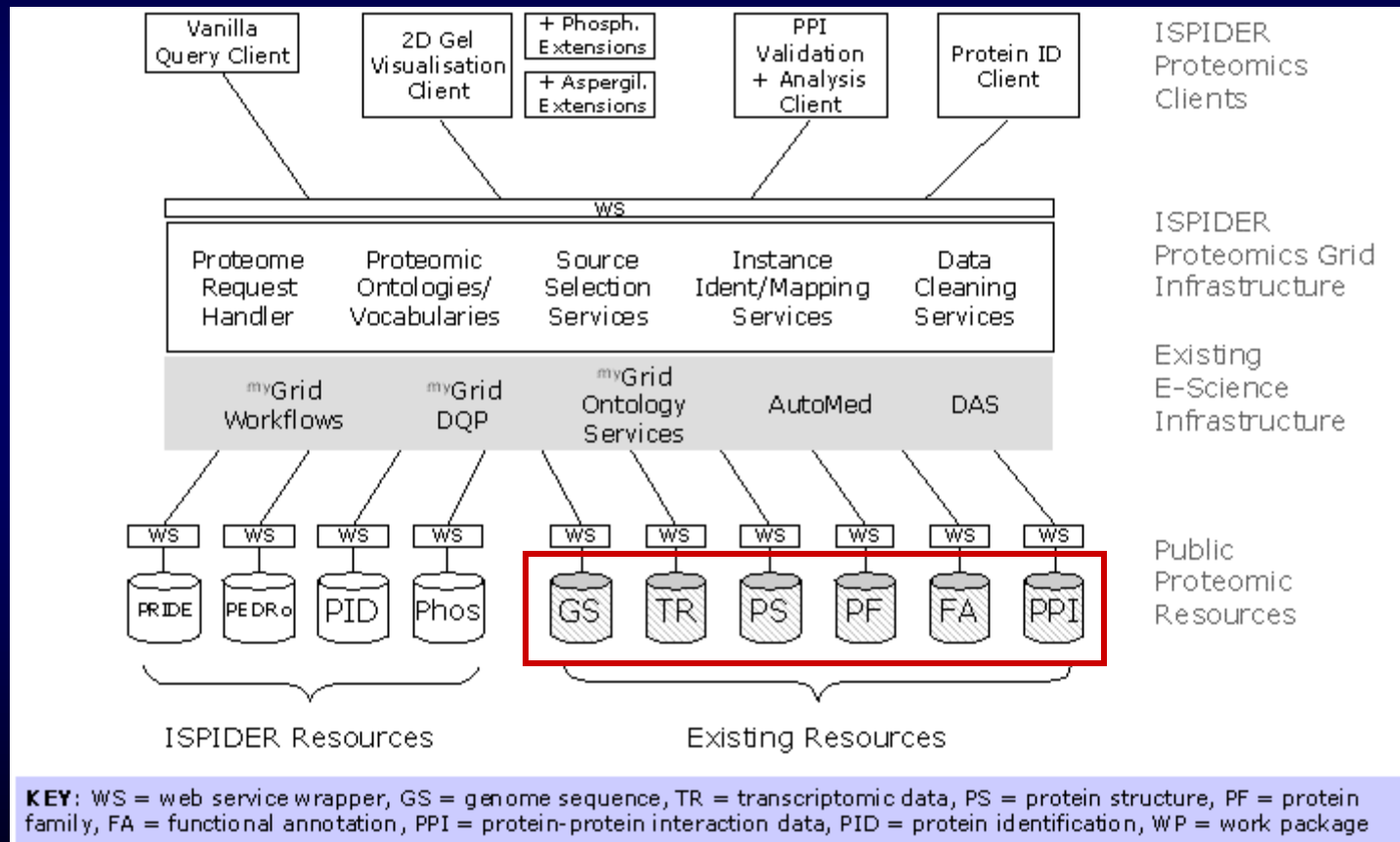


Problem Definition

- Biological repositories
 - In separate locations: interoperability problems
 - Rapidly updated/modified/evolved
 - Overlapping data
 - Need processing power

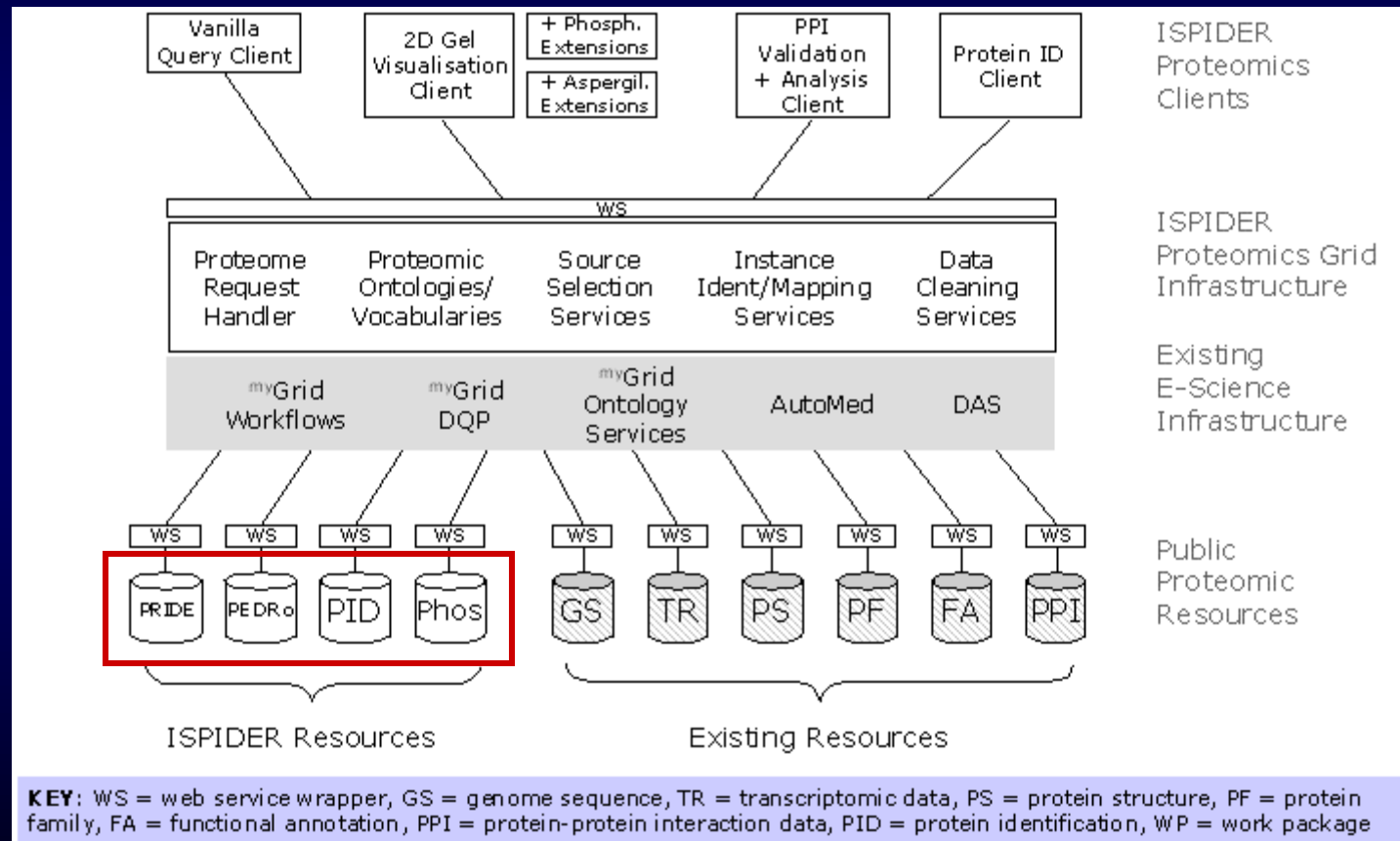


ISPIDER Objectives



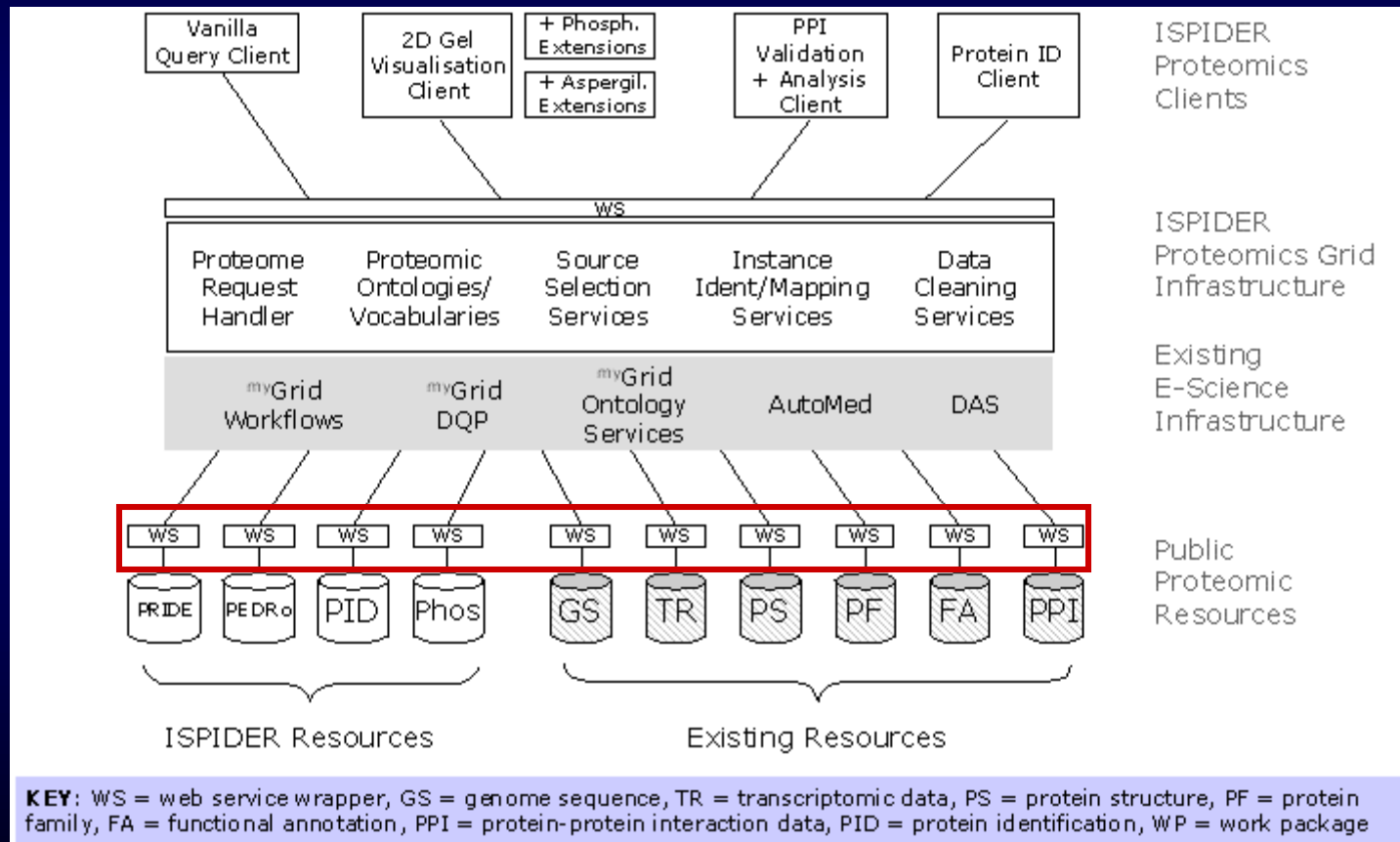


ISPIDER Objectives



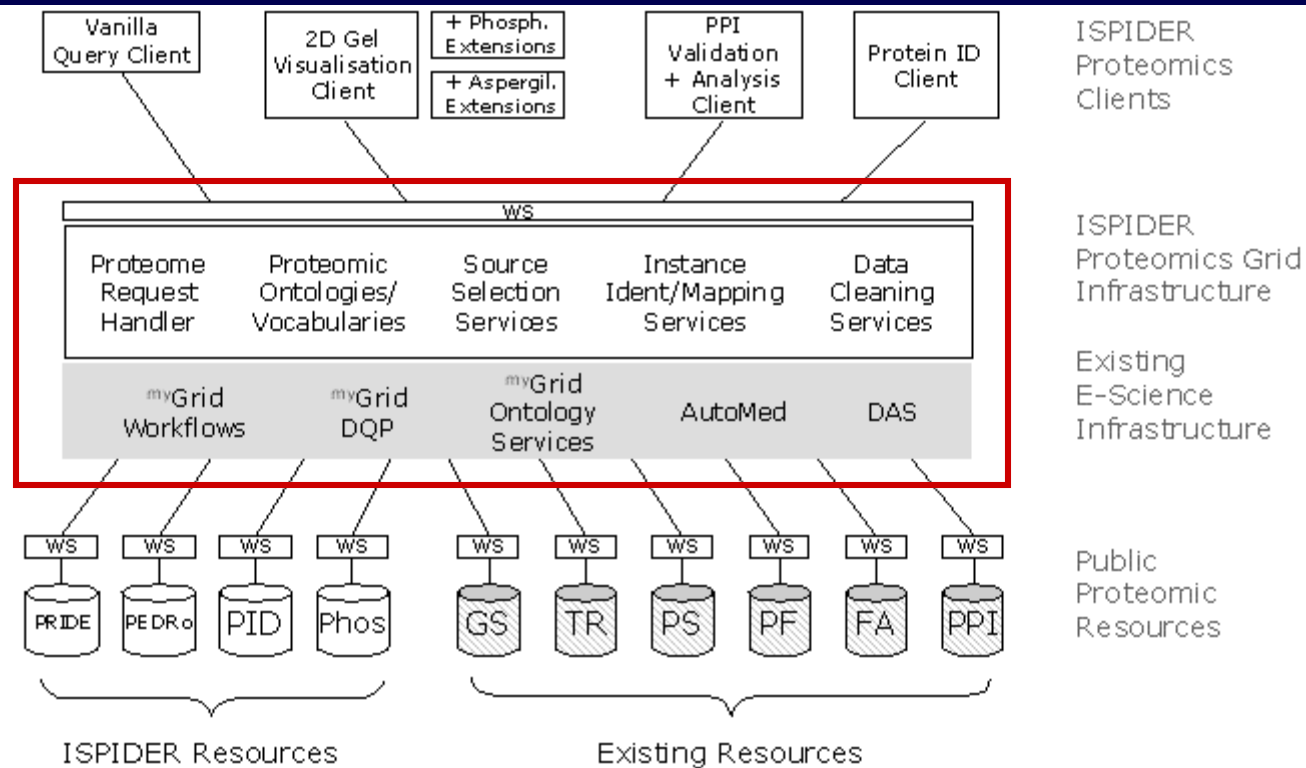


ISPIDER Objectives





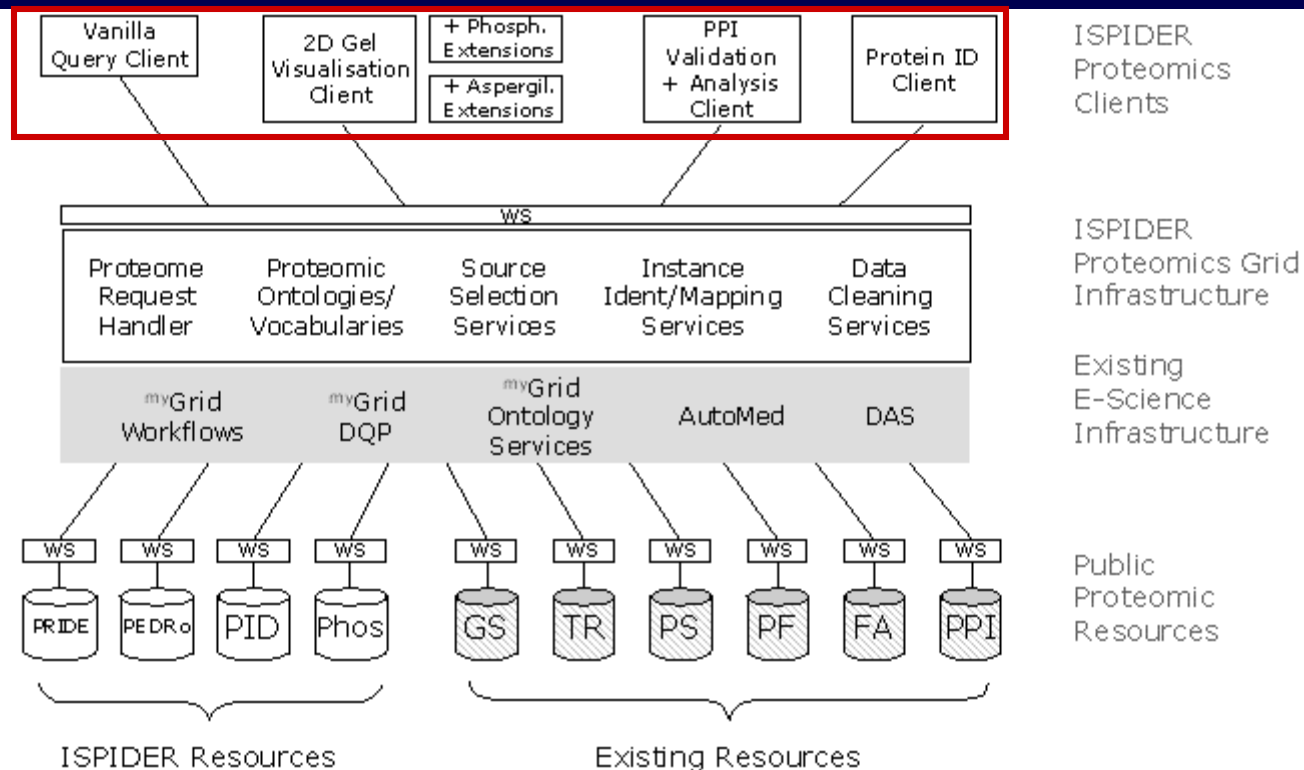
ISPIDER Objectives



KEY: WS = web service wrapper, GS = genome sequence, TR = transcriptomic data, PS = protein structure, PF = protein family, FA = functional annotation, PPI = protein-protein interaction data, PID = protein identification, WP = work package



ISPIDER Objectives



KEY: WS = web service wrapper, GS = genome sequence, TR = transcriptomic data, PS = protein structure, PF = protein family, FA = functional annotation, PPI = protein-protein interaction data, PID = protein identification, WP = work package



Middleware (1/2)

- myGrid: collection of services/components allowing high-level integration of data/applications
 - Taverna Workbench
- AutoMed heterogeneous data integration system
- OGSA-DAI
- OGSA-DQP

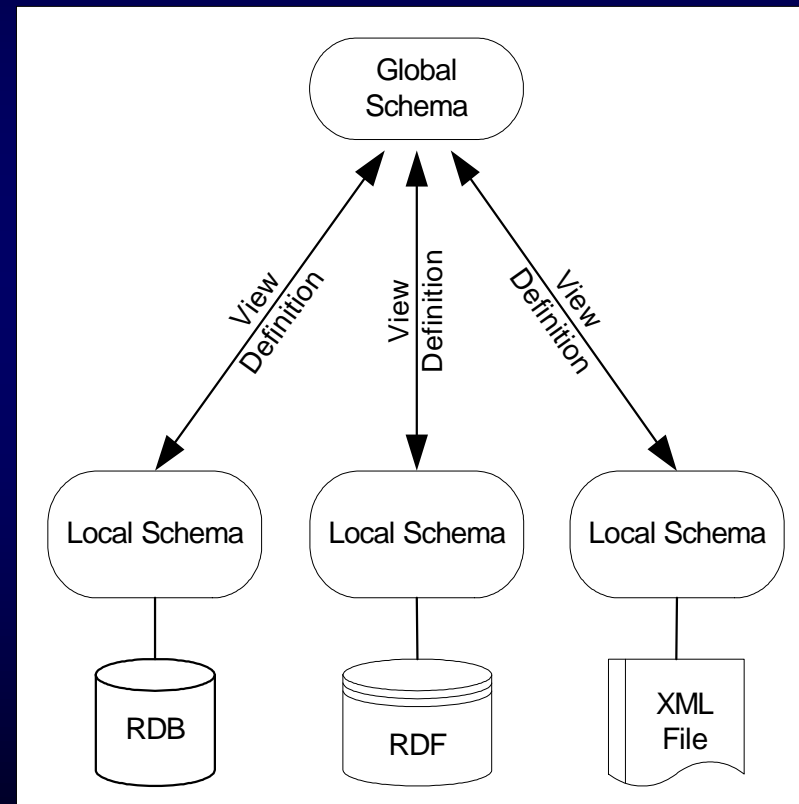


Middleware (2/2)

- AutoMed toolkit: heterogeneous data integration system - developed by Birkbeck College/Imperial College
 - Subsumes traditional data integration approaches
 - Handles various data models – easily extensible
 - Virtual/materialised/ hybrid integration
 - Data warehousing tools
 - Schema evolution

GAV & LAV Approaches

- Global-As-View (GAV) approach: describe GS constructs with view definitions over LS_i constructs
- Local-As-View (LAV) approach: describe LS_i constructs with view definitions over GS constructs





GAV Example

S_g student(id, name, left#, degree)
monitors(sno, id)
staff(sno, sname, dept#)

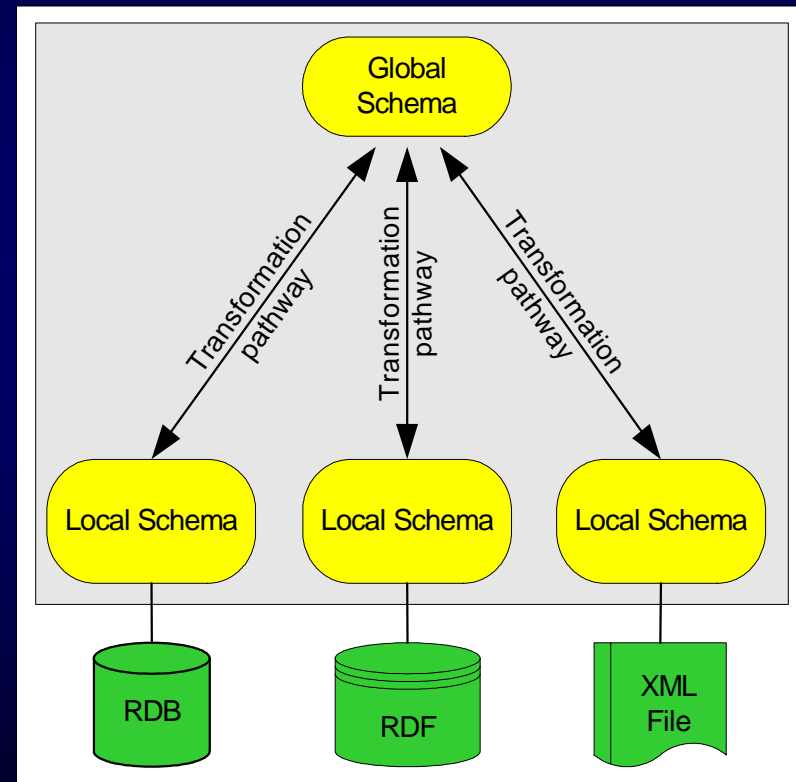
S_1 ug(id, name, left#, degree, sno)
tutor(sno, sname)

S_2 phd(id, name, left#, title)
supervises(sno, id)
supervisor(sno, sname, dept)

- student(id, name, left, degree) =
[{x, y, z, w} | $\langle x, y, z, w, _ \rangle \in \text{ug} \wedge$
 $\langle x, _, _, _, _ \rangle \notin \text{phd} \vee$
 $\langle x, y, z, w, _ \rangle \in \text{phd} \wedge$
 $w = \text{'phd'}$]
- monitors(sno, id) =
[{x, y} | $\langle x, _, _, _, y \rangle \in \text{ug} \wedge$
 $\langle x, _, _, _, _ \rangle \notin \text{phd} \vee$
 $\langle x, y \rangle \in \text{supervises}$]
- staff(sno, sname, dept) =
[{x, y, z} | $\langle x, y, z, w, _ \rangle \in \text{tutor} \wedge$
 $\langle x, _, _ \rangle \notin \text{supervisor} \vee$
 $\langle x, y, z \rangle \in \text{supervisor}$]

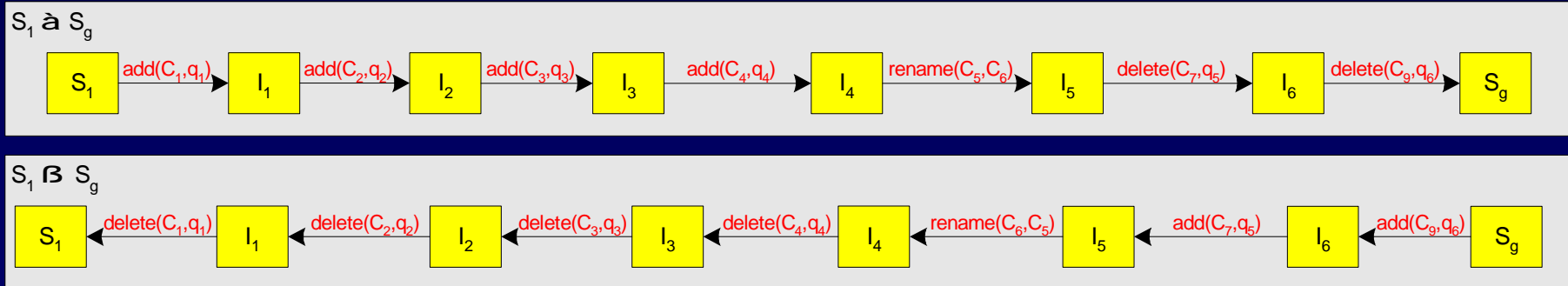
Both-As-View (BAV)

- Schema transformation approach
- For each pair (LS_i, GS) : incrementally modify LS_i/GS to match GS/LS_i





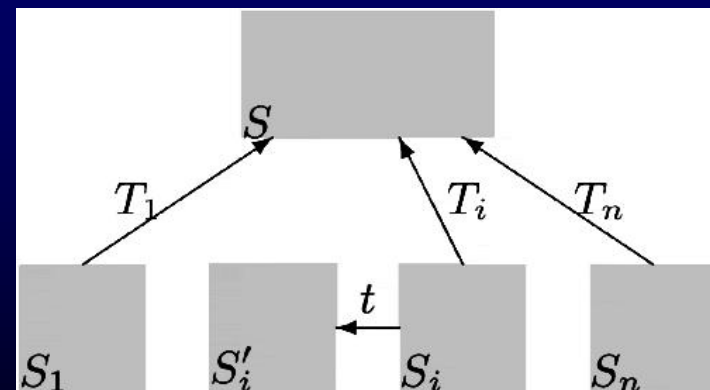
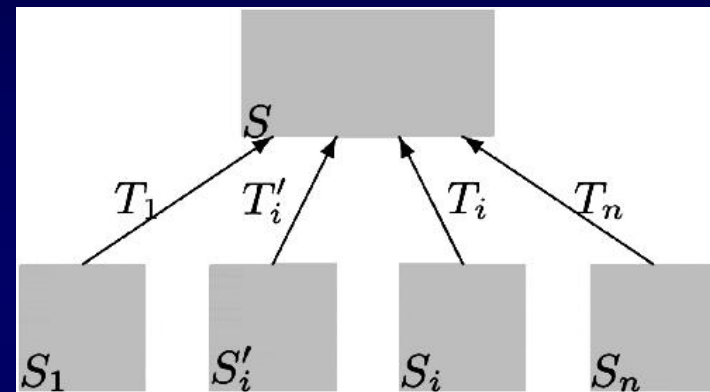
BAV Example



- Transformation pathway consists of primitive transformations
- Pathway contains both GAV & LAV definitions
- Transformations are automatically reversible
- Metadata in AutoMed Repository

Schema Evolution Example

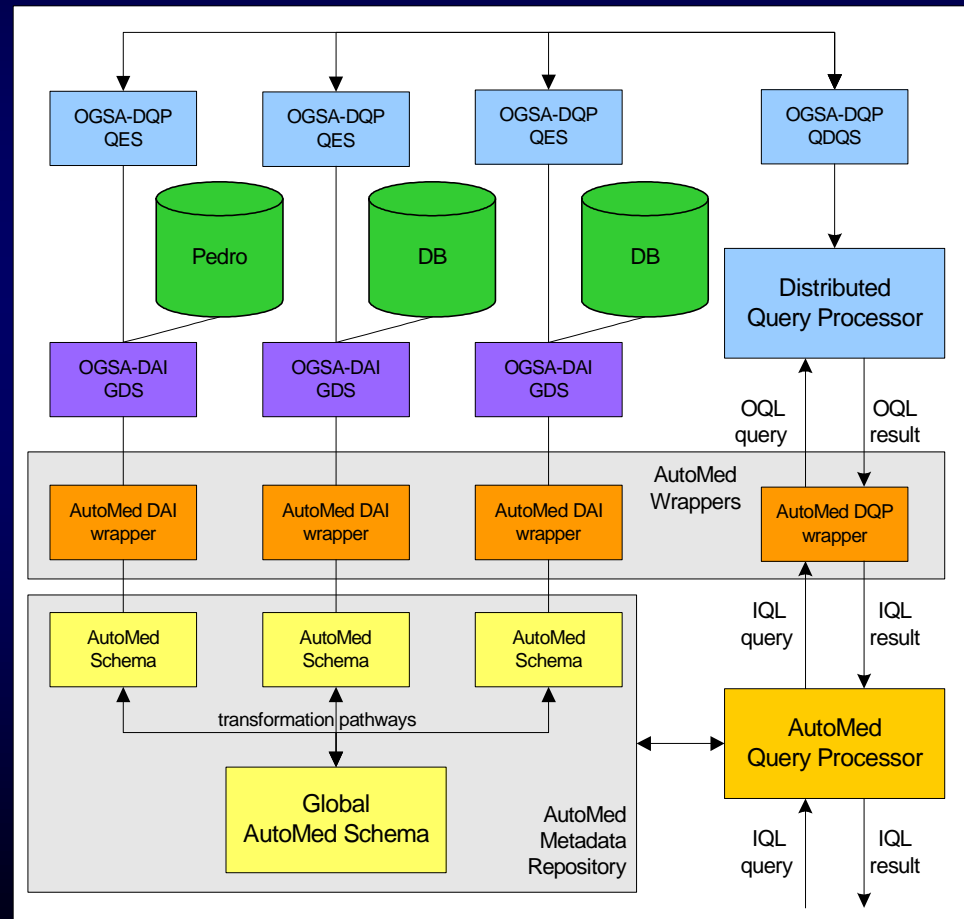
- No need to redefine pathway
- Instead, simply describe the evolution as a pathway
- Automatic in most cases





Interoperability

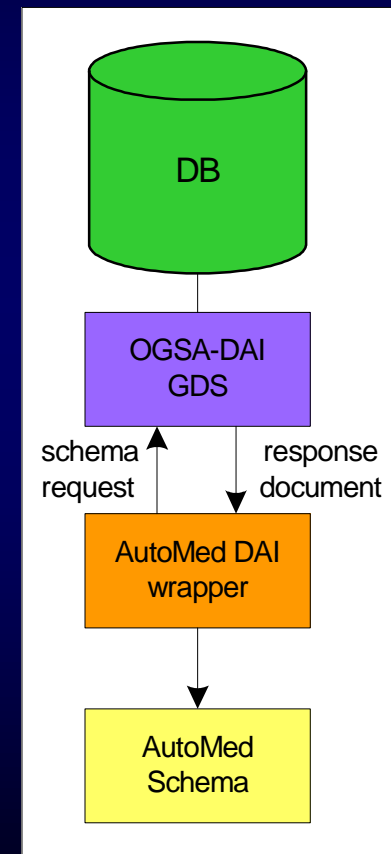
- Sources wrapped with OGSA-DAI
- AutoMed wrappers extract source metadata
- Integration using AutoMed
- Queries submitted:
 - Reformulated using AutoMed metadata
 - Submitted to OGSA- DQP





Schema extraction

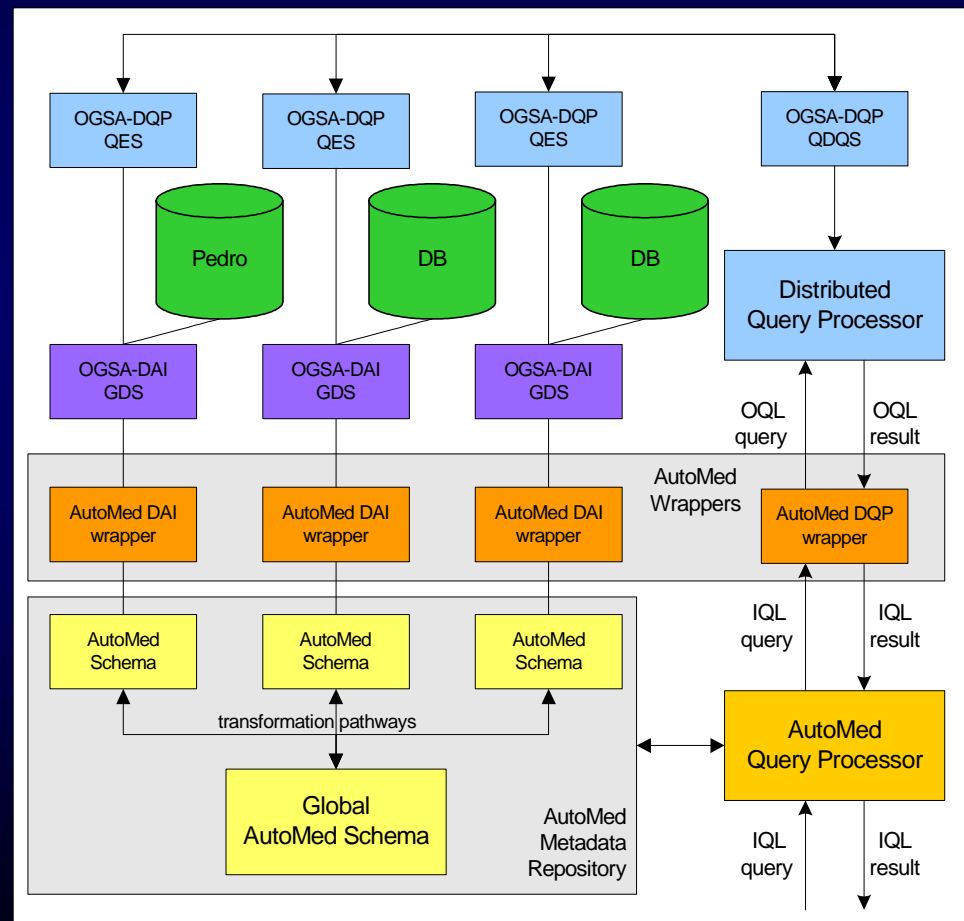
- AutoMed wrapper requests the schema of the data source using an OGSA-DAI service
- The service replies with the source schema encoded in XML
- The AutoMed wrapper creates the corresponding schema in the AutoMed repository





Query Processing

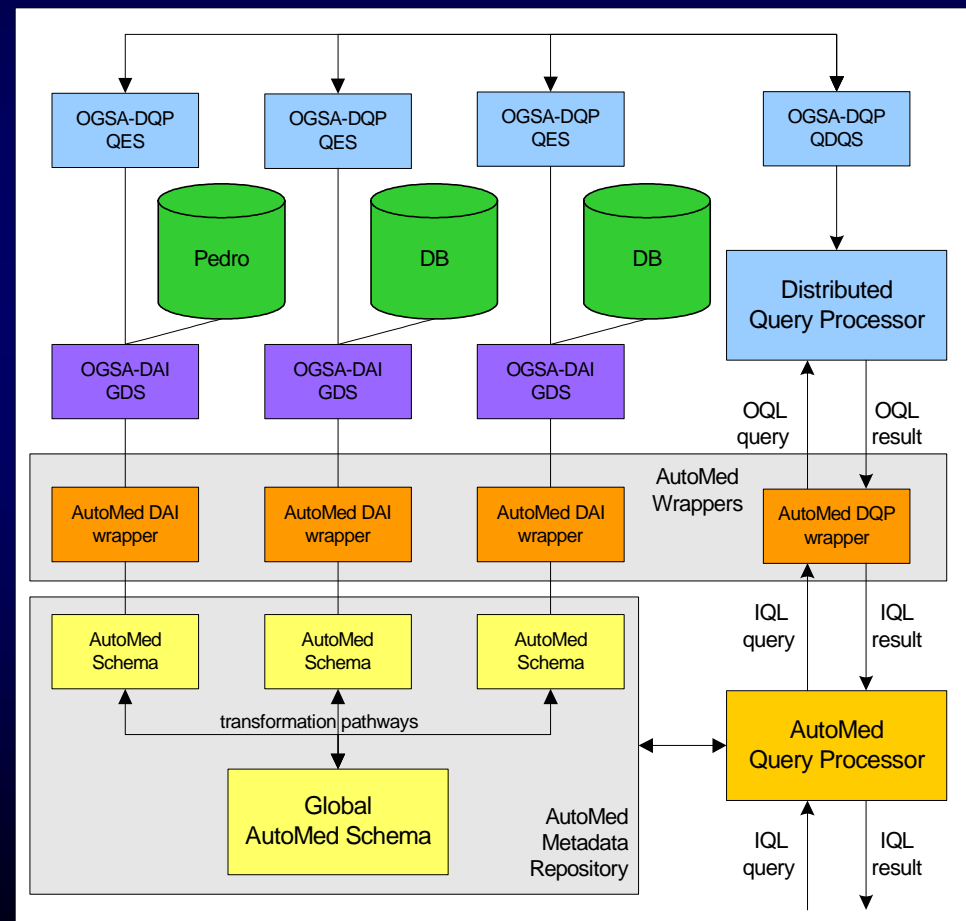
- Query is submitted to AutoMed's GQP:
 - Reformulated
 - Optimised
- AutoMed-DQP Wrapper:
 - IQL à OQL
 - Submits OQL to OGSA-DQP
 - OQL result à IQL result





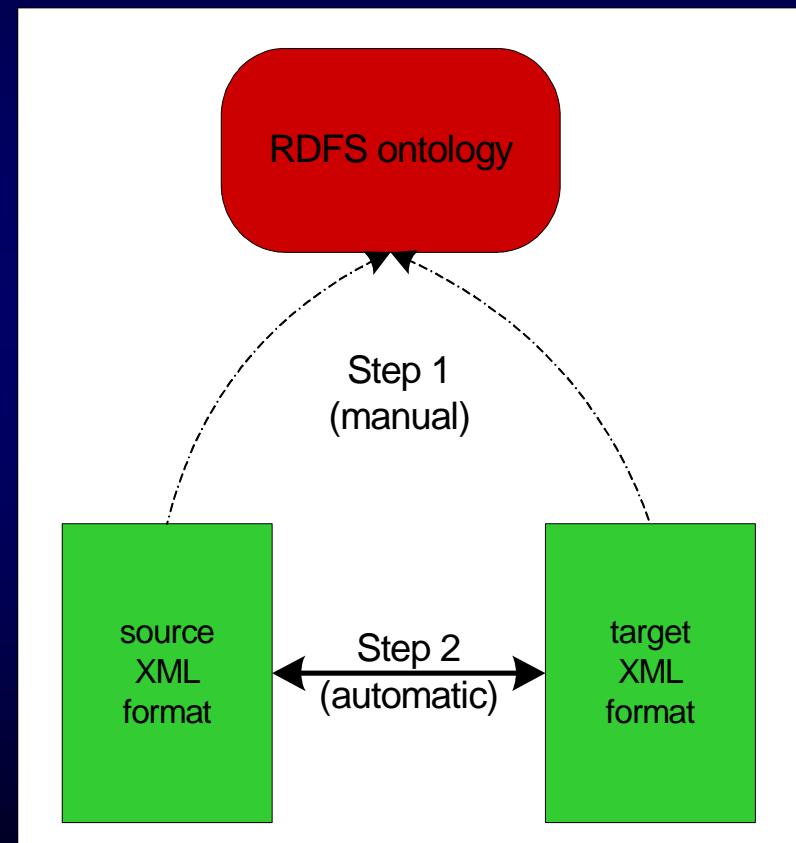
Query Processing

- OGSA-DQP:
 - Evaluates OQL query using QES
 - Sends OQL result back to AutoMed DQP Wrapper: OQL result → IQL result



Future Work

- Workflow integration
 - AutoMed toolkit
 - Taverna Workbench
 - Integration of services with XML input/output
 - LAV Mappings from XML to one or more RDFS ontologies





Future Work

- AutoMed extensions:
 - Web/Grid Services for AutoMed
 - Data warehousing
 - Materialised/hybrid integration
 - Data provenance
 - Incremental view maintenance
 - Schema evolution



Summary

- ISPIDER aims to:
 - Create an integrated platform of proteomic resources
 - Use existing resources – produce new ones
 - Create clients for querying, visualisation, etc.
- ISPIDER is using:
 - myGrid – middleware for in silico experiments in biology
 - OGSA-DQP – service-based distributed query processor
 - AutoMed – heterogeneous data integration system



Project Members

- Birkbeck College
 - Nigel Martin
 - Alex Poulouvassilis
 - Lucas Zamboulis (R.A.)
 - Hao Fan (former R.A.)
- European Bioinformatics Institute
 - Rolf Apweiler
 - Henning Hermjakob
 - Weimin Zhu
 - Chris Taylor
 - Phil Jones
 - Nisha Vinod
- University of Manchester
 - Simon Hubbard
 - Steve Oliver
 - Suzanne Embury
 - Norman Paton
 - Carol Goble
 - Robert Stevens
 - Khalid Belhajjame (R.A.)
 - Jennifer Siepen (R.A.)
- U.C.L.
 - David Jones
 - Christine Orengo
 - Melissa Pentony (R.A.)