# DataFoundry's Meta-Data Infrastructure

**Terence Critchlow**

*Ron Musick* *Madhavan Ganesh*

**Center for Applied Scientific Computing**

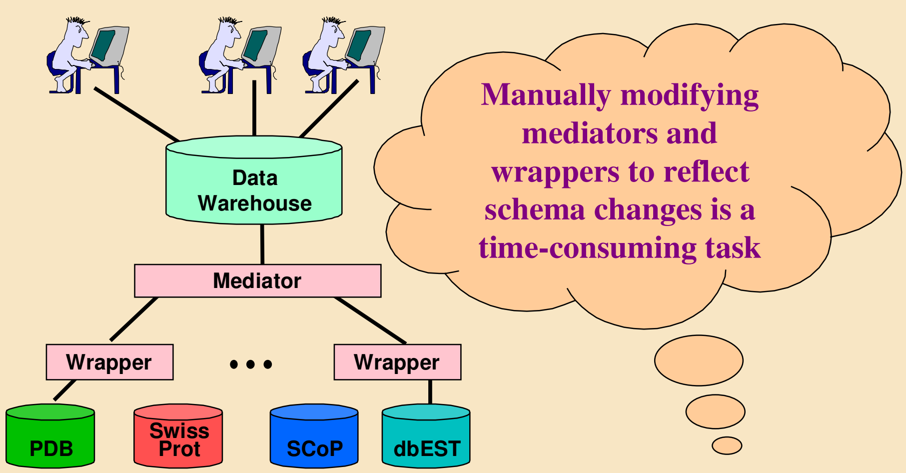
*Krzysztof Fidelis* *Tom Slezak*

**Biology and Biotechnology Research Program**

**Lawrence Livermore National Laboratory**

CASC

# Based on a mediated warehouse architecture

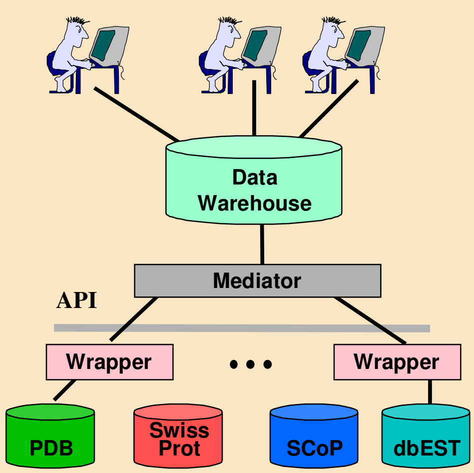



CASC

## Three techniques to reduce maintenance costs

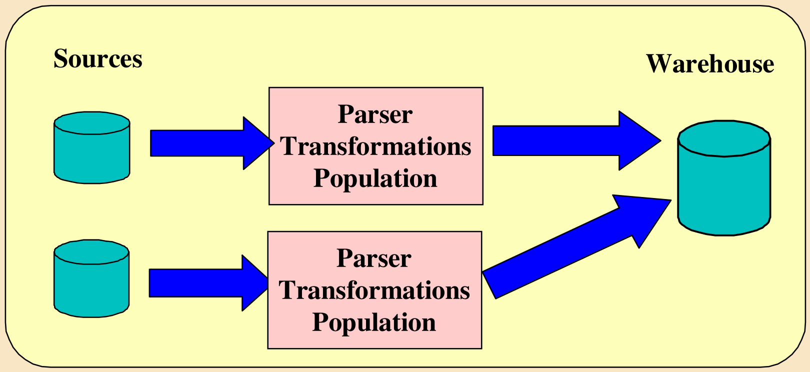
- Schema evolution techniques.
  - automating updates when source schema change
    - ∗context-based correspondence identification
    - ∗reason about changes in a data source based on old meta-data
- Schema integration techniques.
  - automating new source integration into warehouse
    - ∗automatic correspondence identification
    - ∗identify relationships bases on names and meta-data
- **Data integration techniques.**
  - **automating how data is loaded into the database**
    - **∗automatic mediator generation**
    - **∗relate source representation to warehouse representation**

## Key Insight

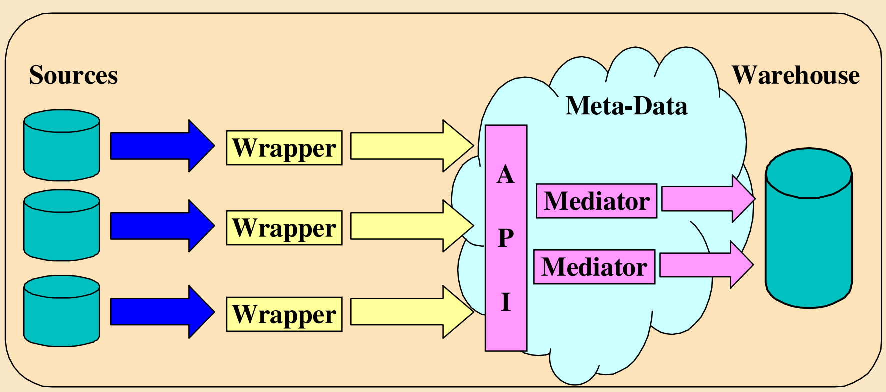



**Extensive use of meta-data can dramatically reduce maintenance costs.**

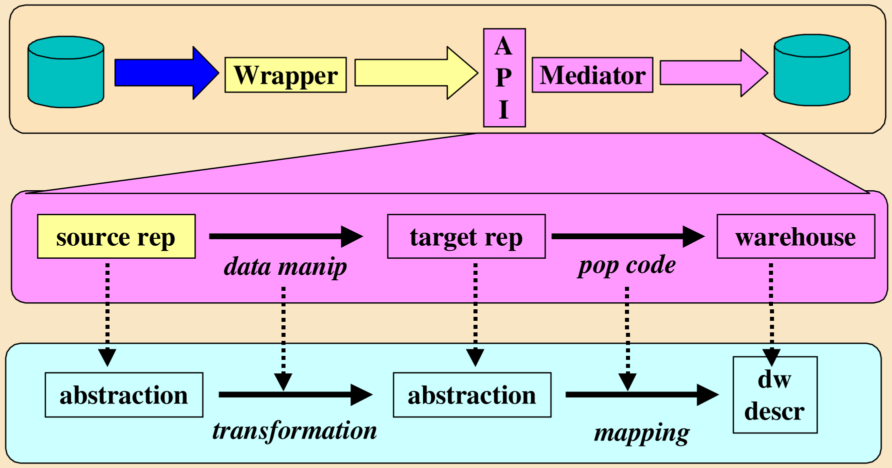
## The typical approach:




## The DataFoundry approach:

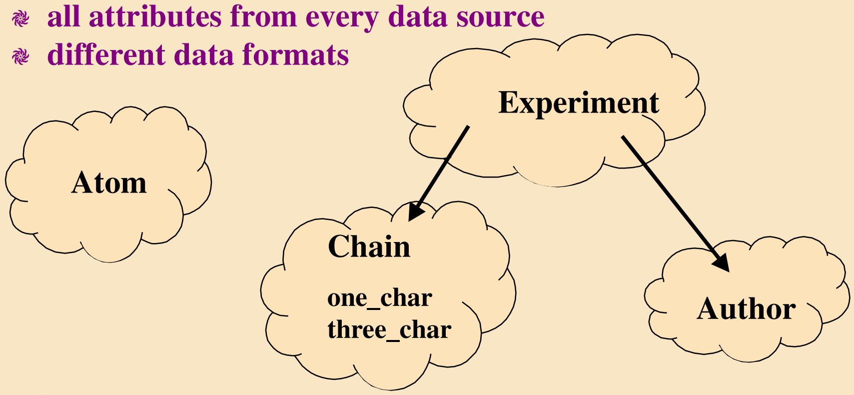


## Four types of meta-data required



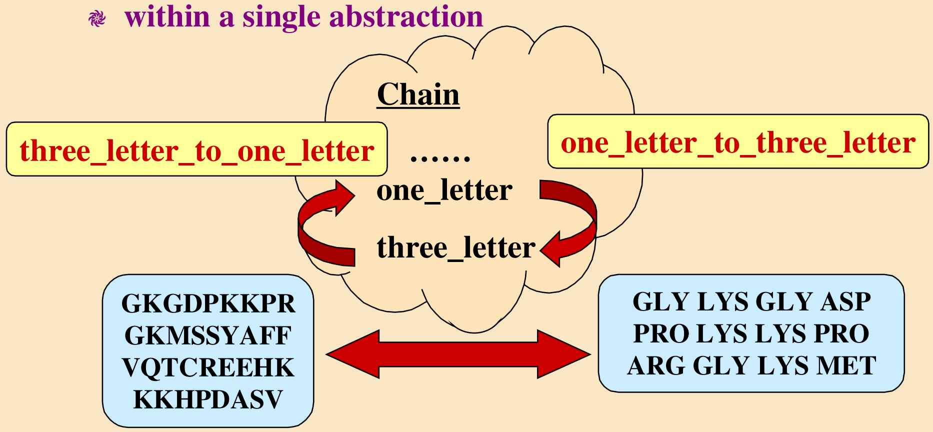

## Abstractions

- Collection of well-defined, domain specific objects
- Includes all information about these objects
  - all attributes from every data source
  - different data formats



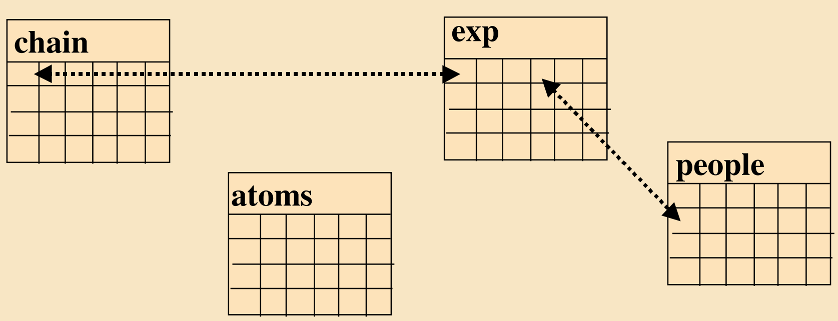
## Transformations

- Convert between different data formats
  - within a single abstraction



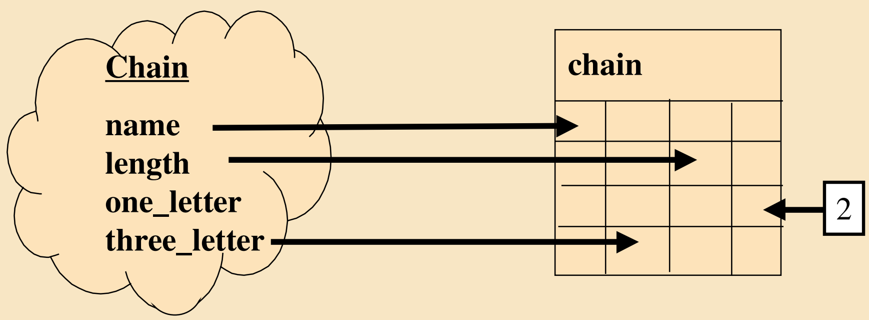

## Database descriptions

- Source independent schema definition
  - very similar to a relational schema
  - able to define complex objects
  - comment field

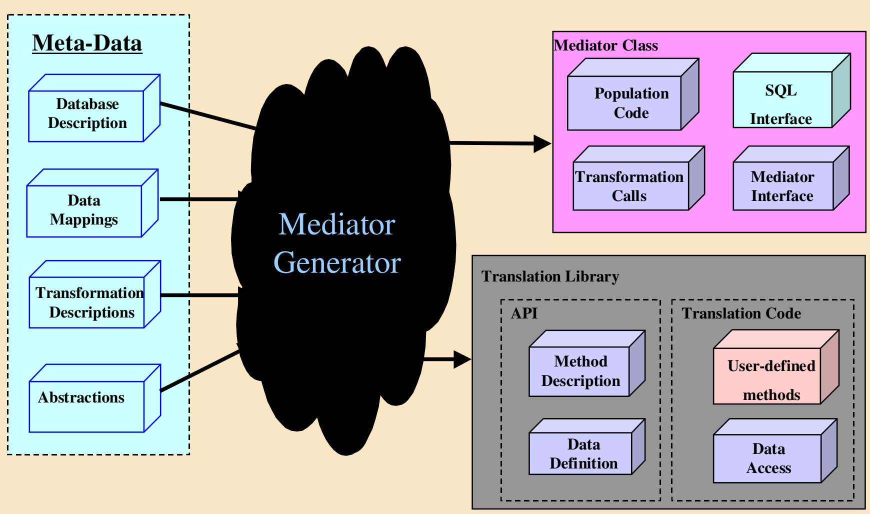


## Mappings

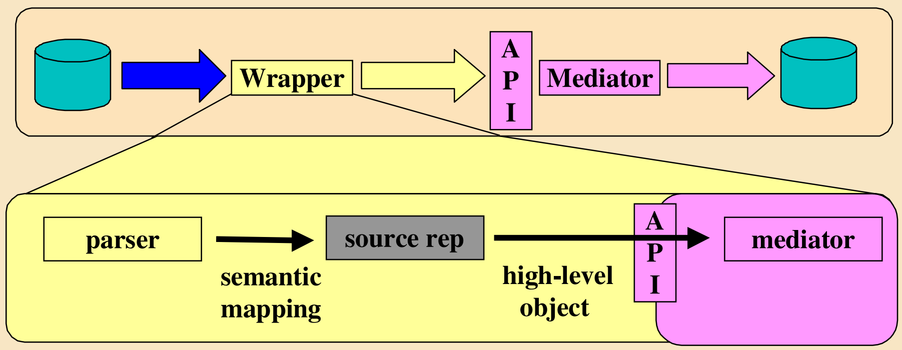
- Relationship between abstractions and database descriptions
  - always a direct attribute to attribute correspondence




## Generating the mediators



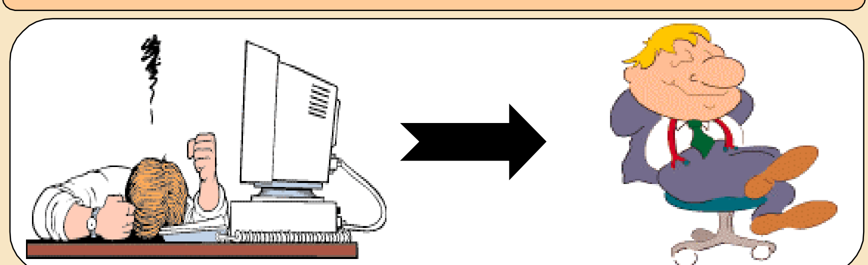
## The translation library and mediator class are used by the wrapper




## Benefits:

- Explicitly defined computer readable meta-data.
- Automatically generated mediator and API.
- Simplified wrapper creation and maintenance.

**Overall reduction in integration and evolution costs.**



## Results:

Integrating SCoP into warehouse that already contains PDB and SWISS-PROT.

| Activity/ integration style | manual | meta-data | diff | %diff |
|---|---|---|---|---|
| understanding SCOP | 2.0 | 2.0 | 0.0 | 0 |
| writing wrapper | 4.5 | 2.5 | 2.0 | 44% |
| modifying schema | 0.5 | 0.5 | 0.0 | 0 |
| writing mediator | 4.0 | 0.0 | 4.0 | --- |
| modifying meta-data | 0.0 | 1.0 | (1.0) | --- |
| total time in days | 11.0 | 6.0 | 5.0 | 45% |

## Beyond data integration

**Schema integration and evolution techniques need:**

- Descriptions of the source and target.
- Mapping information from previous attempts.
- Semantics associated with attributes / concepts.

**This will provide automatically generated wrapper skeletons.**

## Conclusions

- We have created a meta-data infrastructure that reduces the cost of data integration.
- This meta-data can also be applied to automatic schema integration and evolution techniques.

We have made significant progress in reducing the cost of maintaining a data warehouse.

# Data Analysis in Scientific Warehouses

**Madhavan Ganesh**

*Ron Musick* *Terence Critchlow*

Center for Applied Scientific Computing

*Krzysztof Fidelis* *Tom Slezak*

Biology and Biotechnology Research Program

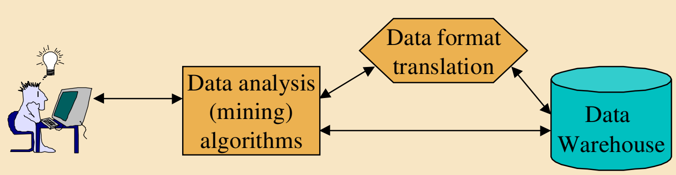
Lawrence Livermore National Laboratory

## Data Access vs Data Analysis

- Data Access
  - Directly expressed data or relations between data items
  - Data source - experiment, transaction, etc.
  - Queries are simple selections and joins in a database
- Data Analysis
  - Implicit/summary property of the expressed data
  - Represents trends and meta-level information
  - Require various ways of organization and exploration
  - Queries are complex with aggregate operations

## General Framework for Scientific Analysis

- Business domain analysis
  - Separates operational and analytical data stores
  - Star schemas are suitable for most analytical queries
- Scientific domain analysis
  - Data more complex and common schema not known
  - Queries are exploratory and may be repeated with different parameter settings to obtain sensible results



## Common Operations in Data Analysis

- Classification
  - Supervised learning method
  - User provides examples of different classes of items
  - Learning system automatically learns features of the different classes
- Clustering
  - Data items are provided to the algorithms along with a distance function which measures similarities of objects
  - Algorithm classifies the objects into groups such that intra-group distances are minimized and inter-group distances are maximized

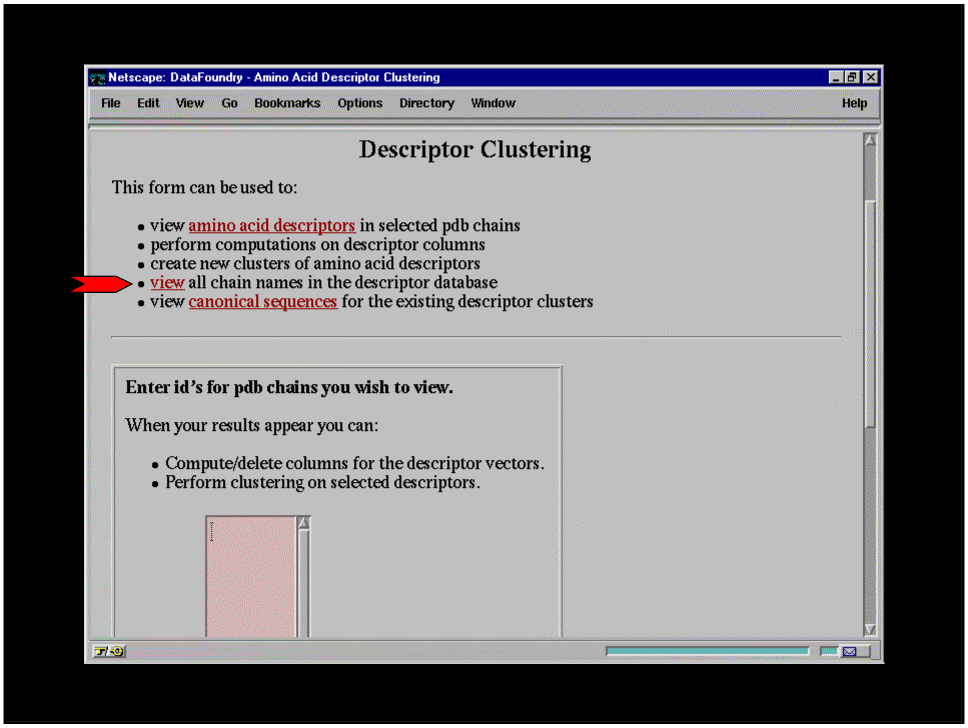
## Data Analysis in DataFoundry

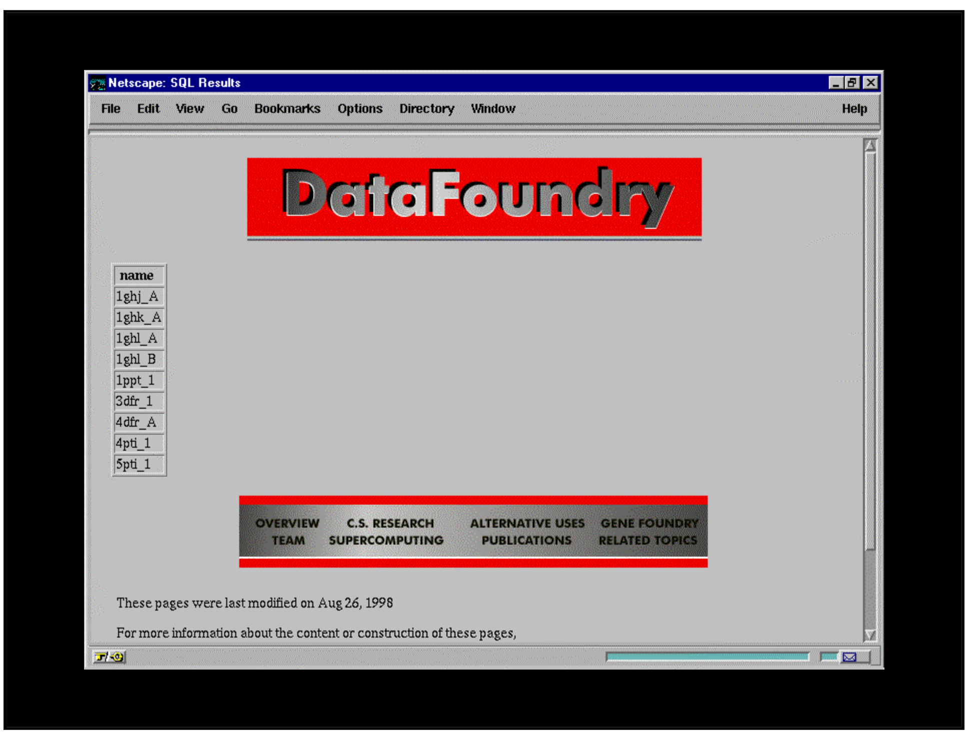
- Implemented interfaces for
  - Clustering
    - Selection of clustering algorithms from pull-down menus
    - Web-based selection of attributes for clustering
    - Derived attributes specified by combining selected attributes
    - Write-back results into warehouse with user annotations
  - Classification
    - Interfaces for selecting classified examples
    - Decision-tree algorithms for learning classification rules

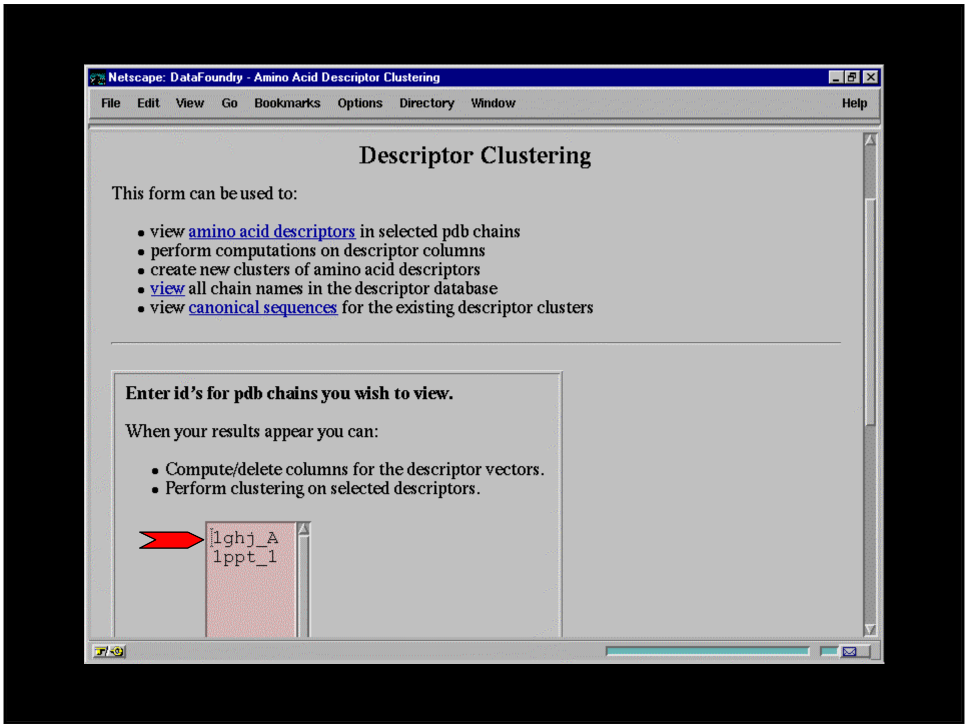
## Protein Structure Analysis

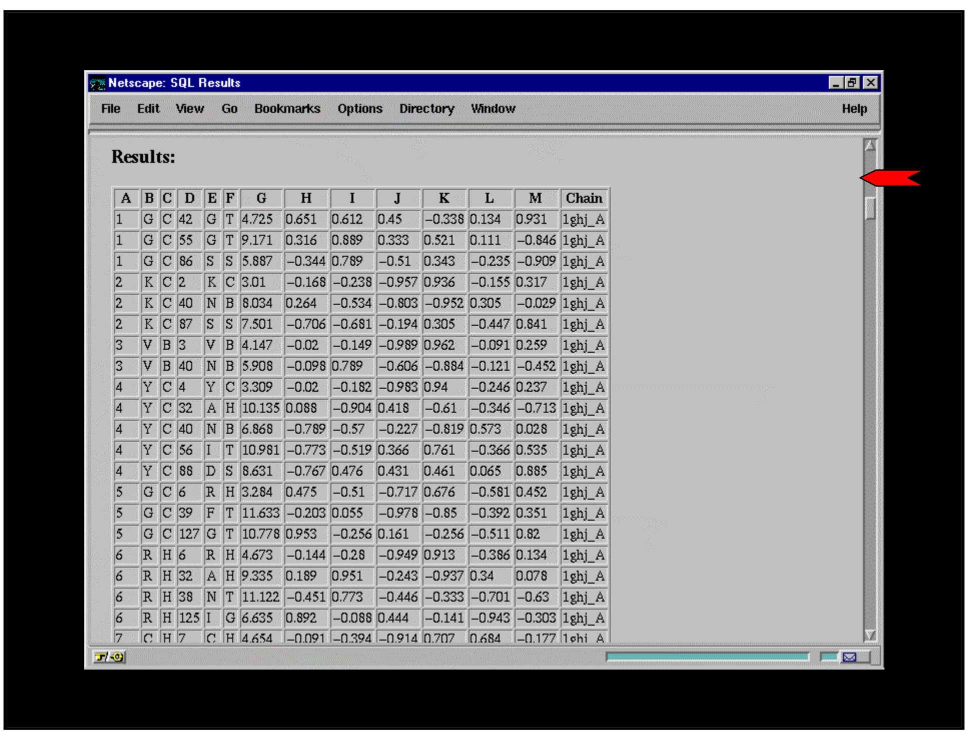
- **Goal: Determine frequent sub-domain structures**
- **Approach:**
  - **Describe local contexts of amino-acids in the sequences**
  - **Cluster descriptors using similarity functions**
  - **Infer canonical sequences for descriptor clusters**
- **Results:**
  - **Initial experiments on clustering done**
  - **Distance functions are being evaluated and improved**
  - **Generated canonical sequences are being evaluated with respect to confidences in the supporting evidence**

## Clustering Interface

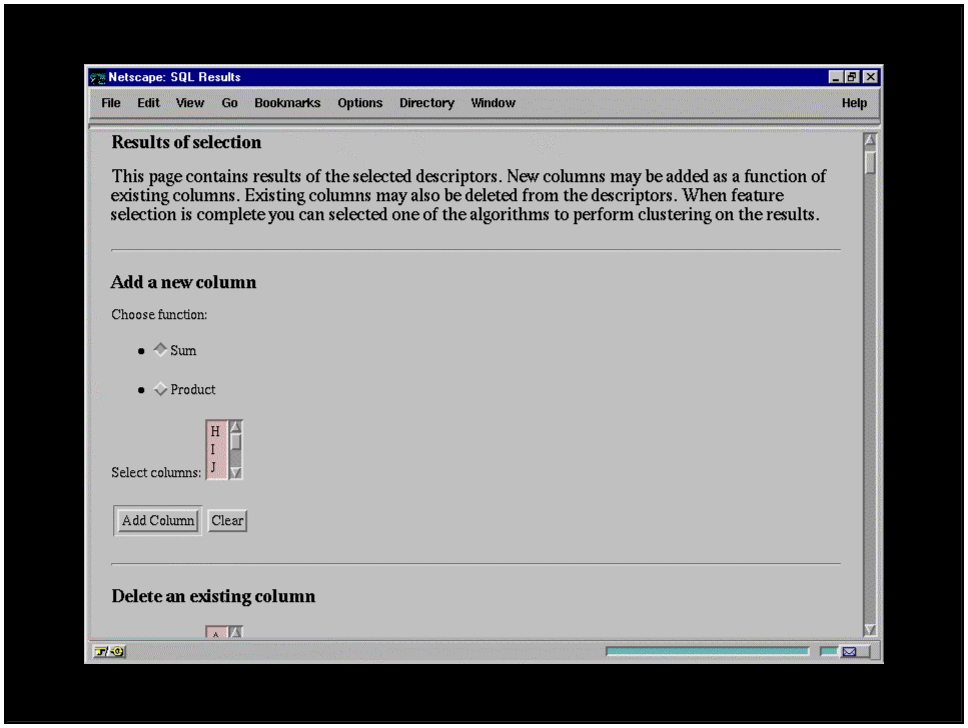

Netscape: DataFoundry - Amino Acid Descriptor Clustering
File Edit View Go Bookmarks Options Directory Window Help
Descriptor Clustering
This form can be used to:
• view amino acid descriptors in selected pdb chains
• perform computations on descriptor columns
• create new clusters of amino acid descriptors
• view all chain names in the descriptor database
• view canonical sequences for the existing descriptor clusters
Enter id's for pdb chains you wish to view.
When your results appear you can:
• Compute/delete columns for the descriptor vectors.
• Perform clustering on selected descriptors.

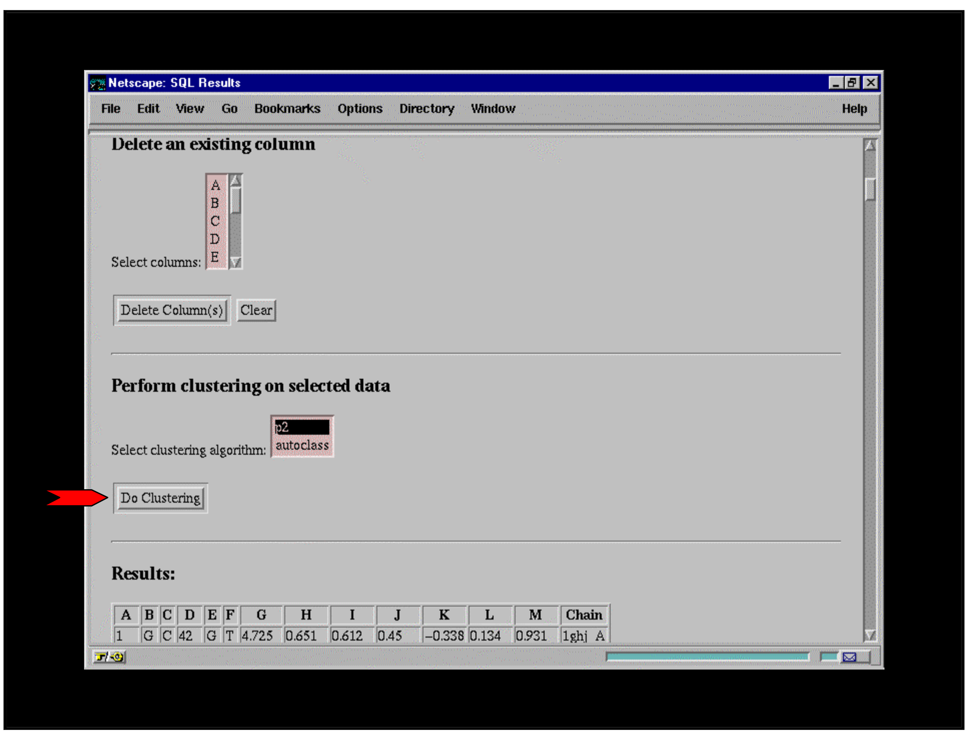

Netscape: SQL Results
File Edit View Go Bookmarks Options Directory Window Help
DataFoundry
name
1ghj_A
1ghk_A
1ghl_A
1ghl_B
1ppt_1
3dfr_1
4dfr_A
4pti_1
5pti_1
OVERVIEW
TEAM
C.S. RESEARCH
SUPERCOMPUTING
ALTERNATIVE USES
PUBLICATIONS
GENE FOUNDRY
RELATED TOPICS
These pages were last modified on Aug 26, 1998
For more information about the content or construction of these pages,

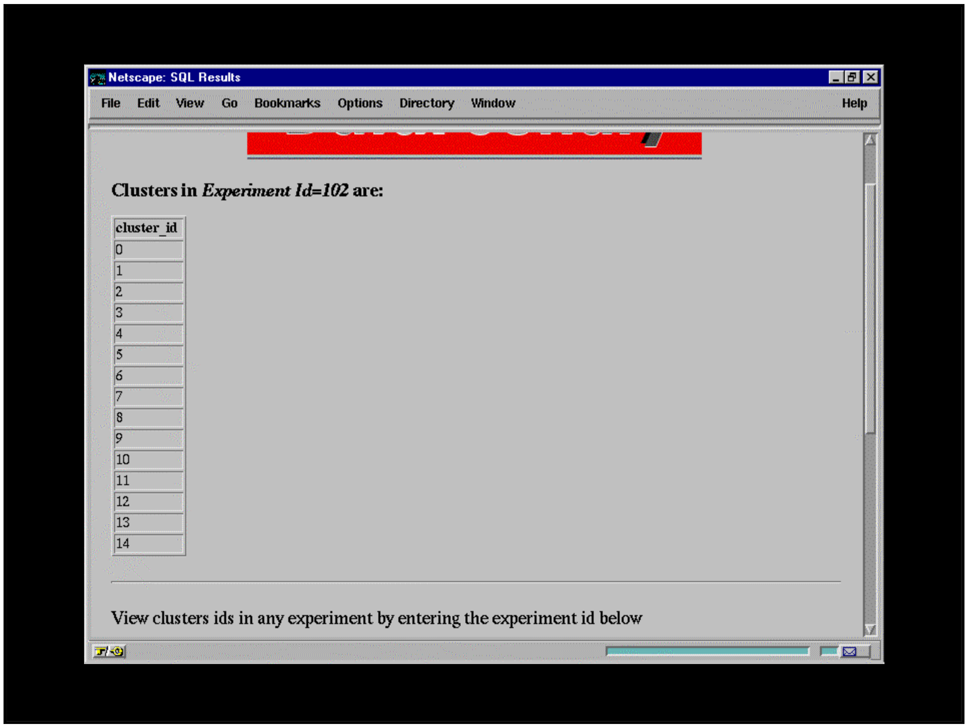


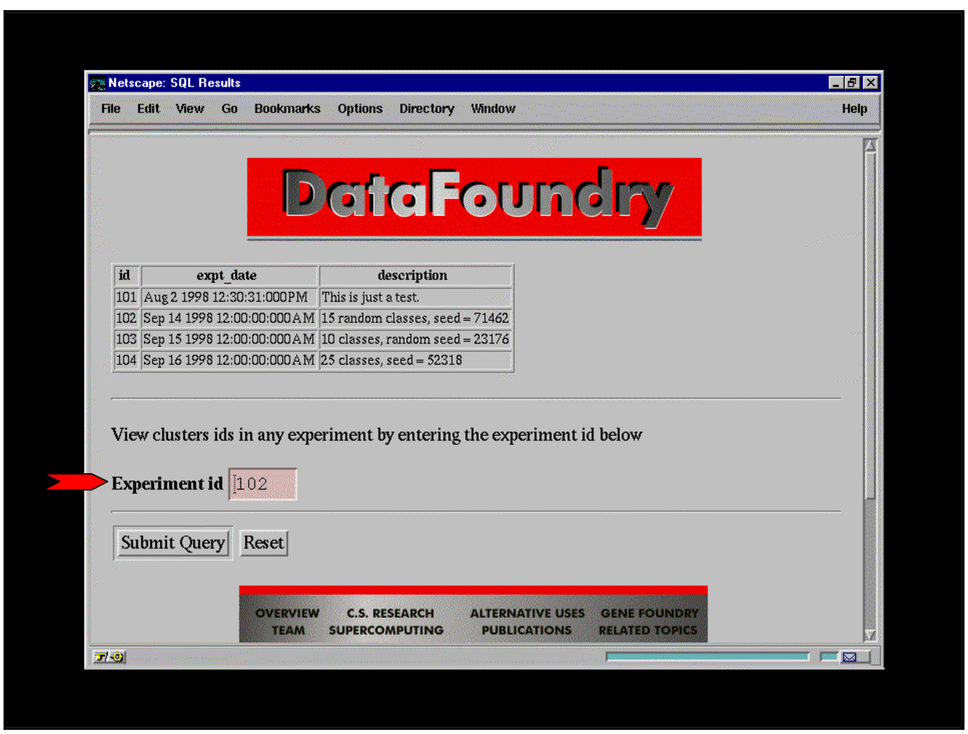



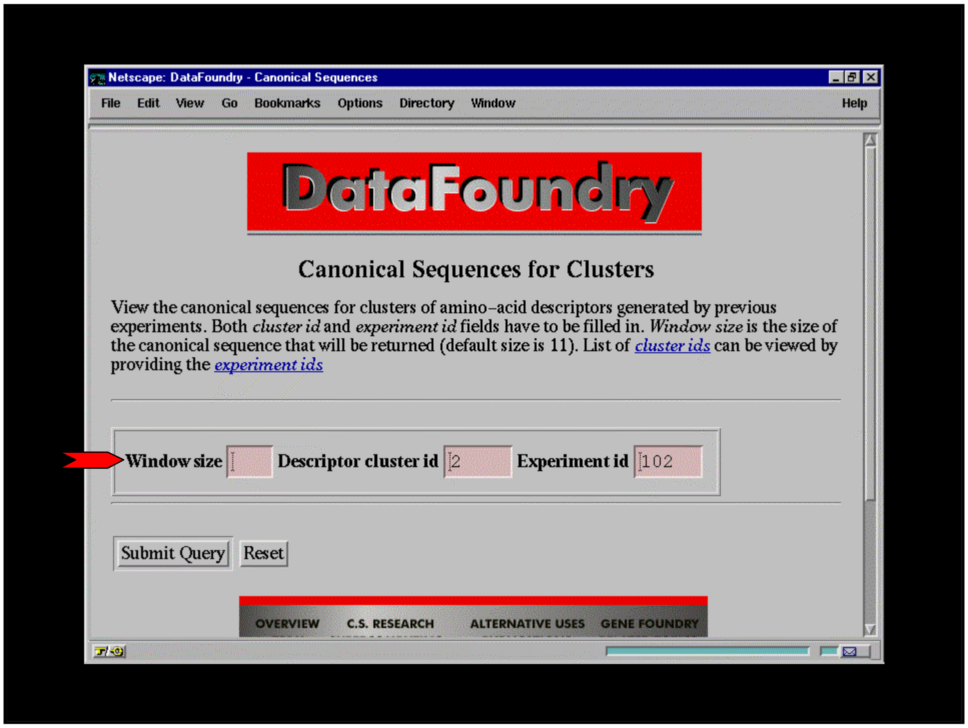

| A | B | C | D | E | F | G | H | I | J | K | L | M | Chain |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | G | C | 42 | G | T | 4.725 | 0.651 | 0.612 | 0.45 | –0.338 | 0.134 | 0.931 | 1ghj_A |
| 1 | G | C | 55 | G | T | 9.171 | 0.316 | 0.889 | 0.333 | 0.521 | 0.111 | –0.846 | 1ghj_A |
| 1 | G | C | 86 | S | S | 5.887 | –0.344 | 0.789 | –0.51 | 0.343 | –0.235 | –0.909 | 1ghj_A |
| 2 | K | C | 2 | K | C | 3.01 | –0.168 | –0.238 | –0.957 | 0.936 | –0.155 | 0.317 | 1ghj_A |
| 2 | K | C | 40 | N | B | 8.034 | 0.264 | –0.534 | –0.803 | –0.952 | 0.305 | –0.029 | 1ghj_A |
| 2 | K | C | 87 | S | S | 7.501 | –0.706 | –0.681 | –0.194 | 0.305 | –0.447 | 0.841 | 1ghj_A |
| 3 | V | B | 3 | V | B | 4.147 | –0.02 | –0.149 | –0.989 | 0.962 | –0.091 | 0.259 | 1ghj_A |
| 3 | V | B | 40 | N | B | 5.908 | –0.098 | 0.789 | –0.606 | –0.884 | –0.121 | –0.452 | 1ghj_A |
| 4 | Y | C | 4 | Y | C | 3.309 | –0.02 | –0.182 | –0.983 | 0.94 | –0.246 | 0.237 | 1ghj_A |
| 4 | Y | C | 32 | A | H | 10.135 | 0.088 | –0.904 | 0.418 | –0.61 | –0.346 | –0.713 | 1ghj_A |
| 4 | Y | C | 40 | N | B | 6.868 | –0.789 | –0.57 | –0.227 | –0.819 | 0.573 | 0.028 | 1ghj_A |
| 4 | Y | C | 56 | I | T | 10.981 | –0.773 | –0.519 | 0.366 | 0.761 | –0.366 | 0.535 | 1ghj_A |
| 4 | Y | C | 88 | D | S | 8.631 | –0.767 | 0.476 | 0.431 | 0.461 | 0.065 | 0.885 | 1ghj_A |
| 5 | G | C | 6 | R | H | 3.284 | 0.475 | –0.51 | –0.717 | 0.676 | –0.581 | 0.452 | 1ghj_A |
| 5 | G | C | 39 | F | T | 11.633 | –0.203 | 0.055 | –0.978 | –0.85 | –0.392 | 0.351 | 1ghj_A |
| 5 | G | C | 127 | G | T | 10.778 | 0.953 | –0.256 | 0.161 | –0.256 | –0.511 | 0.82 | 1ghj_A |
| 6 | R | H | 6 | R | H | 4.673 | –0.144 | –0.28 | –0.949 | 0.913 | –0.386 | 0.134 | 1ghj_A |
| 6 | R | H | 32 | A | H | 9.335 | 0.189 | 0.951 | –0.243 | –0.937 | 0.34 | 0.078 | 1ghj_A |
| 6 | R | H | 38 | N | T | 11.122 | –0.451 | 0.773 | –0.446 | –0.333 | –0.701 | –0.63 | 1ghj_A |
| 6 | R | H | 125 | I | G | 6.635 | 0.892 | –0.088 | 0.444 | –0.141 | –0.943 | –0.303 | 1ghj_A |
| 7 | C | H | 7 | C | H | 4.654 | –0.091 | –0.394 | –0.914 | 0.707 | 0.684 | –0.177 | 1ghj_A |

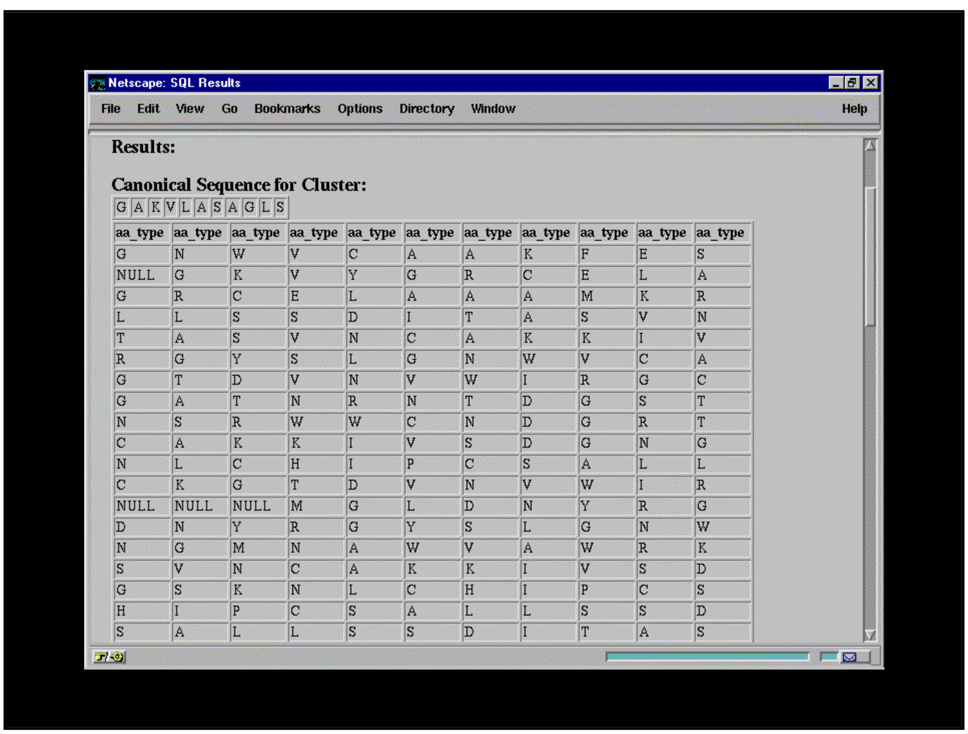

Netscape: SQL Results
File Edit View Go Bookmarks Options Directory Window Help
Results of selection
This page contains results of the selected descriptors. New columns may be added as a function of existing columns. Existing columns may also be deleted from the descriptors. When feature selection is complete you can selected one of the algorithms to perform clustering on the results.
Add a new column
Choose function:
Sum
Product
Select columns:
H
I
J
Add Column
Clear
Delete an existing column


Netscape: SQL Results
File Edit View Go Bookmarks Options Directory Window Help
Delete an existing column
Select columns:
A
B
C
D
E
Delete Column(s)
Clear
Perform clustering on selected data
Select clustering algorithm:
b2
autoclass
Do Clustering
Results:
A B C D E F G H I J K L M Chain
1 G C 42 G T 4.725 0.651 0.612 0.45 -0.338 0.134 0.931 1ghj A


Netscape: SQL Results
File Edit View Go Bookmarks Options Directory Window Help
Clusters in *Experiment Id=102* are:
cluster_id
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
View clusters ids in any experiment by entering the experiment id below


Netscape: SQL Results
File Edit View Go Bookmarks Options Directory Window Help
DataFoundry
id | expt_date | description
101 | Aug 2 1998 12:30:31:000PM | This is just a test.
102 | Sep 14 1998 12:00:00:000AM | 15 random classes, seed = 71462
103 | Sep 15 1998 12:00:00:000AM | 10 classes, random seed = 23176
104 | Sep 16 1998 12:00:00:000AM | 25 classes, seed = 52318
View clusters ids in any experiment by entering the experiment id below
Experiment id 102
Submit Query
Reset
OVERVIEW
TEAM
C.S. RESEARCH
SUPERCOMPUTING
ALTERNATIVE USES
PUBLICATIONS
GENE FOUNDRY
RELATED TOPICS




| aa_type | aa_type | aa_type | aa_type | aa_type | aa_type | aa_type | aa_type | aa_type | aa_type | aa_type |
|---|---|---|---|---|---|---|---|---|---|---|
| G | N | W | V | C | A | A | K | F | E | S |
| NULL | G | K | V | Y | G | R | C | E | L | A |
| G | R | C | E | L | A | A | A | M | K | R |
| L | L | S | S | D | I | T | A | S | V | N |
| T | A | S | V | N | C | A | K | K | I | V |
| R | G | Y | S | L | G | N | W | V | C | A |
| G | T | D | V | N | V | W | I | R | G | C |
| G | A | T | N | R | N | T | D | G | S | T |
| N | S | R | W | W | C | N | D | G | R | T |
| C | A | K | K | I | V | S | D | G | N | G |
| N | L | C | H | I | P | C | S | A | L | L |
| C | K | G | T | D | V | N | V | W | I | R |
| NULL | NULL | NULL | M | G | L | D | N | Y | R | G |
| D | N | Y | R | G | Y | S | L | G | N | W |
| N | G | M | N | A | W | V | A | W | R | K |
| S | V | N | C | A | K | K | I | V | S | D |
| G | S | K | N | L | C | H | I | P | C | S |
| H | I | P | C | S | A | L | L | S | S | D |
| S | A | L | L | S | S | D | I | T | A | S |

# The DataFoundry Architecture

**Ron Musick**

*Terence Critchlow* *Madhavan Ganesh*

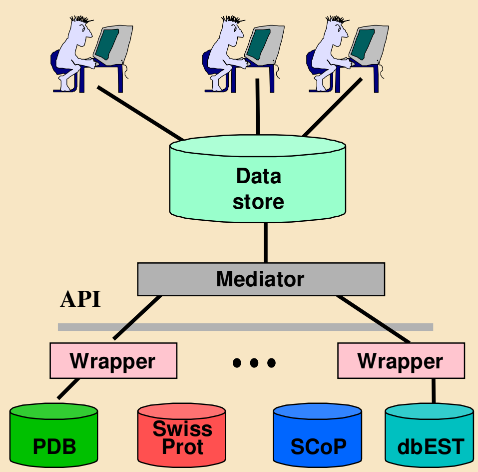
Center for Applied Scientific Computing

*Krzysztof Fidelis* *Tom Slezak*

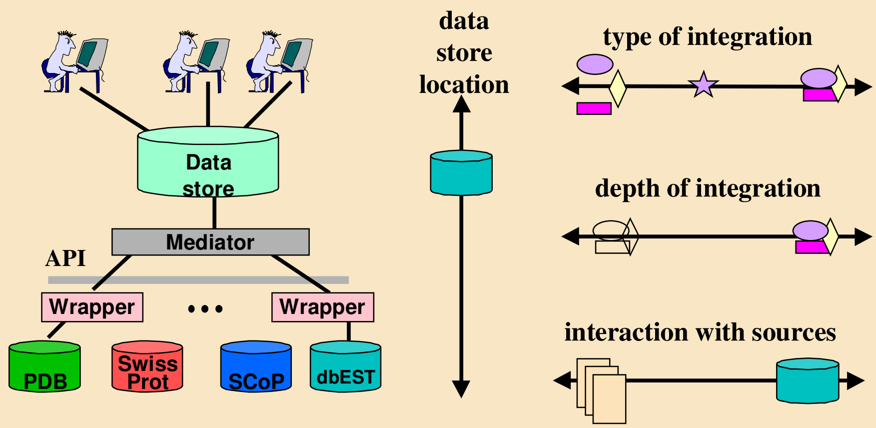
Biology and Biotechnology Research Program

Lawrence Livermore National Laboratory

## Our mediated architecture: design decisions, directions, and benefits



## Details of the design drive the capabilities of the system




## A good design is essential, since the decisions impact every aspect of the warehouse

- Where data is kept, how it is accessed will impact
  - Performance
  - System scalability to # sources, size of data, # users
  - Expected mode of interaction with external sources
  - System adequacy to support interactive browse and analysis-oriented activities
- **The degree of schema and data integration impacts**
  - **How hard the user works to use the system**
  - **How hard the developers work to create and maintain it**
  - **The types of tasks the user can perform**
  - **Flexibility and scalability**

## We have focused on making data warehousing practical in dynamic, scientific settings

**Primary criteria:**

- **Improve the scientists' interaction with data**
  - Fast access to data from several sources
  - Common interface and vocabulary
  - Specific user features: alerts, updates, cleaning, ...
  - Local data analysis tools + data mining
- **Make the warehouse cheaper to build and maintain**
  - Able to rapidly add new sources
  - Easy to react when a source changes

**These place challenging constraints on the underlying system architecture.**


## We have focused on making data warehousing practical in dynamic, scientific settings

**Primary criteria:**


- **Improve the scientists' interaction with data**
  - Data cached locally for performance
  - Global schema to minimize complexity to user
  - Private user spaces, support for data curation
  - Support day to day scientific activity, detailed data access


- **Make the warehouse cheaper to build and maintain**
  - Able to rapidly add new sources
  - Easy to react when a source changes

**These place challenging constraints on the underlying system architecture.**

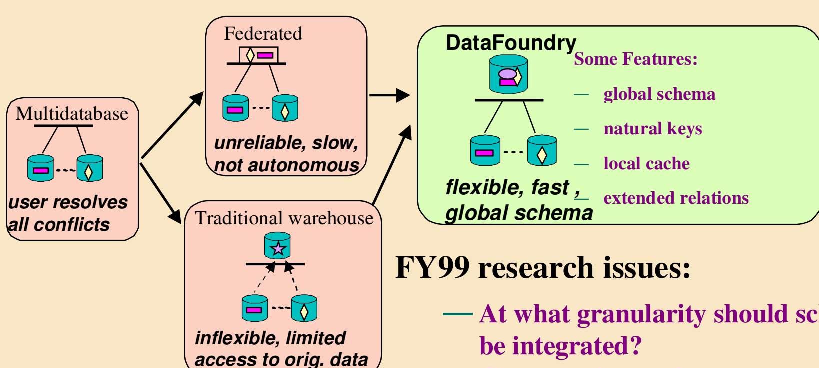
## No system we know of can satisfy these constraints today

- **Improve the scientists' interaction with data**
  - Data cached locally for performance
  - Global schema to minimize complexity to user
  - Private user spaces, support for data curation
  - Support day to day scientific activity, detailed data access
- **Make the warehouse cheaper to build and maintain**
  - Automated data integration and ingest
  - Schema evolution management
  - Flexible architecture

**And no standard warehousing architecture has been designed to meet these needs**

## We are exploring a new warehouse architecture to provide these capabilities

**Target: provide rich, easy, fast access to data, while reducing the work to integrate a new source**

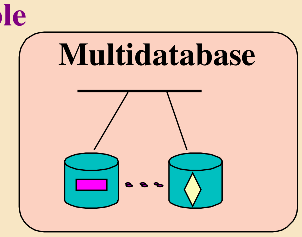



**FY99 research issues:**

- At what granularity should schema be integrated?
- Characterize performance, flexibility, cost tradeoffs

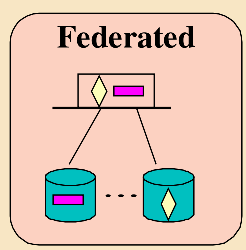
## Multidatabases provide a lightweight shell over heterogeneous DBMSs

- **Pros**
  - Easy to build
  - Flexible
  - Cheap to maintain
  - Several multidatabase tools available
- **Cons**
  - Impractical for the user
    - No common vocabulary
    - Performance issues
  - Restricts data sources



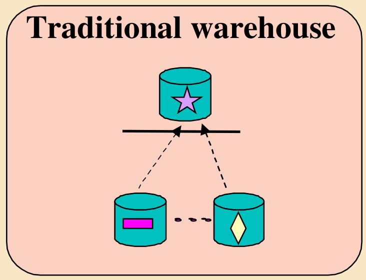

## Federated systems provide an integrated view of data over a federation of sources

- **Pros**
  - Common vocabulary
  - Common interface
- **Cons**
  - Performance issues
  - Reliability issues
  - Requires participating sources



## Traditional data warehouses are varied, but typically serve off-line work

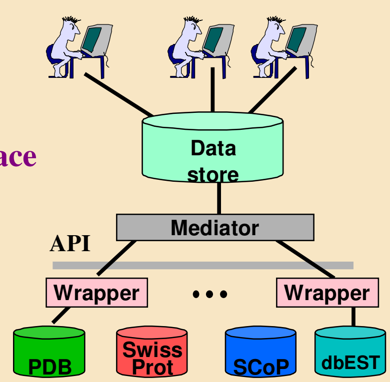
- A "traditional" warehouse
  - ad-hoc designs; bottom up, or top down collections of data marts
  - offline decision support
- Pros
  - local data for performance
  - Consistent vocabulary
  - Better support from industry
- Cons
  - inflexible
  - limited access to original data




CASC

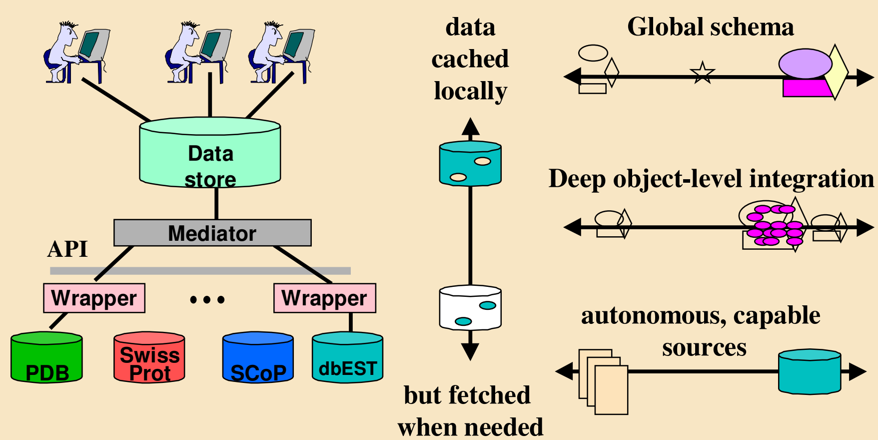
## DataFoundry is designed to support day to day scientific activity

- Pros
  - local data for performance
  - Consistent vocabulary, interface
  - full access to original data
  - flexible, scalable
- Cons
  - still unproven




CASC

## Our architecture combines elements of the three previous approaches

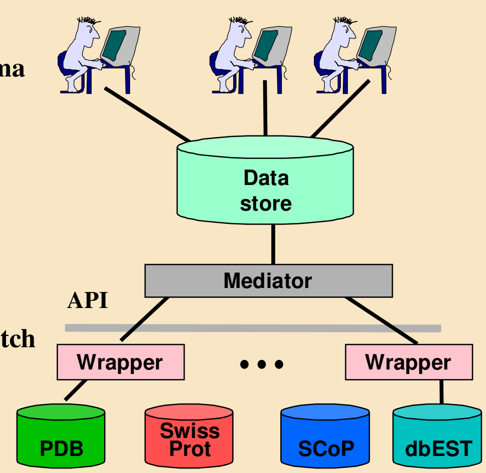



## Current state of DataFoundry prototype

- Users see a single global schema
- Deeply integrated data and schema

- All source data is stored locally
- Source data is updated weekly

- Sources are autonomous
- Sources are interacted with in batch mode only, currently



## In summary

- Unique architecture
- Supports day to day scientific activity
- Designed to be easier to build, cheaper to maintain

**We enhance user capabilities, and reduce costs**

## The Joint Genome Institutes Perspective on Genome Data Warehousing

Tom Slezak

Informatics Team Leader, JGI
slezak@llnl.gov

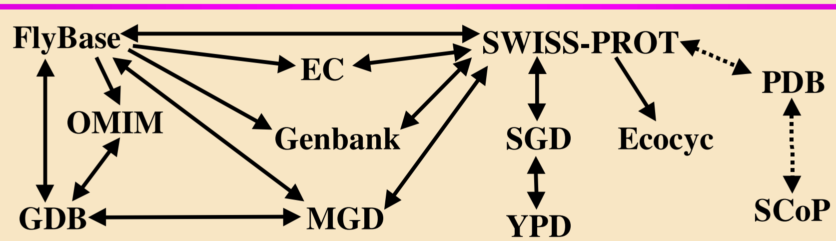
## The tangled Web of genome links

**Genome databases tend to consist of highly-specialized and focused information, with annotation links to other similar specialized databases. Obvious problems include:**

- **Inability to easily query across multiple databases.**
- **Asynchronous and unreliable updating of links.**
- **Uni-directional links.**
- **Specialized sub-discipline terminology.**
- **Requiring DB structural knowledge for querying.**

**(Reference: William Gelbart's "Databases in Genomic Research", Science, Vol. 282, 23 October, 1998)**

## Genome DB linkage example




**Consider an example of the difficulties establishing and maintaining links between genomic databases.**

**•Gelbart's PKG1 gene example did not find PDB or SCoP because protein structure is not yet known for this gene.**

**•Unidirectional links make some starting points quite problematic: consider starting in Genbank and trying to get data from flybase or starting in OMIM and trying to find a Fly disease phenotype.**

(Reference: William Gelbart's "Databases in Genomic Research", Science, Vol. 282, 23 October, 1998)

## JGI Goals for Data Interconnectivity

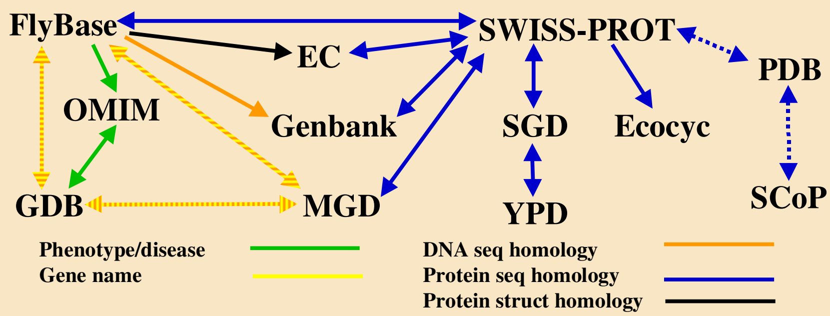
- Define an extensible set of "natural keys" between relevant genomic databases (gene names, protein sequences, disease names, phenotypes, authors, etc).
- Construct a JGI-specific schema that normalizes external data to this view of natural keys.
- Automate the extraction and maintenance of all linkages between these natural keys.

## JGI Goals for Data Interconnectivity (cont)

- Provide a suite of interfaces that give democratic access to data (ie starting with any source, you can discover data from all the others).
- Provide a suite of data discovery and data mining tools that can effectively navigate and exploit this wealth of connected data.

Note that the JGI has a rather limited scope and focus compared to NIH, drug manufacturers, or clinicians.

## DB linkage motivates natural keys




**Although many of the database linkages are done via accession numbers, shared concepts such as gene names, disease/phenotype names, marker names, or DNA/protein sequence similarity are what actually establish the connections.**

## JGI and DataFoundry

- DOE funding is focused rather brutally on DNA sequencing for the near-term.
- Significant lead time is needed to prepare adequate data infrastructure.
- JGI wants a partner who can help it build scalable integration tools for the new millenium.

**Although the JGI focus differs from that of industry, the DataFoundry design makes it feasible for partnerships to produce a system that could readily be specialized for the diverse needs of both the JGI and the industrial partner.**

# DataFoundry and You

**Terence Critchlow**

*Ron Musick* *Madhavan Ganesh*

**Center for Applied Scientific Computing**

*Krzysztof Fidelis* *Tom Slezak*

**Biology and Biotechnology Research Program**

**Lawrence Livermore National Laboratory**

## Tying it all together

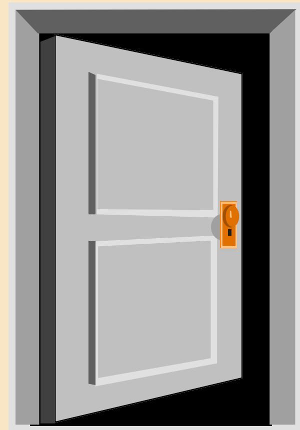
**The capability of bioinformatics tools to access data from several different gene and protein databases will be the top industry challenge in 5 years.**

Frost & Sullivan: US Bioinformatics in the Emerging Drug Discovery Marketplace 1998

**Meeting this goal will require an intensive research and development effort.**

## DataFoundry has: access

- Computer resources
  - high-performance platforms available if required
- Joint Genome Institute scientists
  - partially funding DataFoundry
  - used to working in multi-disciplinary environment
- Academic institutions
  - able to attract students (LSI goal)
  - on-going collaborations with research leaders



CASC

## DataFoundry has: a meta-data infrastructure

| Activity/ integration style | manual | meta-data | diff | %diff |
|---|---|---|---|---|
| understanding SCOP | 2.0 | 2.0 | 0.0 | 0 |
| writing wrapper | 4.5 | 2.5 | 2.0 | 44% |
| modifying schema | 0.5 | 0.5 | 0.0 | 0 |
| writing mediator | 4.0 | 0.0 | 4.0 | --- |
| modifying meta-data | 0.0 | 1.0 | (1.0) | --- |
| total time in days | 11.0 | 6.0 | 5.0 | 45% |

CASC

## DataFoundry has: a proven team

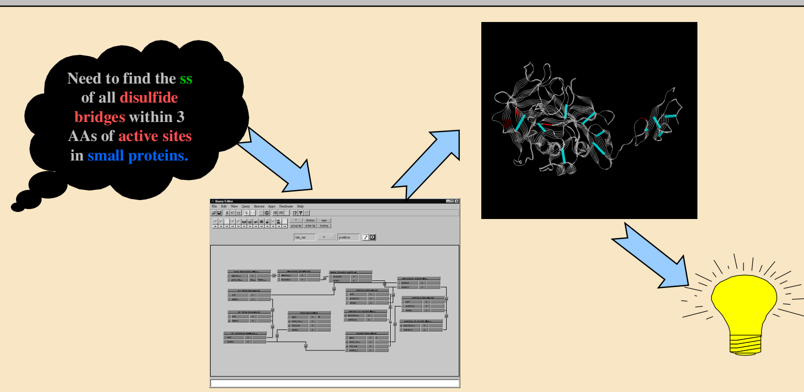
- CS Researchers
  - Ron Musick
  - Terence Critchlow
  - Madhavan Ganesh
- Domain experts
  - Krzysztof Fidelis & Adam Zemla
  - Tom Slezak
- Academic collaborators
  - Prof. Nabil Adam & Igg Adiwijaya (Rutgers)
  - Todd Taylor (UC Davis)
  - Rachel Karchin (Stanford)

**Been working together for 2 years.**

CASC

## DataFoundry has: a working prototype

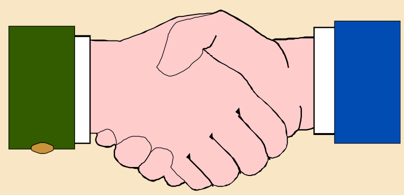
**Semantically integrated data from several sources.**




CASC

## DataFoundry wants: a strong collaboration

**Work with us to transform our leading-edge research into your market-dominating technology.**



## What's in it for you?

**Guide a successful, ongoing bioinformatics research project towards the challenges facing your company, and license the resulting technology, without having to hire the entire team.**

## So, what's next?

- Identify areas of potential collaboration
  - where does DataFoundry fit in with your goals
  - and vice versa
- Meet with key decision makers in your company
  - at LLNL? or at company headquarters?
  - dive into the details of collaboration
- Decide on how to best organize collaboration
  - LSI proposal
  - CRADA

Work performed under the auspices of the U.S. DOE by LLNL under contract No. W-7405-ENG-48