Cyberinfrastructure & Bioinformatics

Transformation of The Biology Workbench

R. H. Niedner

Bioinformatics

- Biological Data are
  - distributed: 500 - 1000 databases
  - heterogeneous: type, format, source, methods
  - massive: up to 100K of individual data points, or 100MB blobs (image files)

Bioinformatics

- Tools:
  - hundreds of tools for sequence, alignment and structure analyses
  - different input / output formats
  - hardware and software requirements
  - which tool for which job
  - how do I run them and how do I run them right

To Much Information

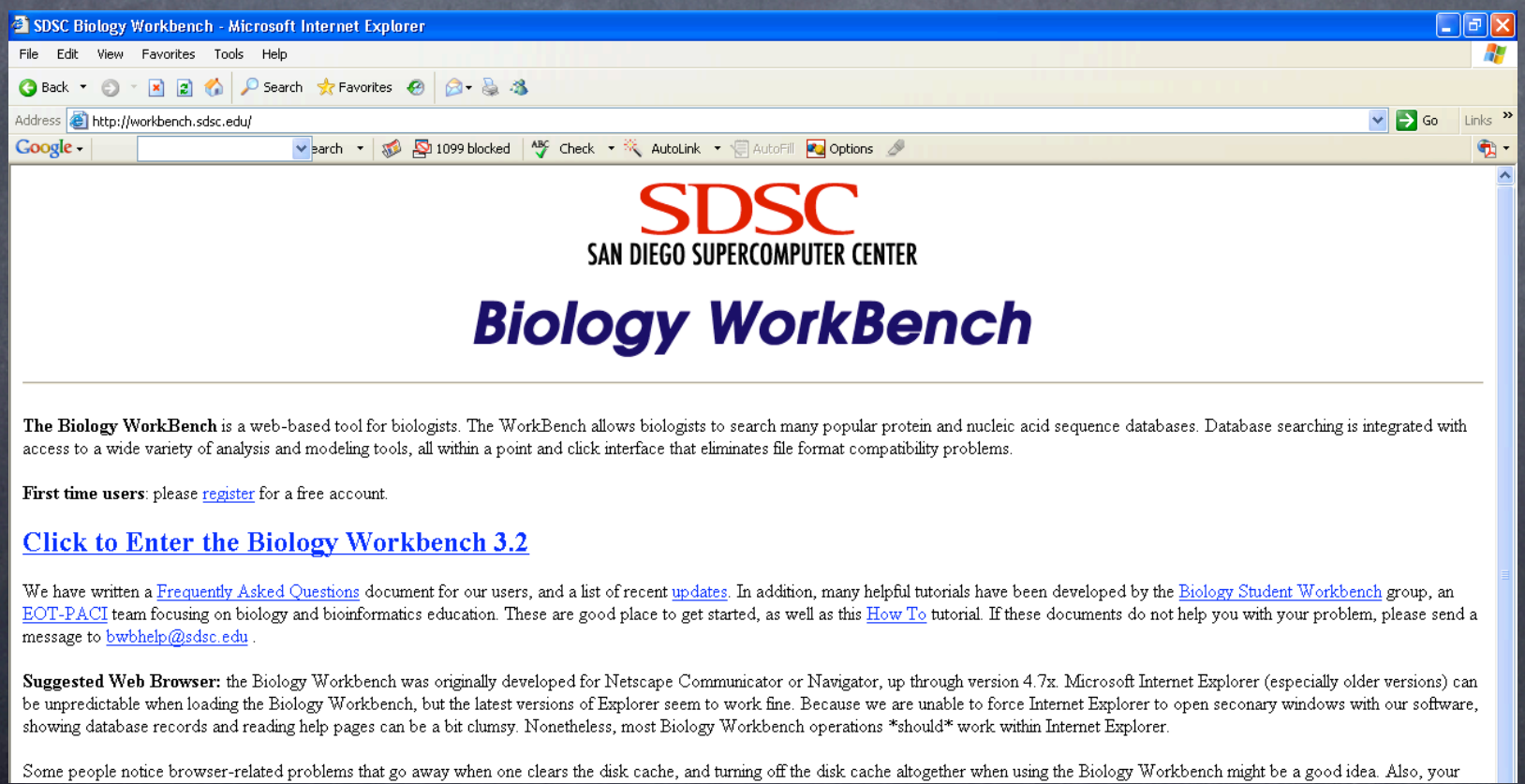
Biology Workbench

- 1996 Desktop Computers
  - 200 MHz Pentium
  - 1-2 GB HDD
  - 32 MB Ram
- The original concept behind BWB: “Wouldn't it be nice if I had a web site that would let me run BLAST, ClustalW, etc. on my collection of sequences, or a collection of sequences from many remote resources?”

The current Workbench

Created 1996-1997 at NCSA by Shankar Subramaniam, Eric Jakobsson, Roger Unwin, Brian Saunders, Mark Stupar, Dawn Cotter, Jim Fenton, Curt Jamison, Brad Mills, George Pappas, David Tcheng (at SDSC since 2000)

http://workbench.sdsc.edu/



Workbench Features

- Platform independent: only a web browser is needed (no plugins required)
- All calculations provided by the Workbench Server
- Individual login password security provided.
- Data storage area provided for results.

Features cont.

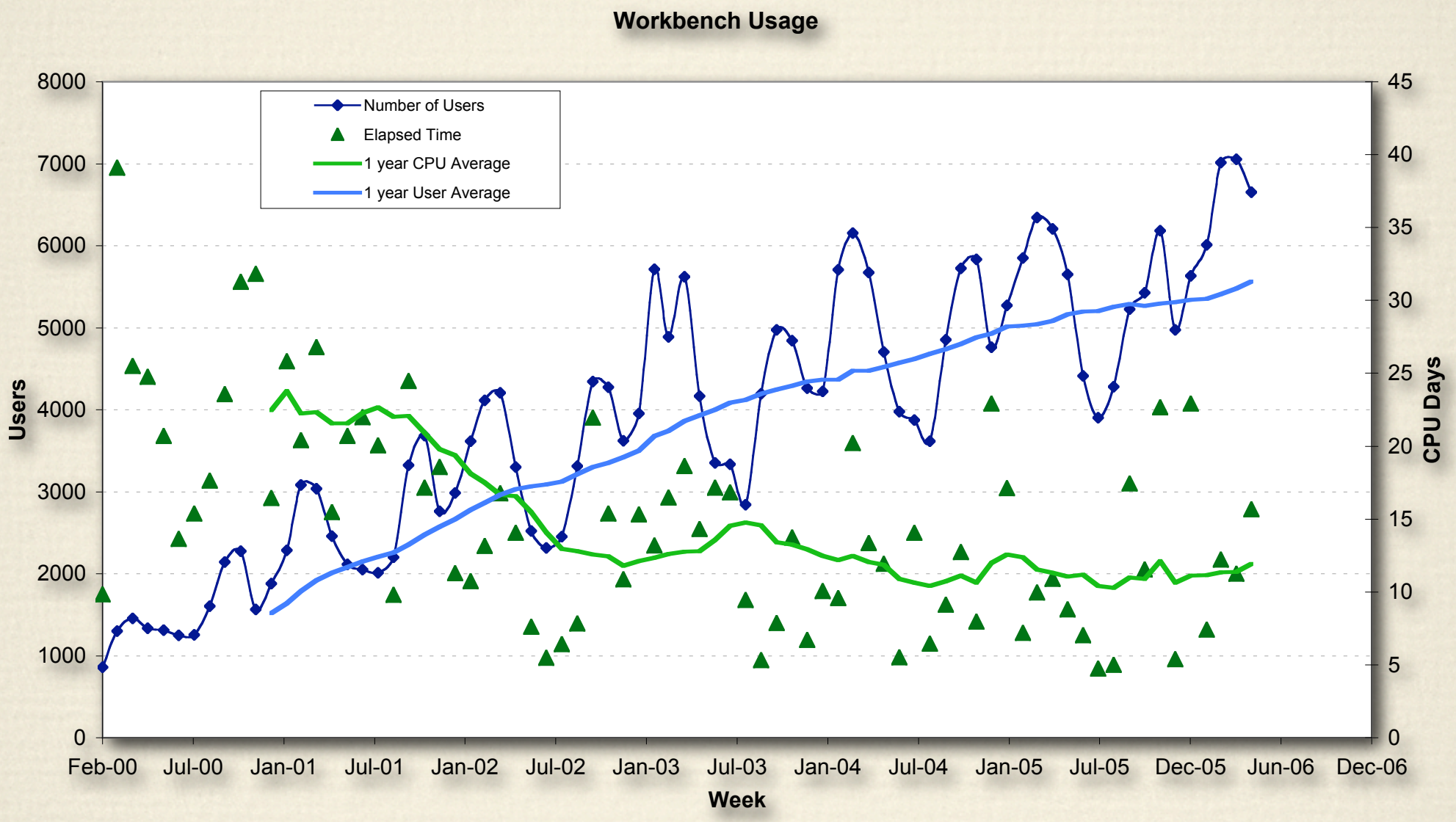
- 33 Federated protein and nucleic databases with robust search utility
- 66 of protein, nucleic, and alignment tools
- Seamless movement of sequence data between the various tools
- Can be (and is) used over phone modem.

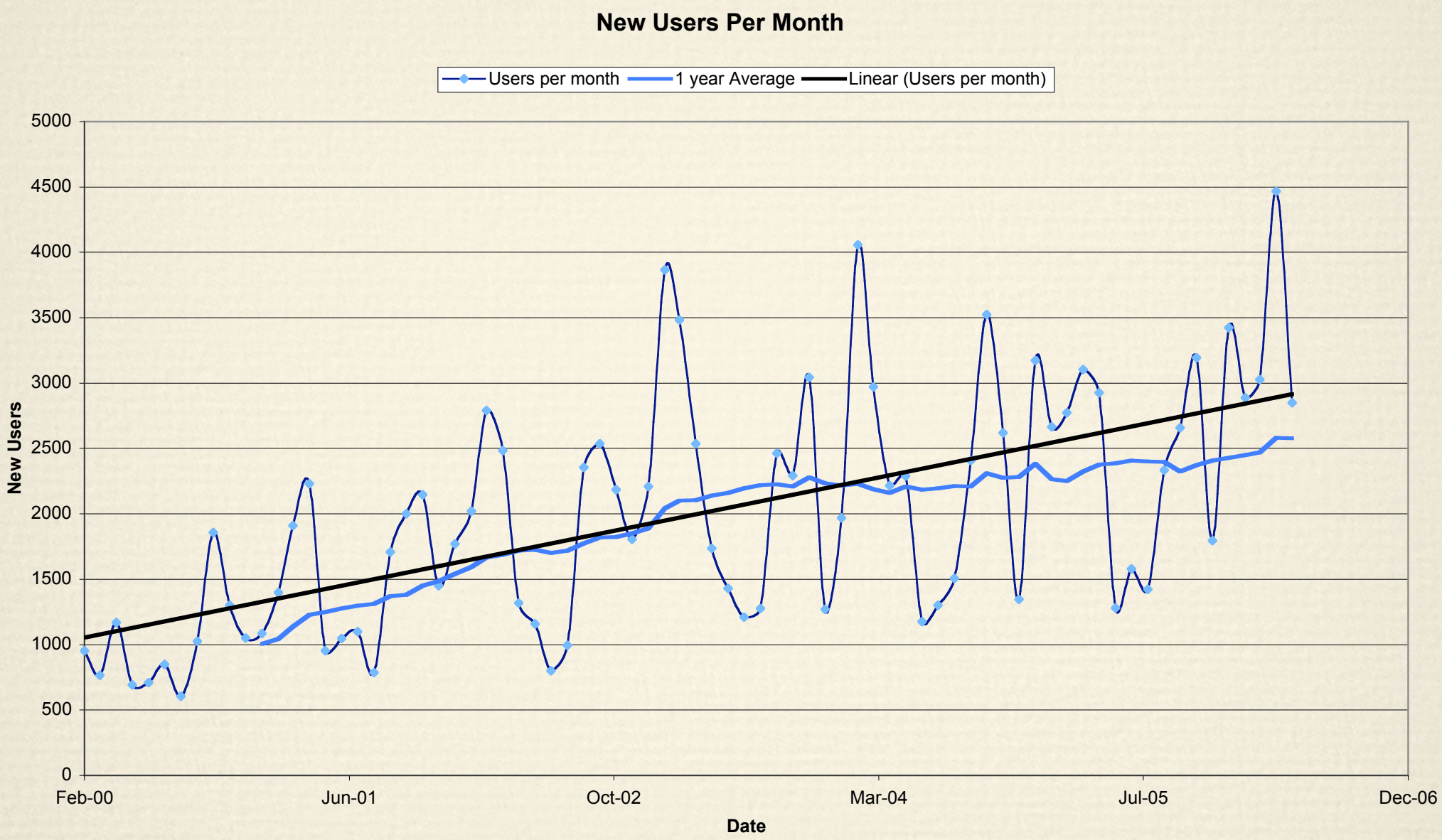
Hardware

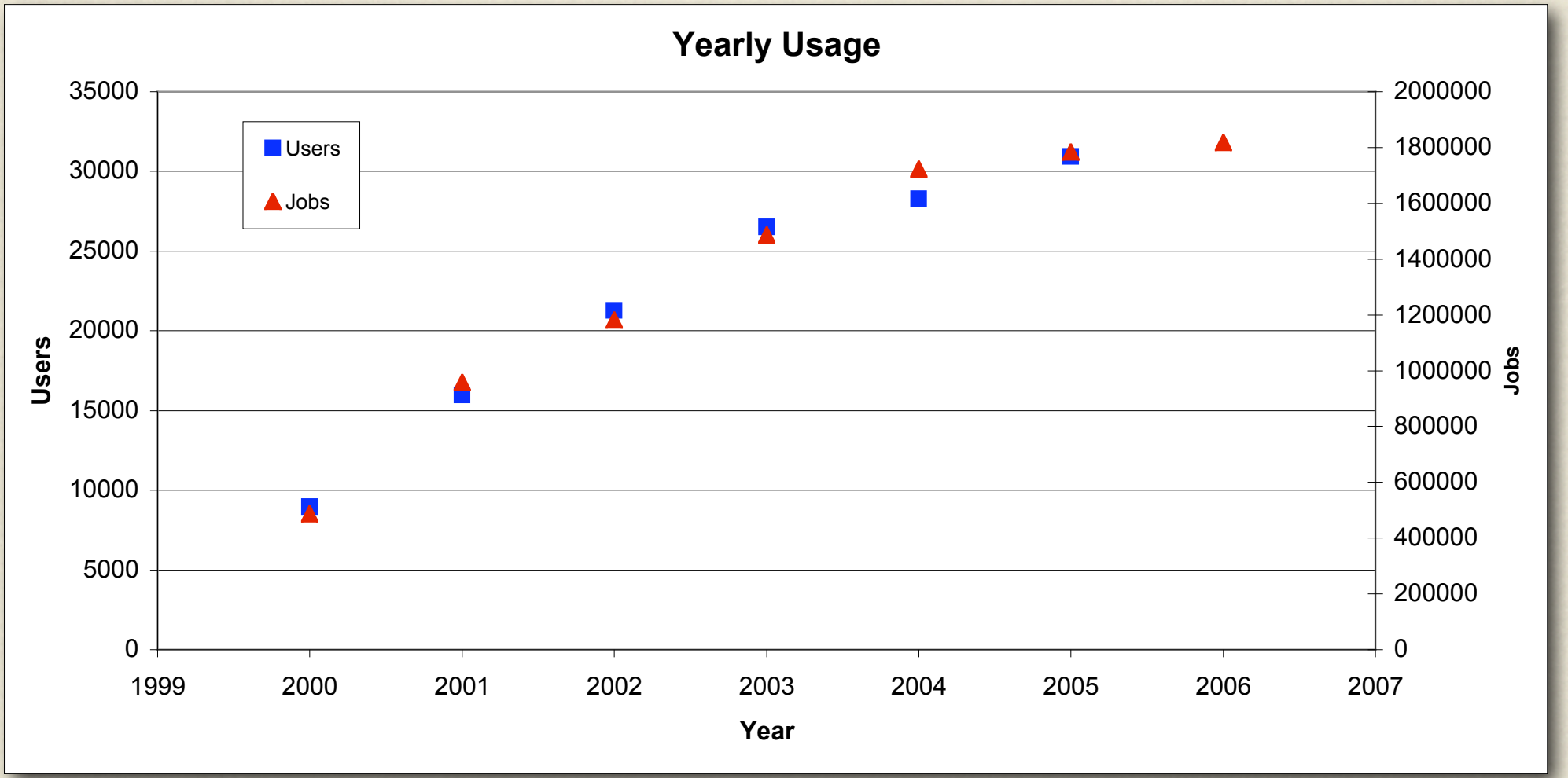
- 4x900 MHz processor Sun Fire 480R system, 8 GB memory
- 768 GB disk (670 GB with RAID)
  - 500 GB actively used
  - 400 GB used for database mirroring
  - Genbank alone uses over 300 GB
  - Need close to 50 GB temporary space for database mirroring process

Statistics

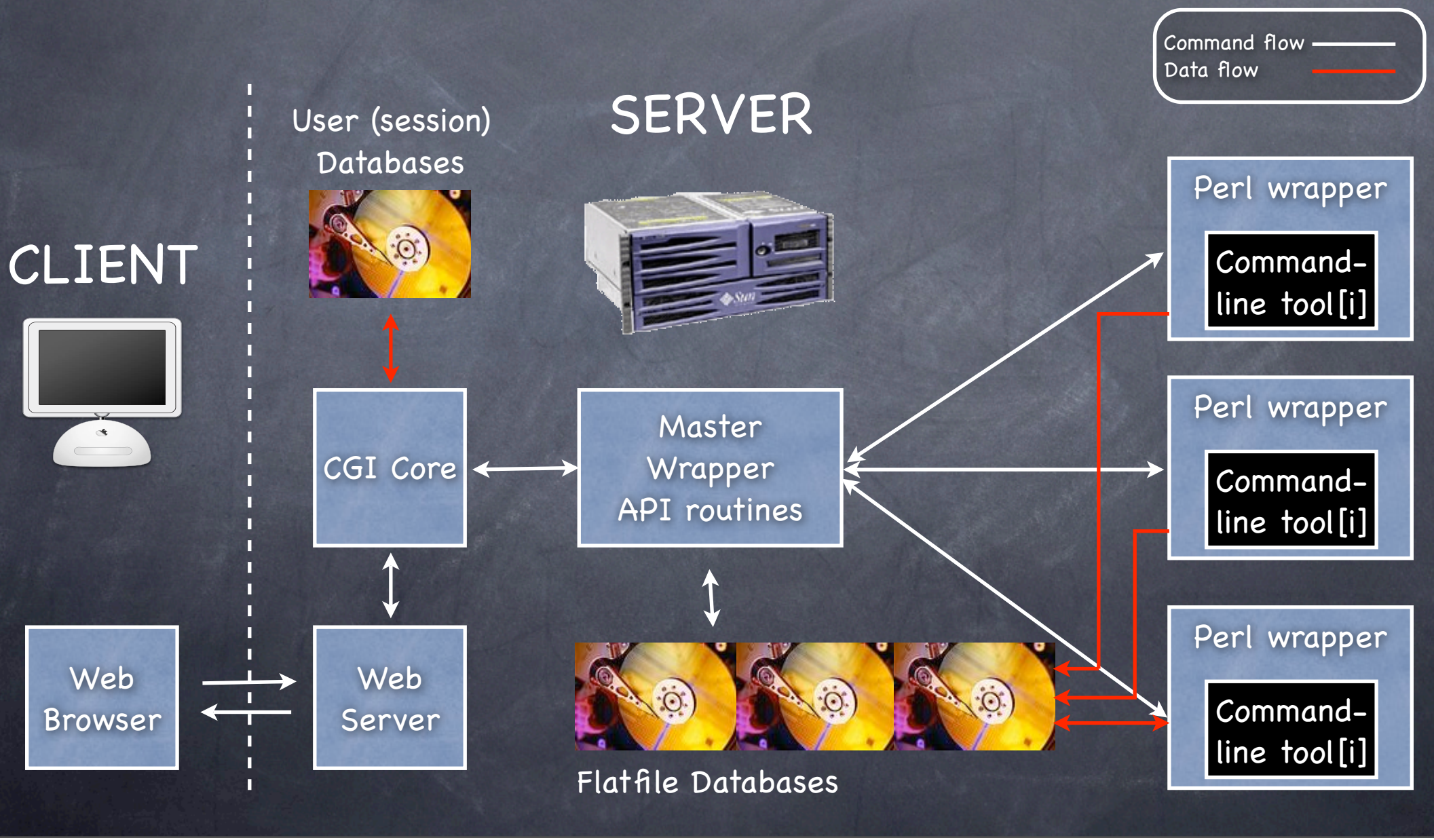
- Over 150,000 users since the Biology Workbench was moved to SDSC
- 2000 users / 200,000 hits each week
- Relatively low CPU usage (10-20% CPU utilization)
  - Most CPU usage by a small number of "power users"


Workbench Usage
Number of Users
Elapsed Time
1 year CPU Average
1 year User Average
Users
CPU Days
Week
8000
7000
6000
5000
4000
3000
2000
1000
0
45
40
35
30
25
20
15
10
5
0
Feb-00
Jul-00
Jan-01
Jul-01
Jan-02
Jul-02
Jan-03
Jul-03
Jan-04
Jul-04
Jan-05
Jul-05
Dec-05
Jun-06
Dec-06


New Users Per Month
Users per month
1 year Average
Linear (Users per month)
New Users
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
Feb-00
Jun-01
Oct-02
Mar-04
Jul-05
Dec-06
Date


Yearly Usage
Users
Jobs
35000
30000
25000
20000
15000
10000
5000
0
2000000
1800000
1600000
1400000
1200000
1000000
800000
600000
400000
200000
0
1999
2000
2001
2002
2003
2004
2005
2006
2007
Year

Command Flow



CGI Core

(C legacy code)

- Collects form and hidden variables
- Writes and Reads in session files
- Processes old and new sequences
- Performs certain "core" operations (also not good)
- Draws basic interface

Commandline Tool Wrappers (Perl)

- Core calls module wrapper script, which then "calls" API, which then calls a "main" script in the module wrapper script.
- Scripts set parameters in form the application programs require
- Parse input data into required format
- Parse output (render "prettier" form/identify sequences)

Wrappers cont.

- Sequence conversion - done by Perl application ("seqvert") [formerly done with Readseq]
- Output from application programs is parsed, to get information and to process into "prettier" form
- Importable sequences identified as such, and stored as hidden variables

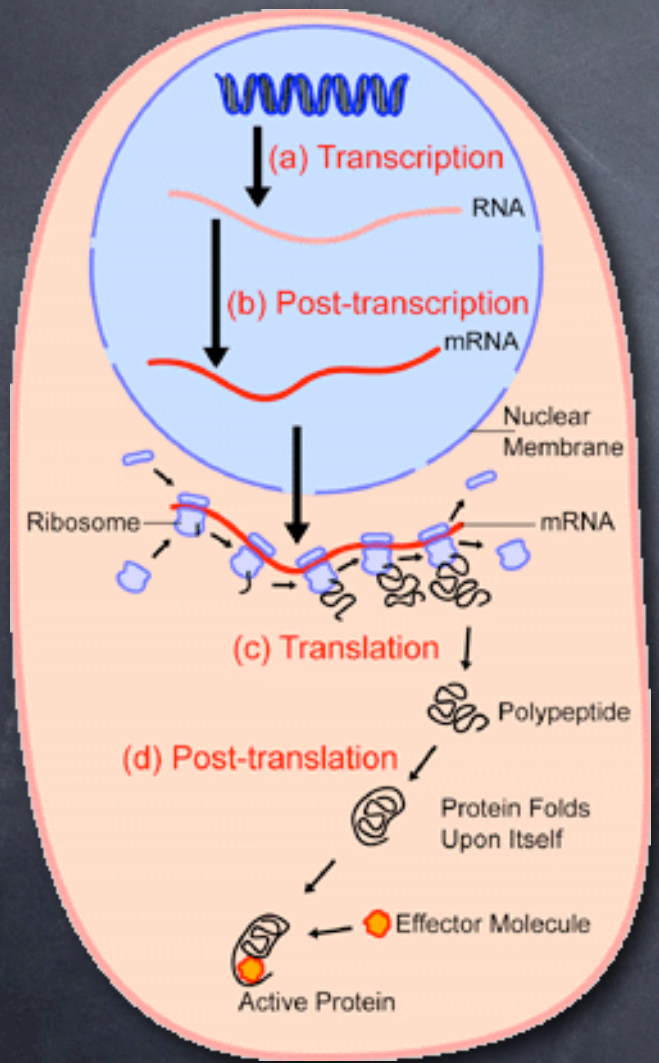
Workbench Tools

- BLAST (including PSI-BLAST)
- Fasta
- Clustal W
- Boxshade
- Assorted Phylip Tools (drawtree, drawgram, protdist, protpars, dnadist, dnapars)
- NDJINN (database searching interface)
- other global and local alignment tools
- secondary-structure prediction
- sequence statistics
- pattern-match only homology
- restriction enzyme and primer tools

Workbench Statistics

- 71% of the user base is domestic.
- 44% are academic
- 15% noncommercial
- 11% commercial
- 1% government
- The 29% international user population represents over 40 countries
- 50% of present users employ the BW for government-funded research programs

What we really want



"There should be a web site that can host all of my biological data—not just sequences —and allow me to analyze it using any modern tool I choose."

So what is wrong?

- User directories all in one directory
  - Physical limit of close to 32,000
  - Purge users on regular basis by time since last access
- Session file / core interaction
  - Constantly read and rewritten
  - Large sessions (many sequences or large sequences) cause errors
- Synchronicity errors (Often blows away session files)
- Data types (Hard to define non-sequence data types)
- Browser idiosyncrasies (embedded non-standard CGI in the core)

and more

- very difficult to add new tools/data
  - no structure, protein interaction or pathway data
  - no genome or SNP analysis
- current architecture doesn't scale
  - monolithic CGI Core, no RDBMS, no segregation of function
  - hard- and hand coded adapters for each tool
- very limited interface and functionality
  - limited search and visualization capabilities
  - only one tool at the time - no workflows

So lets use the Grid

- The promise:
  - Super powerful processing power
  - Super large memory
  - Super high-speed network
  - Super high-capacity storage space
- Transformation of remote supercomputing power into a virtual local resource

Cyberinfrastructure

- BUT where is the SUPER EASY
  - Differing Authentication, Authorization
  - Different Access and Allocation Policies
  - Multitude of Platforms and Standards
  - How do I find what I need (Data, Applications)
  - Where and When can I run my job .....

Cyberinfrastructure

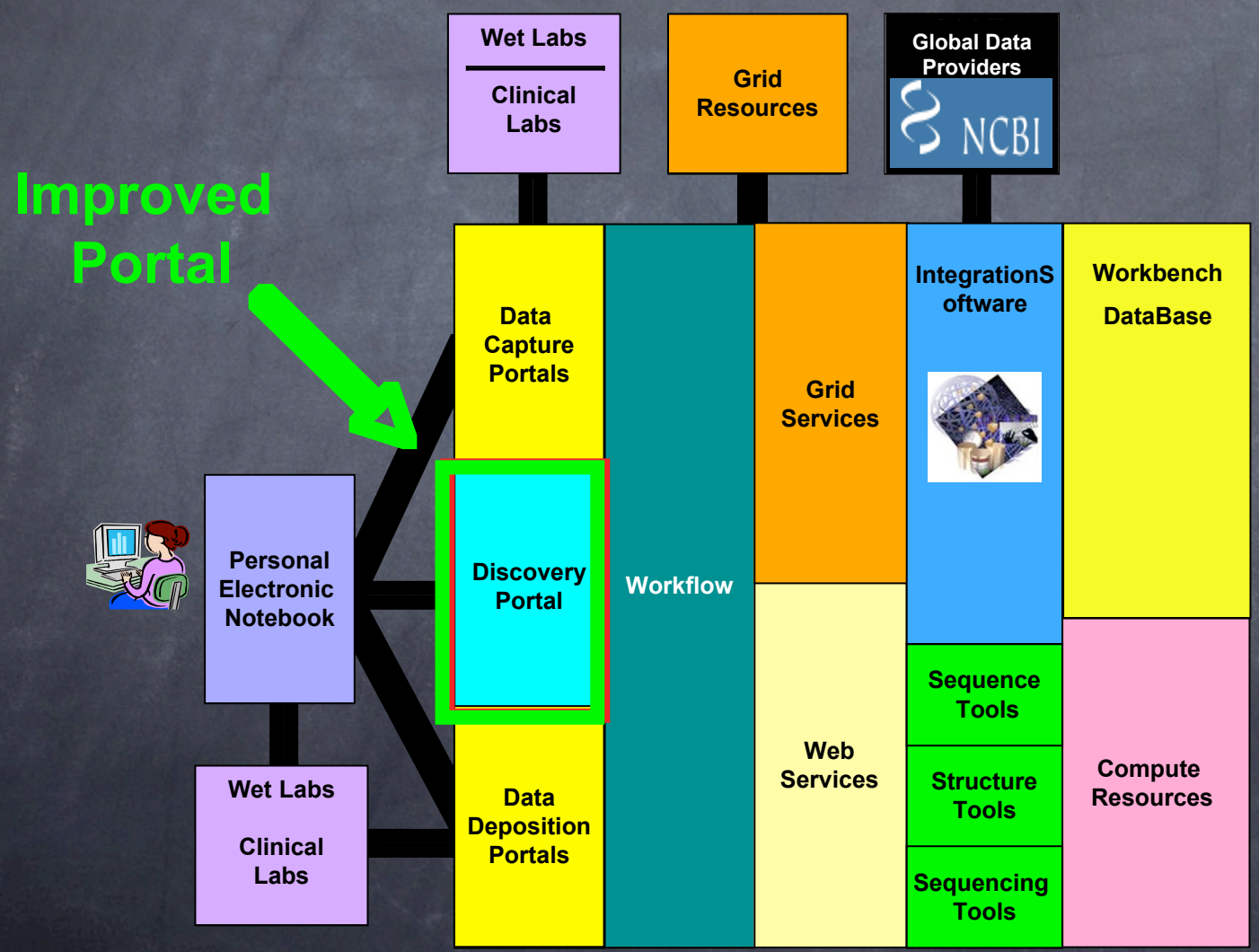
- Current Reality is that researchers still:
  - need to know to many details
  - have to have to much local support
  - deal with to many standards
- before they get to do grid-enabled science!

SDSC Mission

- To serve as a premiere resource for design, development, and deployment of cyberinfrastructure for the national scientific community.
- Harness the power of the grid and shield researchers from complexities of the implementation:

  Create Science Portals

The new Grid-enabled Workbench



Key Technologies for the new Workbench

- XML and Ontologies
- Database Federation
- Object Relational Mapping
- SOA and Webservices
- Workflows

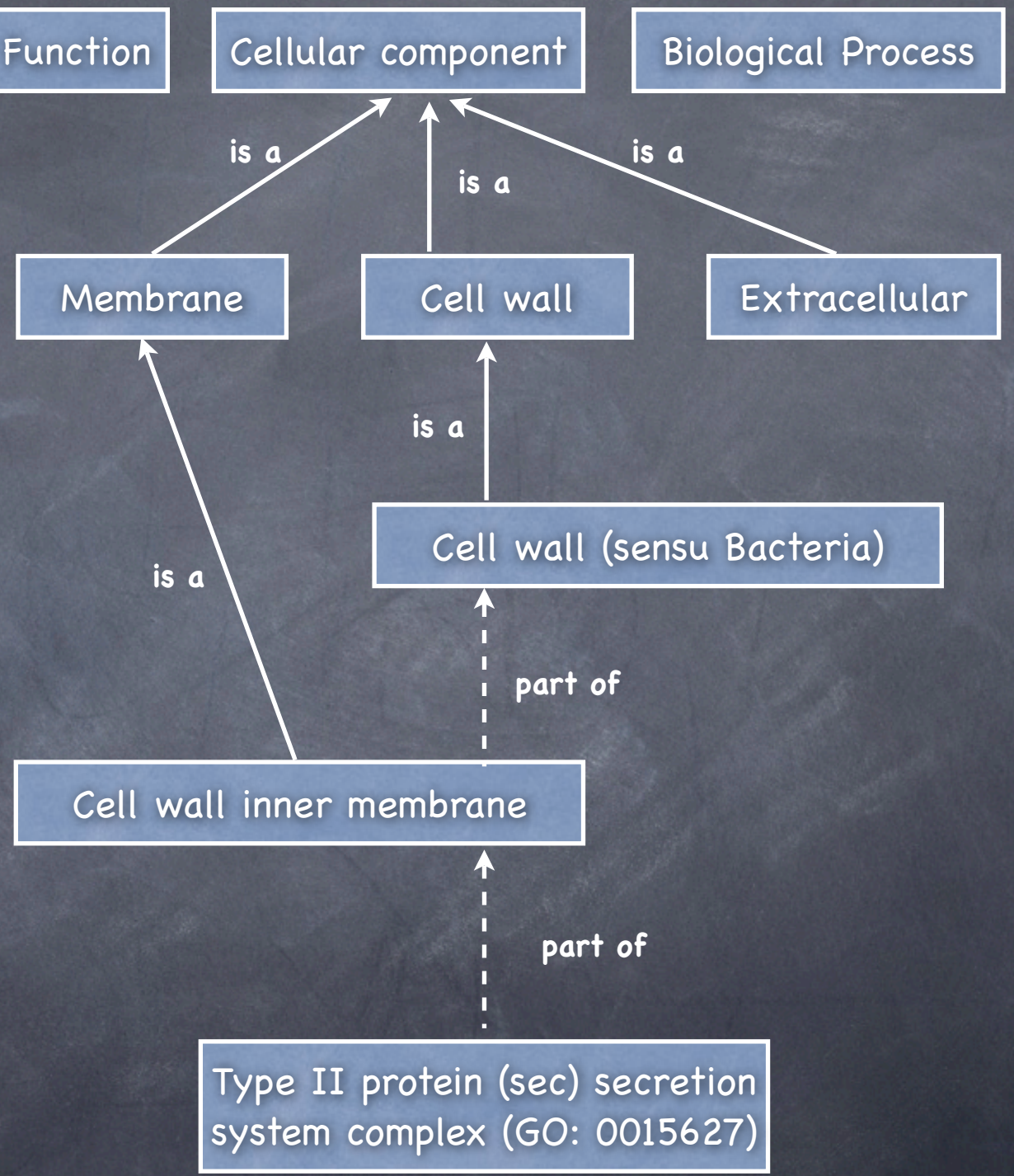
Idiosyncrasies of Bioinformatics data

- Data are complex to model (many different data types)
- New types of data emerge regularly (Data analysis generates new data that also have to be modeled and integrated)
- Raw data must be archived (The terabyte of bioinformatics data consists of a large number of objects)
- Data are updated very frequently, accessed intensively and exchanged very often by researches
- All kinds of users (biologists, programmers, database managers ..) need to issue complex queries
- Data volume grows exponentially, is disseminated in a myriad of different databases and comes in heterogeneous formats

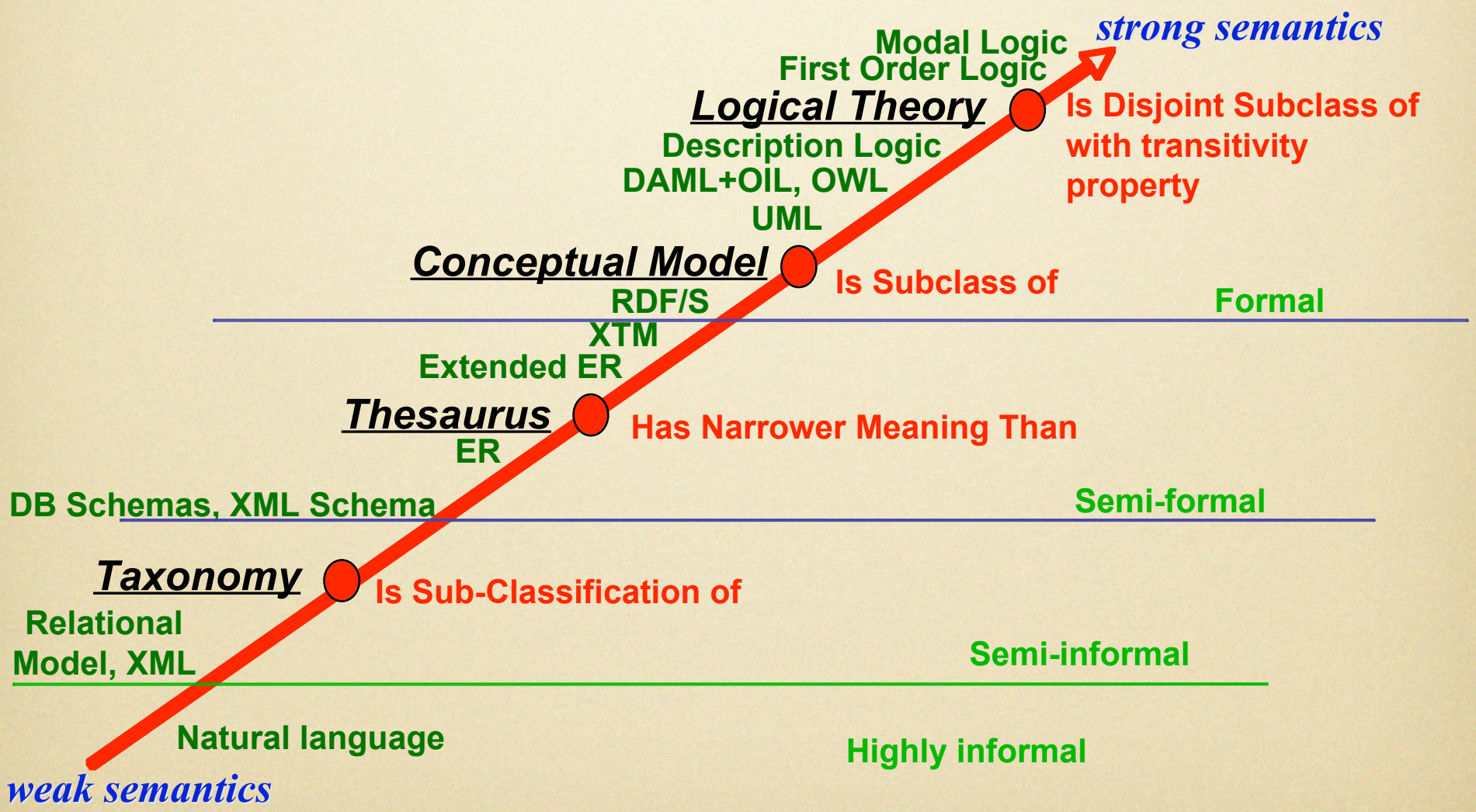
Advantages of XML

- XML is highly flexible (simple to modify a DTD or XML Schema)
- The XML and DTD files are human readable (i.e. they can be easily edited by people with only few computer skills)
- XML is Internet-oriented and has very rich capabilities for linking data (can be used to link databases)
- XML provides an open framework for defining standard specifications (important point because bioinformatics clearly lacks standardization)

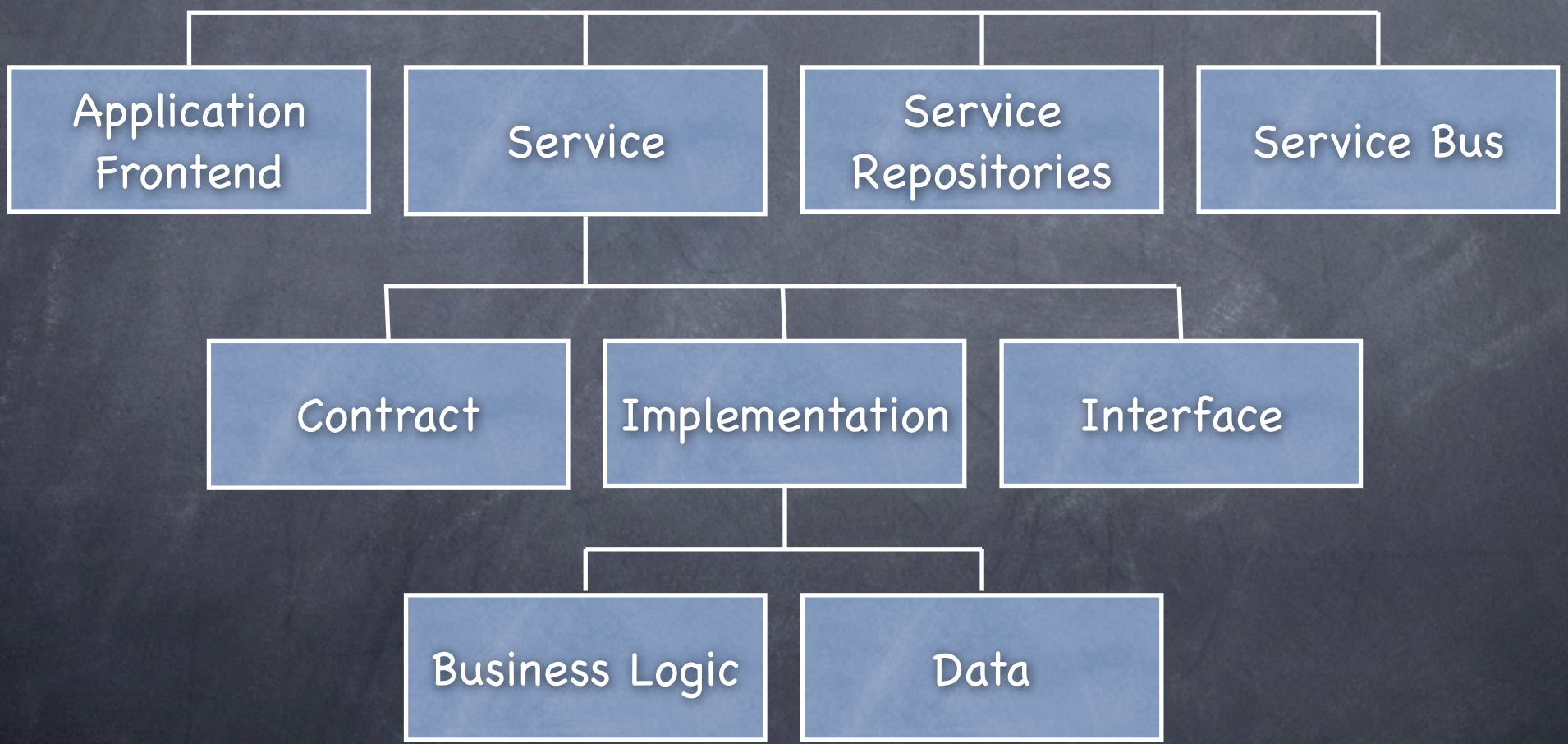

When is the party ?
12/10
whenever
10/12
286
my sensors do not detect activity **now** and only **now** exists
17.22 orbits
timepost sRg.6/Z.a80.K
3 bleeps down the Universe timeline
7.2 cycles after the 3rd anniversary of my creation
not then, but later
2934 units
what is time ?
These bots had better adopt a common **ontology** or I'll never know when the party is.

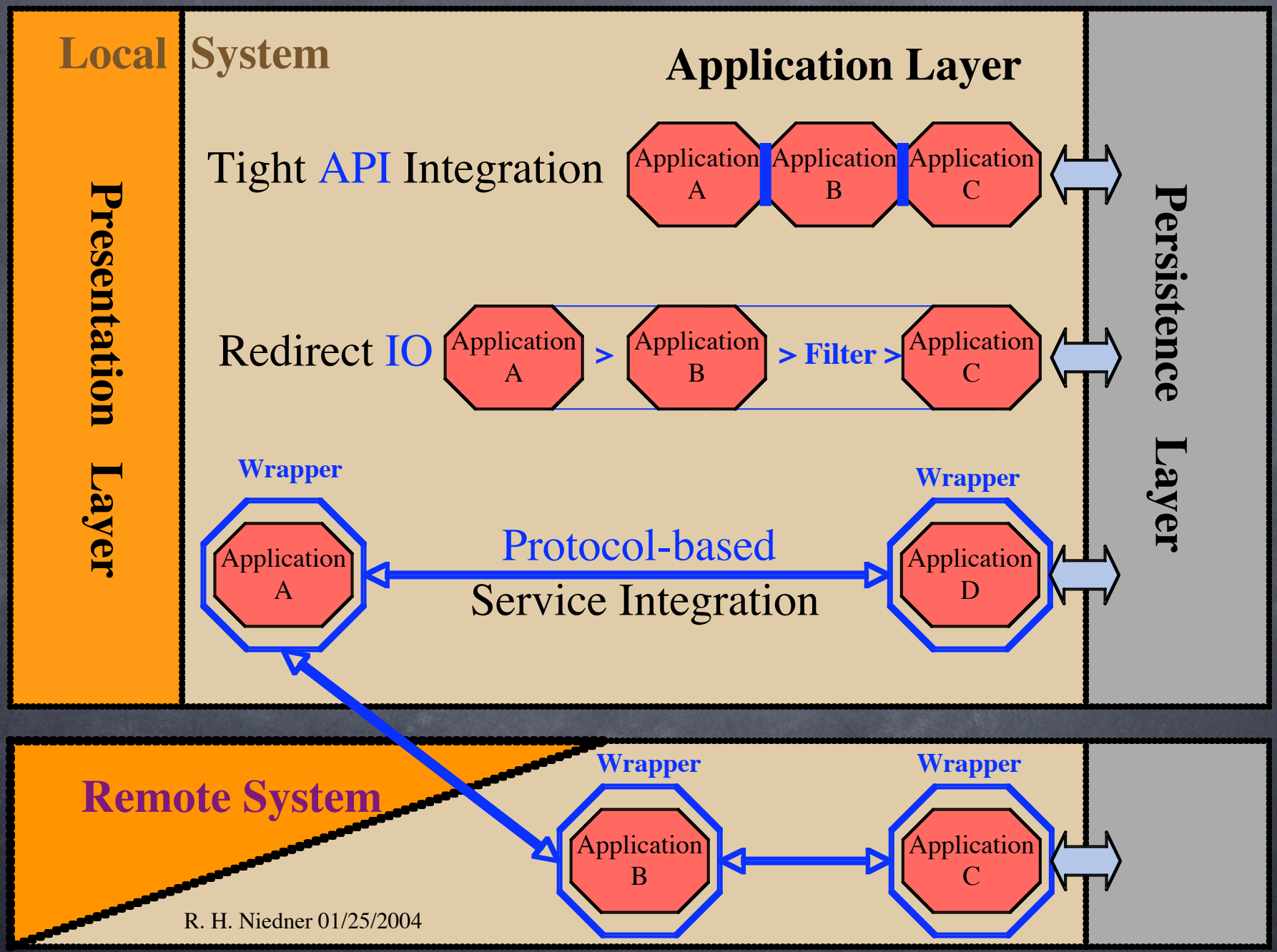

Molecular Function
Cellular component
Biological Process
is a
is a
is a
Membrane
Cell wall
Extracellular
is a
Cell wall (sensu Bacteria)
is a
part of
Cell wall inner membrane
part of
Type II protein (sec) secretion system complex (GO: 0015627)

Ontology Types

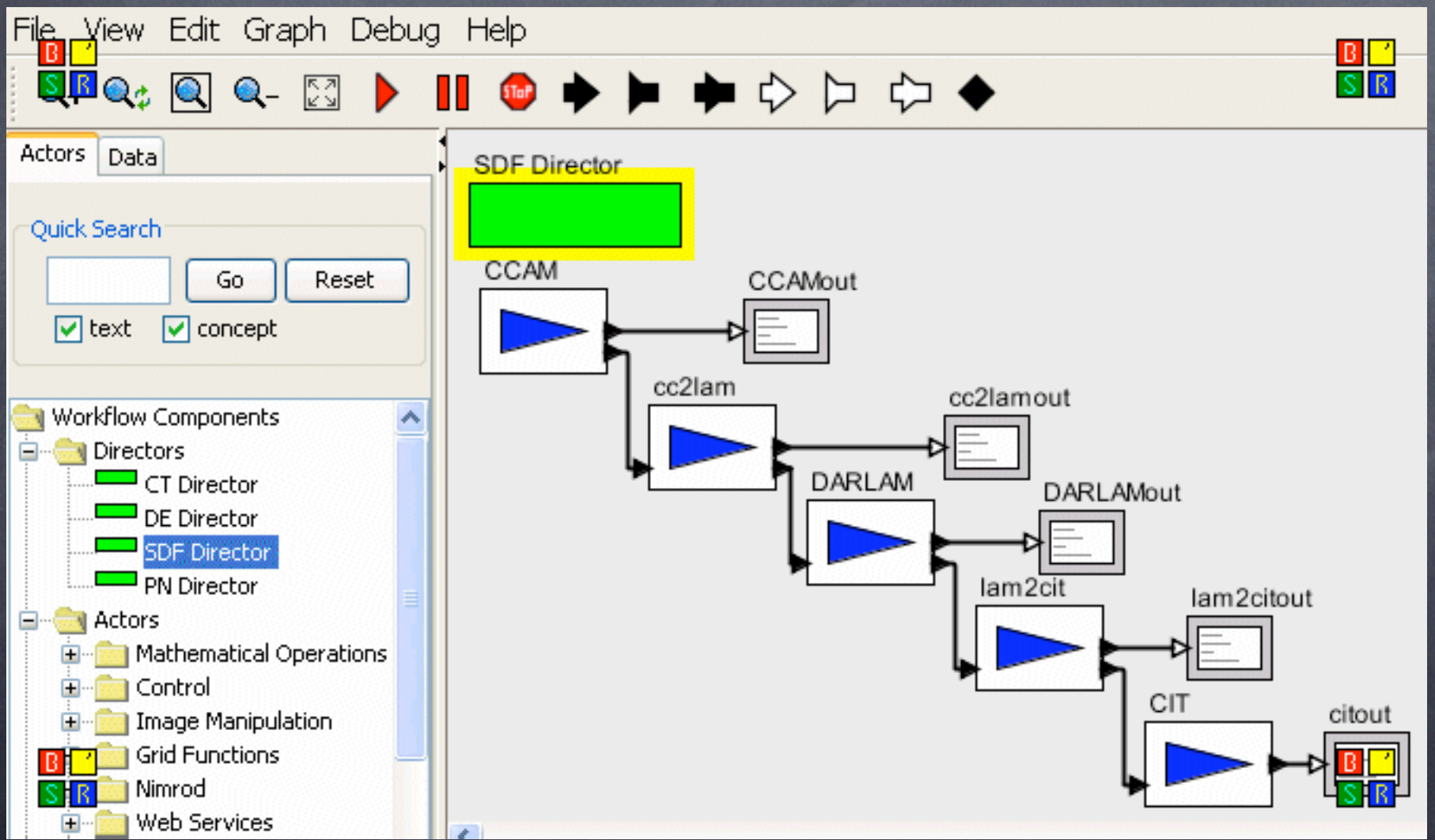


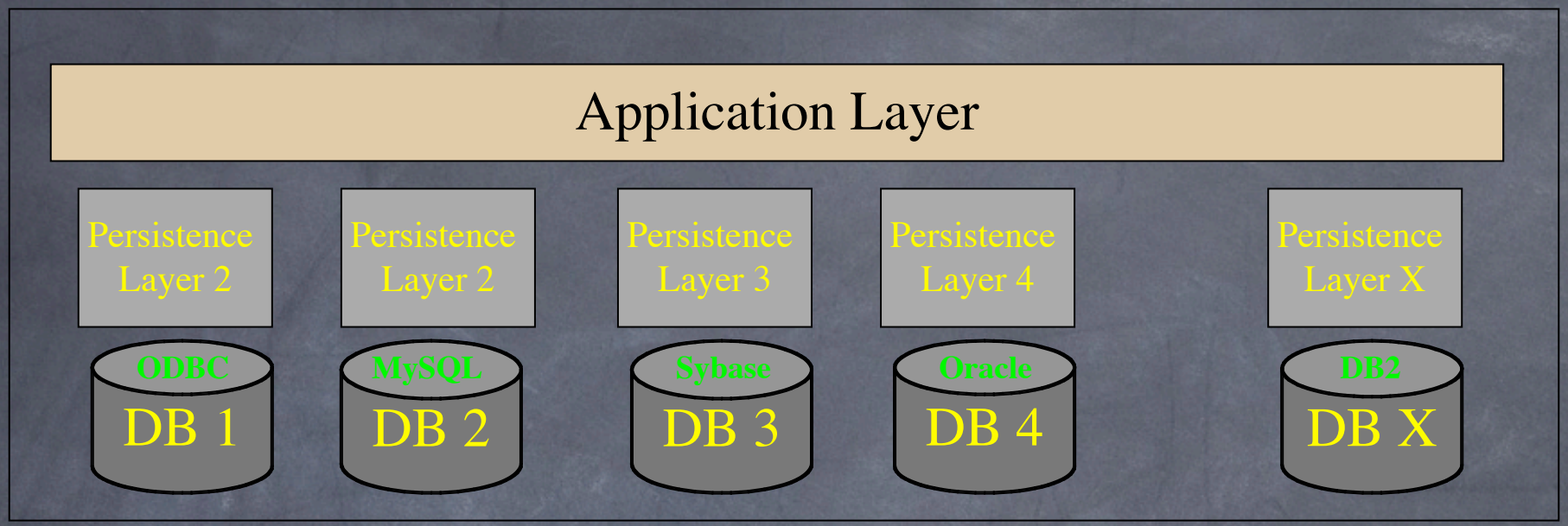
Service Oriented Architecture

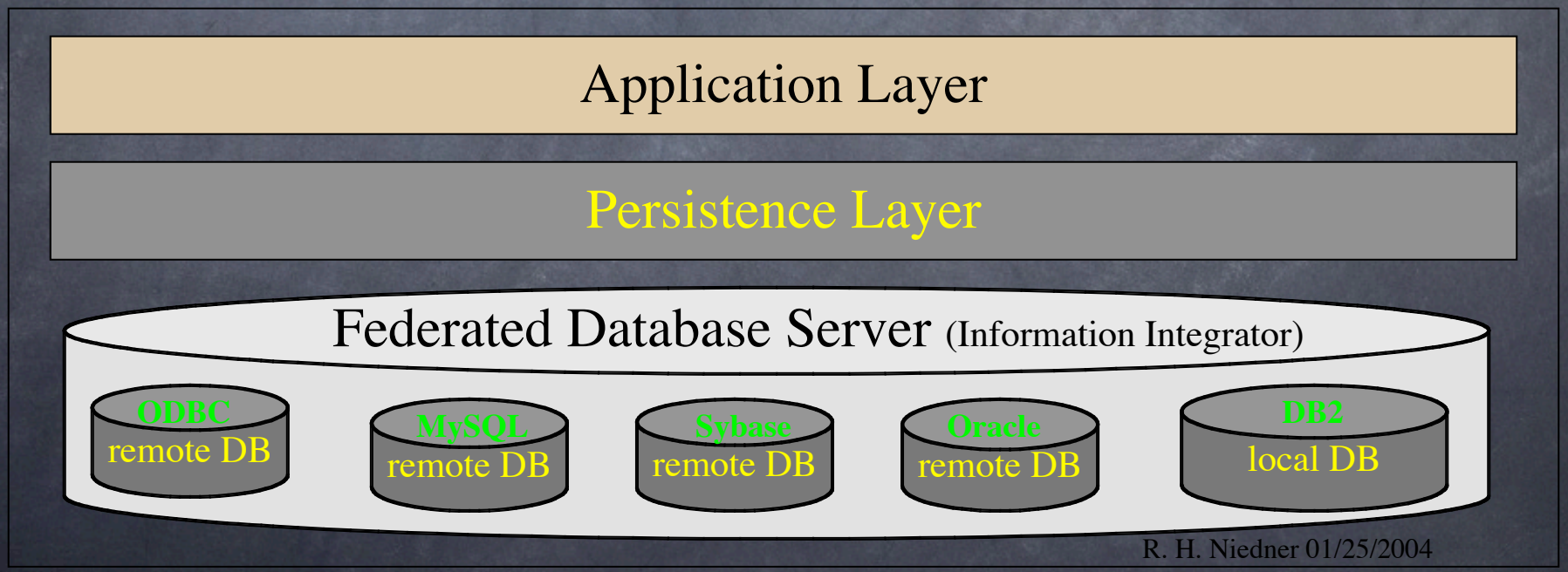


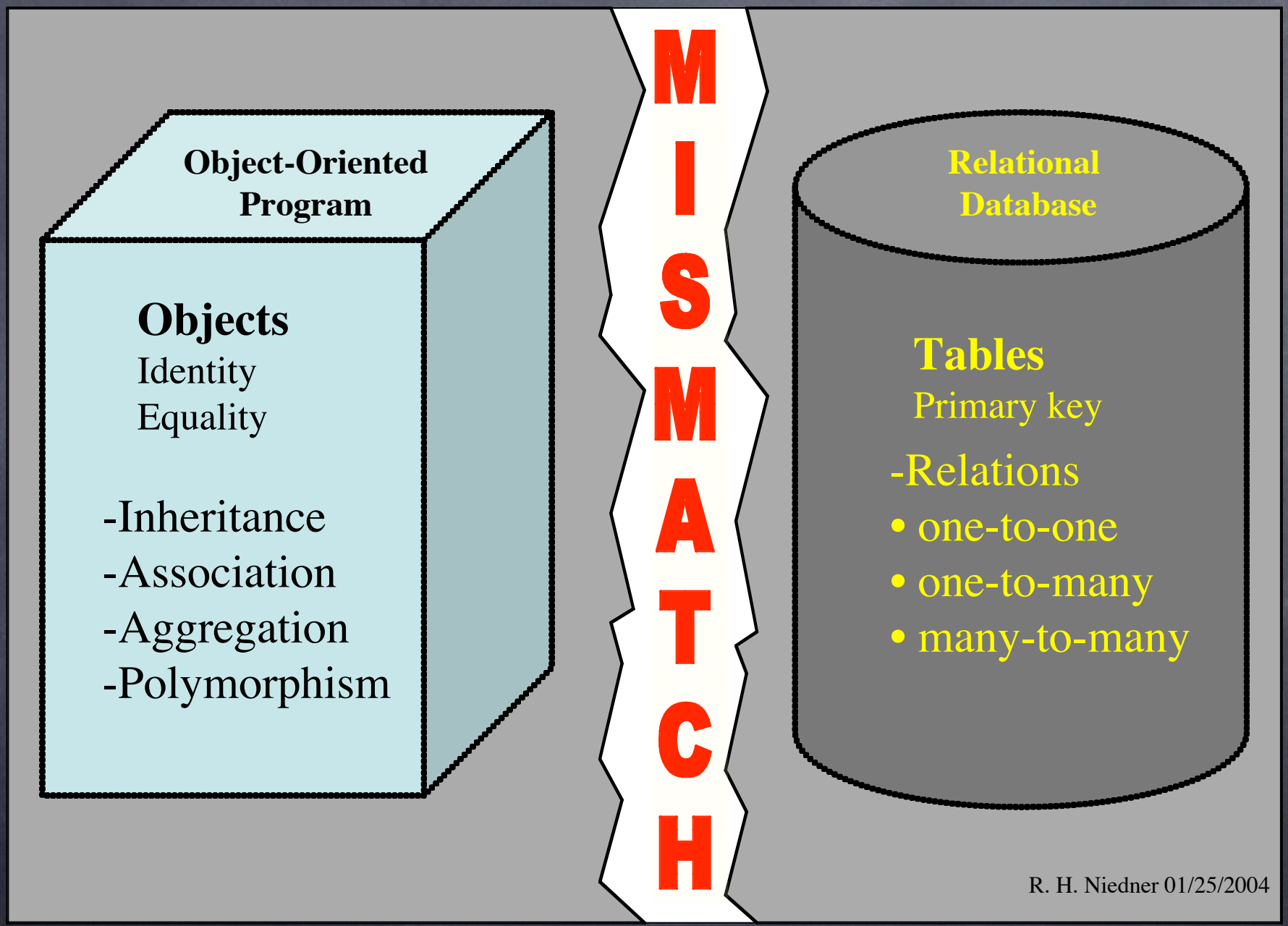

Local System
Application Layer
Presentation Layer
Persistence Layer
Tight API Integration
Application A
Application B
Application C
Redirect IO
Application A
> Application B
> Filter >
Application C
Wrapper
Application A
Protocol-based
Service Integration
Wrapper
Application D
Remote System
Wrapper
Application B
Wrapper
Application C
R. H. Niedner 01/25/2004

Workflows

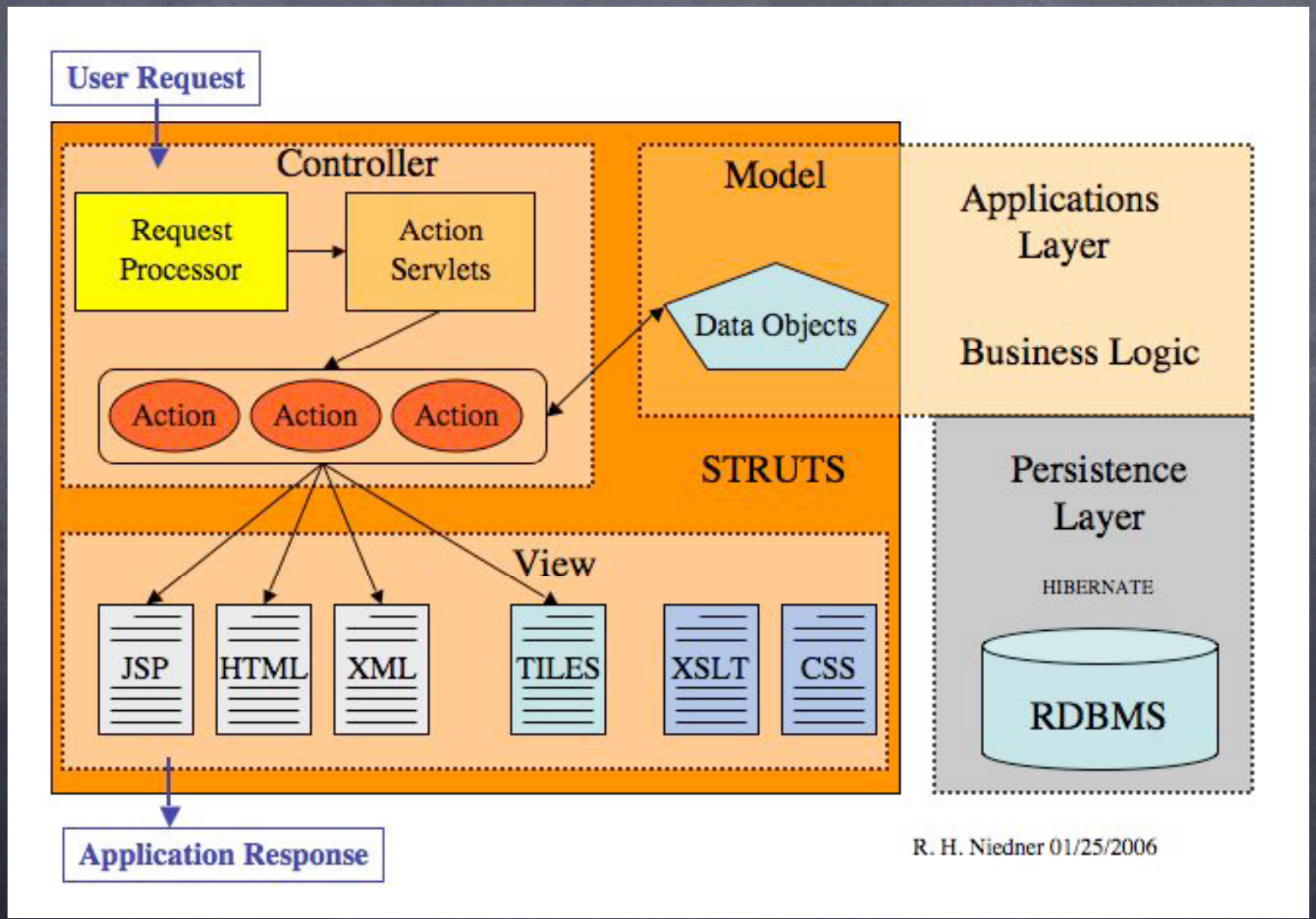

File View Edit Graph Debug Help
Actors
Data
Quick Search
Go
Reset
text
concept
Workflow Components
Directors
CT Director
DE Director
SDF Director
PN Director
Actors
Mathematical Operations
Control
Image Manipulation
Grid Functions
Nimrod
Web Services
SDF Director
CCAM
CCAMout
cc2lam
cc2lamout
DARLAM
DARLAMout
lam2cit
lam2citout
CIT
citout

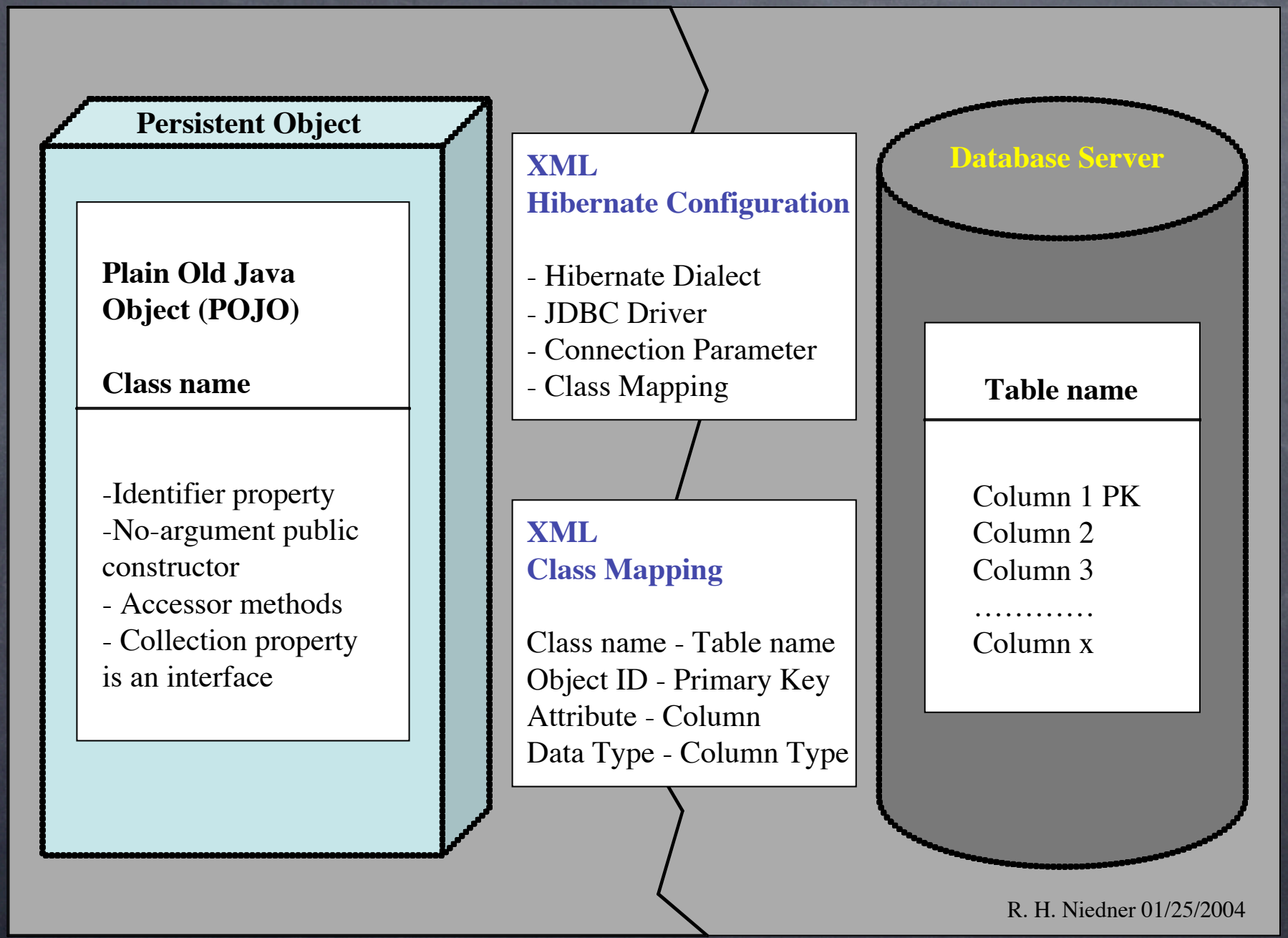

Application Layer
Persistence Layer 2
Persistence Layer 2
Persistence Layer 3
Persistence Layer 4
Persistence Layer X
ODBC
DB 1
MySQL
DB 2
Sybase
DB 3
Oracle
DB 4
DB2
DB X


Application Layer
Persistence Layer
Federated Database Server (Information Integrator)
ODBC
remote DB
MySQL
remote DB
Sybase
remote DB
Oracle
remote DB
DB2
local DB
R. H. Niedner 01/25/2004


Object-Oriented Program
Objects
Identity
Equality
-Inheritance
-Association
-Aggregation
-Polymorphism
MISMATCH
Relational Database
Tables
Primary key
-Relations
• one-to-one
• one-to-many
• many-to-many
R. H. Niedner 01/25/2004

TwinDB Software Architecture




Persistent Object
Plain Old Java Object (POJO)
Class name
-Identifier property
-No-argument public constructor
- Accessor methods
- Collection property is an interface
XML
Hibernate Configuration
- Hibernate Dialect
- JDBC Driver
- Connection Parameter
- Class Mapping
XML
Class Mapping
Class name - Table name
Object ID - Primary Key
Attribute - Column
Data Type - Column Type
Database Server
Table name
Column 1 PK
Column 2
Column 3
……
Column x
R. H. Niedner 01/25/2004

Service Provider

- Cross-domain expertise translating research goals into effective IT-infrastructure
- Analyses of data acquisition, storage, and analyses
- Database design and development
- Application design and development industry standards, scalable techniques, appropriate technologies
- Parallelizing and optimize application code
- Assisting in moving desktop applications into a Client-Server architecture fit to run on the grid