

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

Problem: Biologists need to be able to carry out novel, complex, computations *themselves*.

Why?

Need/Desire to “invent” novel computations...

Need/Desire to understand the methods in detail...  
(for communication/replication/modification/...)

Few biology-savvy programmers (most places)...

Potential communications problems...

Lack of (money to pay) reliable programmers...

Need/Desire to “turn the data over in their hands”...

Goal: Enable biologists to carry out novel, complex, knowledge-based computations *themselves*.

Some biologists can already write complex programs, but most can't.

Why not?

They're busy with other things...

Programming isn't easy...

Most programming languages are terrible...

Data and knowledge is all over the place...

Their formats are also all over-the-place...

The knowledge structures are completely random...

Getting setup is a mess (even for a linux guru)...

Most databases are in the style of "WebTV"...

Biologists are used to "WebTV" computation...

Goal: Enable biologists to carry out novel, complex, knowledge-based computations themselves.

Approach: Make complex biocomputing easier for biologists.

How?

- Provide a uniform programming interface...

- Using a simple, interactive, paradigm...

- To a programmable data/knowledge base...

- With all the relevant data/knowledge preloaded...

- And all the relevant tools (incl. advanced ones)...

- And intelligent semi-automatic programming tools...

- In a biological community...

- All through the web.

Goal: Enable biologists to carry out novel, complex, knowledge-based computations themselves.

Approach: Make complex biocomputing easier for biologists.

What?

KnowOS (Knowledge Operating System):

BioBike

BioDeducta

CACHE

The Prime Directive (first clause):

All data and knowledge can be manipulated by user-written programs that approximate user's natural protocols.

The Prime Directive (second clause):

Users should not have to do any work to make this happen that is not directly relevant to the problem being solved.

**Warning!**  
**~20 slides of Scary-Looking**  
**Programming Detail**  
**Ahead!**

# A Course in Symbolic BioComputing

::

## **Name**

[A Tour of BioBike](#)

[Lisp I: Evaluation](#)

[Lisp II: Data and Iteration](#)

[Functions](#)

[Tables](#)

[Biocomputing I](#)

[Graph R&R](#)

[GO Analysis I](#)

[Microarray Analysis](#)

[Metabolic Simulation I](#)

[Metabolic Simulation II](#)

[Regulation Simulation](#)

[Natural Language I](#)

[Functional Phylogeny](#)

[Model Discovery I](#)

[Model Discovery II](#)

[Reasoning Intro](#)

[More BioReasoning](#)

[Complex Regulatory Reasoning](#)

## **Description**

Getting one's feet wet with BioLisp and the BioBike Listener

The basics of Lisp evaluation.

Lisp data structures and iteration (loops).

An important little Lisp lesson.

Working with data tables

Compring gene function across organisms using annotation and microarrays

Representation and Reasoning about Graphs.

Simple symbolic biocomputing based on the Gene Ontology

Simple microarray data clustering

Introduction to dynamic analysis of metabolic pathways

Dynamic pathway analysis using the Gene Ontology knowledge base

Simulating the cell cycle (based upon a 2004 PNAS publication)

Extracting biological relationships from PubMed using simple statistics

Exploring phylogeny based upon gene function

Reverse engineering of regulatory networks

Model discovery through partial correlations in microarray data

Introduction to automated reasoning using Snark

Re-understanding the Elhai problem in reasoning terms

Using Snark to reason about the cell cycle



# BioBike Conceptual Architecture

## Web-Based Programming Layer

A *simple* programming language to be used by biologists to answer specific questions regarding the integration of their data with the concepts below.

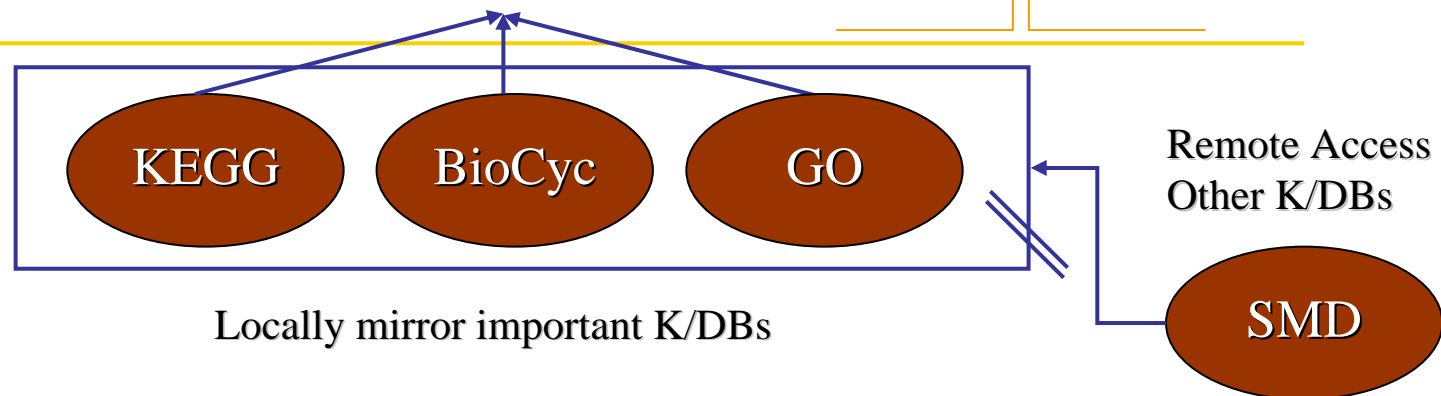
## Computed Concepts Layer

An ever-expanding library of computations that produce complex, virtual, biological concepts, such as pathways, complexes, regulons, etc.

## Unified Basic Concepts Layer

Structures provided for important biological concepts: e.g., reactions, molecules, enzymes, experiments, expression-levels, etc.

## Integrated K/DB Layer



Standard analytic tools, plus discovery tools that combine knowledge and data under user control.

http://nostoc.stanford.edu:8002/redisplay.html?uid=8&pkg=JSHRAGER - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address <http://nostoc.stanford.edu:8002/redisplay.html?uid=8&pkg=JSHRAGER#TAG> Go Links Norton AntiVirus

Back Forward Stop Refresh Home Search Favorites Media History Print Edit Discuss Dell Home Messenger

[11 ~~11~~ 1105phosphoenolpyruvatecarboxylase](#) [1105phosphoenolpyruvatecarboxylase](#)  
#PEPCarboxykinaseC4Photosynthesis #Hexokinase-IndependentSignaling  
#Hexokinase-DependentSignaling #MannokinaseActivity  
#GluconokinaseActivity #WaterPyruvateDikinaseActivity  
#ThiaminPyrophosphokinaseActivity #Selenide,WaterDikinaseActivity  
#RibokinaseActivity #PhosphokinaseActivity  
#PhosphoenolpyruvateCarboxylaseActivity  
#PhosphoenolpyruvateCarboxylateDiphosphateActivity  
#PhosphoenolpyruvateCarboxylateAtpActivity  
#PhosphoenolpyruvateCarboxylateActivity  
#GTPPyrophosphokinaseActivity #XylulokinaseActivity  
#XylulokinaseActivity #RiboseDiphosphatePyrophosphokinaseActivity  
#RhamnulokinaseActivity #6-PhosphofructokinaseActivity  
#1-PhosphofructokinaseActivity #PhosphofructokinaseActivity  
#L-RibulokinaseActivity #L-RibulokinaseActivity  
#L-ArabinokinaseActivity #KetokinaseActivity  
#HexokinaseActivity #GlucokinaseActivity #GalactokinaseActivity  
#FructokinaseActivity #D-RibulokinaseActivity  
#2-Dehydro-3-DeoxygluconokinaseActivity  
#2-Dehydro-3-DeoxygalactonokinaseActivity  
#2-Amino-4-Hydroxy-6-HydroxymethylhydropteridinePyrophosphokinaseActivity)

\*go-frames\*

```
(loop for frame in *go-frames*  
  when (search "kinase" (^fname frame))  
    collect frame)
```

Eval Clear Matching paren context: (loop for frame ...

BioLingua Listener v1.0  
[Lisp Spec](#) [BioLingua Primitives](#) [Upload facility](#) [Help](#)

Internet

Results/History

Simple Exprs

Complex Exprs

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?





Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

# Count the genes of an organism.

```
 <4> *loaded-organisms*  
:: (#$anabaena_pcc7120 #$musmusculus #$nostoc_punctiforme_atcc29133  
   #$promed4 #$anabaena7120 #$synechocystis6803)  
 <5> (setq s #$synechocystis6803)  
:: #$synechocystis6803  
 <6> (#^genes s)  
:: (#$s110400 #$s110401 #$s110402 #$s110403 #$s110404 #$s110405 #$s110406  
   #$s110408 #$s110409 #$s110410 #$s110412 #$s110413 #$s110414 #$s110415  
   #$s111500 #$s110416 #$s111501 #$s111502 #$s110418 #$s111503 #$s110419  
   #$ss10312 #$s111504 #$s111505 #$s111507 #$s111508 #$s111509 #$ss10318  
   #$s110420 #$s110421 #$s110422 #$s110423 #$s110424 #$s111510 #$s110426  
   #$s111511 #$s110427 #$s111512 #$s110428 #$s111513 #$s111514 #$ss10323  
   #$s111515 #$s111516 #$ss11417 #$ss12501 #$ss12502 #$ss12507 #$s110430  
   #$s110431 #$s111520 #$s110436 #$s111521 #$s111522 #$ss10331 #$s111524  
   #$s111525 #$s111526 #$s111527 #$s111528 #$ss11426 #$s110441 #$s110442  
   #$s110443 #$s110444 #$s110445 #$s111530 #$s110446 #$s111531 #$s110447  
   #$s111532 #$s110448 #$s111533 #$s110449 #$s111534 #$s111535 #$s111536  
   #$s111537 #$s111538 #$slr0400 #$slr0401 #$slr0402 #$slr0404 #$slr0406  
   #$slr0407 #$slr0408 #$ss13615 #$s110450 #$s110451 #$s110454 #$s110455  
   #$s111540 #$s110456 #$s111541 #$ss10350 #$s111542 #$s111543 #$s110459  
   #$ss10352 #$s111544 ...)  
 <7> (length *)  
:: 3264
```

## Frame #sll0415

[Help](#) [FindFrames](#) [LispDocs](#) [BioTutorials](#) [BioClik](#)

[Min](#) [Less](#) [More](#) [Max](#) [Biolistener](#)

**Parents** [\[Hide\]](#)

[#\\$FRAMES](#)

..[#\\$Chemicals](#)

.....[#\\$Macromolecules](#)

.....[#\\$Polynucleotides](#)

.....[#\\$Dna](#)

.....[#\\$Dna-Segments](#)

.....[#\\$genes](#)

.....[#\\$Unclassified-Genes](#)

.....[#\\$sll0415](#)

**Children** [\[Hide\]](#)

[#\\$sll0415](#)

**SuperParts** [\[Hide\]](#)

[#\\$Pcc6803genome](#)

..[#\\$sll0415](#)

**Slots**

DBLINKS	( <a href="#">#\$CY</a> <a href="#">#\$sll0415</a> () () () () ())
fName	sll0415
go-frames	( <a href="#">#\$Atp-BindingCassette[Abc]Transporter</a> <a href="#">#\$ATPBinding</a> <a href="#">#\$ATPBinding</a> <a href="#">#\$Transport</a> )
Left-End-Position	2540821
organism	<a href="#">#\$synechocystis6803</a>
PRODUCT	<a href="#">#\$Sll0415-Monomer</a>
Right-End-Position	2540821
SequenceInfo	(2540821 2541885 354 c)
source	ocelot

## BioBike Frame Browser

# How many of those are transporters?

```

<10>> (remove-if-not #'(lambda (g) (member #transport (go-frames g))) (genes s)))
:: (#$sll0415 #sll1521 #sll1533 #sll10477 #sll10478 #sll10479 #sll1599
  #sllr1596 #sllr2045 #sllr2077 #sllr2097 #sll11600 #sll10537 #sll10550
  #sll10594 #sll11685 #sll11699 #sll11017 #sll11082 #sll11087 #sll10616
  #sll10640 #sll11762 #sll10679 #sllr1728 #sllr1730 #sllr1740 #sllr0677
  #sllr0678 #sll11104 #sllr0019 #sll10064 #sll11164 #sllr0063 #sllr0075
  #sllr0079 #sll10771 #sllr0753 #sll10108 #sll11206 #sll10174 #sll10182
  #sll11270 #sllr0161 #sllr1257 #sllr1270 #sllr1908 #sllr0875 #sllr1962
  #sll10217 #sll10219 #sllr0896 #sll10224 #sll11314 #sll11319 #sllr1316
  #sllr1317 #sll11374 #sllr1392 #sll11404 #sll11405 #sll11406 #sll11409
  #sll10355 #sll11450 #sll11451 #sll11452 #sllr0341 #sllr1452 #sllr1453
  #sllr1454 #sllr1455 #sllr1386 #sllr1490)

<11>> (length *)
:: 74

```

Slots	
DBLINKS	( <a href="#">#\$CY</a> <a href="#">#sll0415</a> () () () () ())
fName	sll0415
go-frames	( <a href="#">#\$Atp-BindingCassette[Abc]Transporter</a> <a href="#">#\$ATPBinding</a> <a href="#">#\$ATPBinding</a> <a href="#">#\$Transport</a> )
Left-End-Position	2540821
organism	<a href="#">#\$synechocystis6803</a>
PRODUCT	<a href="#">#sll0415-Monomer</a>
Right-End-Position	2540821
SequenceInfo	(2540821 2541885 354 c)
source	ocelot



Gene Ontology

Pathway Models

Glycolysis

## Instantiate a pathway for an organism.

```
<7>> *loaded-organisms*
:: (#$anabaena_pcc7120 #$musmusculus #$nostoc_punctiforme_atcc29133
   #$promed4 #$anabaena7120 #$synechocystis6803)
<8>> (setq s (first (last *)))
:: #$synechocystis6803
<9>> (length (#^genes s))
:: 3264
→ <10>> (defun instantiate-pathway (pwy org)
         (loop for gene in (#^genes org)
               when (member pwy (#^go-frames gene))
               collect gene))
:: INSTITUTE-PATHWAY
→ <11>> (instantiate-pathway #$glycolysis s)
:: (#$s110587 #$s110593 #$slr1096 #$s110018 #$s111196 #$s110745 #$slr0752
   #$slr0783 #$s111275 #$slr1945 #$slr0884 #$s111342 #$slr1349 #$slr0943
   #$slr0394)
→ <12>> (mapcar #'#^common-name *)
:: ("PYKF" "GLK" "PHDD OR LPD" "CBBA OR CFXA" "PFKA" "PFKA" "ENO" "TPI"
   "PYKF" "YIBO OR PGM" "GAP1" "GAP2" "PGI" "FDA" "PGK")
```



```

<31>> (instantiate-all-pathways s)
:: ((#$AntibioticBiosynthesis ((#$slr0378)))
  ($CellWallBiosynthesis[SensuBacterial
    ((#$sl11833 . "FTSI OR PBPB") ($slr0319)))
  ($LysineBiosynthesisViaDiaminopimelate ((#$slr1665 . "DAPF"))
  ($LysineBiosynthesis
    ((#$sl10504 . "LYSA") ($slr0549 . "ASD") ($slr0550 . "DAPA")
    ($slr1665 . "DAPF") ($sl11058 . "DAPB"))))
  ($ThreonineBiosynthesis
    ((#$sl10455 . "THRA OR HOM") ($sl11760 . "THRB")
    ($sl11172 . "THRC"))))
  ($IsoleucineBiosynthesis
    ((#$sl10455 . "THRA OR HOM") ($slr2072 . "ILVA"))))
  ($LeucineBiosynthesis
    ((#$slr1517 . "LEUB") ($slr0186 . "LEUA") ($sl11444 . "LEUD")
    ($sl11470 . "LEUC"))))
  ($BranchedChainFamilyAminoAcidBiosynthesis
    ((#$slr0452 . "ILVD") ($sl10065 . "ILVN") ($slr0032 . "ILVE")
    ($sl11363 . "ILVC"))))
  ($ArginineBiosynthesis
    ((#$slr0585 . "ARGG") ($slr1022 . "ARGD OR DTU")
    ($sl10080 . "ARGC") ($slr1133 . "ARGH") ($sl11883 . "ARGJ")
    ($slr1898 . "ARGB") ($sl10902 . "ARGF")
    ($sl10370 . "CARB OR PYRA") ($sl11498 . "CARA OR PYRA"))))
  ($GlutamateBiosynthesis
    ((#$sl11502 . "GLTB") ($sl11561 . "PUTA") ($sl11499 . "GLTB"))))
  ($ProlineBiosynthesis
    ((#$slr2035 . "PROB") ($slr0661 . "PROC") ($sl10373 . "PROA"))))
  ($HistidineBiosynthesis
    ((#$slr1560 . "HISS") ($slr0500 . "HISB") ($slr0608 . "HISI")
    ($slr0652 . "HISA") ($slr0682 . "HISD") ($slr0084 . "HISH")
    ($slr1848 . "HISD") ($sl11893 . "HISF") ($sl11958 . "HISC")
    ($sl10900 . "HISG"))))

```

## And for another organism!

→ `<32>> (instantiate-all-pathways #Sanabaena7120)`  
`:: ((#AntibioticBiosynthesis ((#Sal13924)))`  
 `(#CellWallBiosynthesis[SensuBacterial`  
 `((#Salr5045) (#Sal12480) (#Salr0718)))`  
 `(#LysineBiosynthesisViaDiaminopimelate ((#Salr2048)))`  
 `(#LysineBiosynthesis ((#Salr2542) (#Sal13679) (#Salr2048)))`  
 `(#IsoleucineBiosynthesis ((#Salr4232)))`  
 `(#BranchedChainFamilyAminoAcidBiosynthesis ((#Sal12315) (#Salr2771)))`  
 `(#ArginineBiosynthesis`  
 `((#Salr3887) (#Salr5095) (#Salr1155) (#Salr2073) (#Salr3809)))`  
 `(#GlutamateBiosynthesis ((#Salr0540)))`  
 `(#ProlineBiosynthesis ((#Sal12166) (#Salr0488) (#Salr3103)))`  
 `(#HistidineBiosynthesis ((#Salr1965) (#Sal13263) (#Sal11368)))`  
 `(#AlanineBiosynthesis ((#Sal13569)))`  
 `(#AminoAcidBiosynthesis`  
 `((#Sal14644) (#Salr3086) (#Salr3103) (#Salr1207) (#Salr1244)`  
 `(#Salr3644)))`  
 `(#HypusineBiosynthesisFromPeptidyl-Lysine ((#Salr3804)))`  
 `(#PhenylalanineBiosynthesis ((#Salr4334)))`  
 `(#AromaticAminoAcidFamilyBiosynthesis ((#Salr2782) (#Sal10797)))`  
 `(#PurineBaseBiosynthesis ((#Salr3510)))`  
 `(#'DeNovo'PyrimidineBaseBiosynthesis`  
 `((#Salr1912) (#Salr2945) (#Salr2983) (#Sal14272)))`  
 `(#PeptidoglycanBiosynthesis`  
 `((#Sal11663) (#Sal14316) (#Salr1779) (#Salr0094) (#Salr4579)`  
 `(#Sal10174) (#Salr5065) (#Salr2458) (#Salr0477) (#Salr5324)`  
 `(#Salr5326) (#Salr3303) (#Sal12952) (#Sal12981) (#Sal12995)`  
 `(#Sal10980)))`

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

## Frame #GO.Acyl-CoaDehydrogenaseActivity

[Listener](#) [BioDocs](#) [FindFrames](#) [BioFiles](#) [LispDocs](#)

[Min](#) [Less](#) [More](#) [Max](#) [Format: Lisp](#) [Frame->Listener](#)

-> | Slot Name | Slot Value

#^fName	GO.Acyl-CoaDehydrogenaseActivity
#^GO.DBXRefs	EC 1.3.99.3
#^GO.definition	Catalysis of the reaction: a
#^GO.ECRef	<a href="#">#\$EC.1.3.99.3</a>
#^GO.goid	3995
#^GO.Namespace	molecular_function
#^GO.prettyname	acyl-CoA dehydrogenase a
#^Go.products	<a href="#">#\$MOL.2,3-Dehydroacyl-CoA</a>
#^Go.reactants	<a href="#">#\$MOL.Acyl-CoA</a> <a href="#">#\$MOL.A</a>
#^Go.Related-Genes	<a href="#">#\$Cwat.Cw?5697</a> <a href="#">#\$Cwat</a>
#^isA	<a href="#">#\$Go.Reaction</a> <a href="#">#\$GO.Oxide</a>
#^Source	GO
#^subClasses	<a href="#">#\$GO.Very-Long-Chain-Acyl-CoaDehydrogenaseActivity</a> <a href="#">#\$GO.Butyryl-CoaDehydro</a>

### Parents

[#\\$GO.Molecular\\_Function](#)  
[#\\$GO.CatalyticActivity](#)  
[#\\$GO.OxidoreductaseActivity](#)  
[#\\$GO.OxidoreductaseActivity, Acting On The Ch-Ch Group](#)  
[#\\$GO.Acyl-CoaDehydrogenaseActivity](#)  
[#\\$Go.Reaction](#)  
[#\\$GO.Acyl-CoaDehydrogenaseActivity](#)

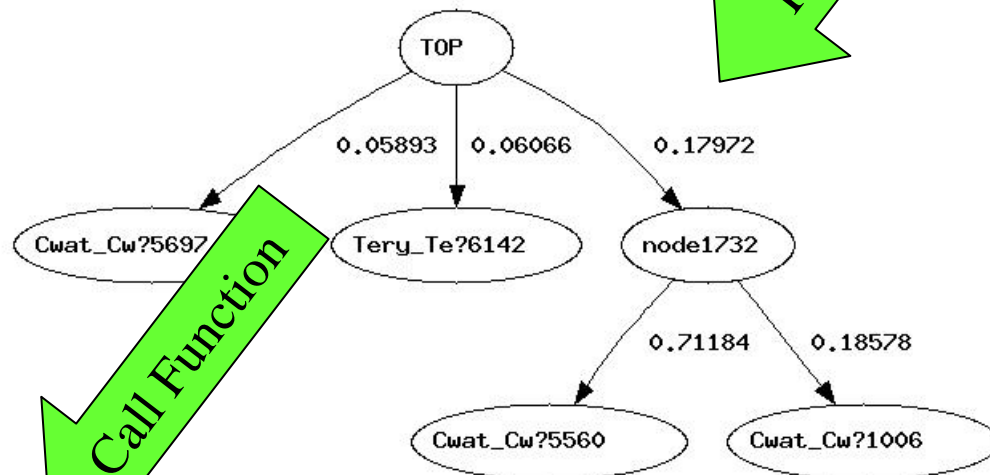
### Children

[#\\$GO.Acyl-CoaDehydrogenaseActivity](#)  
[#\\$GO.Very-Long-Chain-Acyl-CoaDehydrogenaseActivity](#)  
[#\\$GO.Short-Branched-Chain-Acyl-CoaDehydrogenaseActivity](#)  
[#\\$GO.Isovaleryl-CoaDehydrogenaseActivity](#)  
[#\\$GO.Long-Chain-Acyl-CoaDehydrogenaseActivity](#)

```

#$GO.Acyl-CoaDehydrogenaseActivity
#$GO.ProtoporphyrinogenOxidaseActivity
#$GO.15,16-Dihydrobiliverdin:FerredoxinOxidoreductaseActivity ..
<22>> (length *)
:: 1183
<23>> (phylogeny-tree #$GO.Acyl-CoaDehydrogenaseActivity)

```



::

(phylogeny-tree #\$GO.Acyl-CoaDehydrogenaseActivity)

[Enter]

```

(defun phylogeny-tree (go-frame)
  (seegraph
    (run-phylip
      (#^alignments (align (#^Go.Related-Genes go-frame)))
      :labelfn #'first :seqfn #'second)))

```

Eval Clear Info: Package: MT

Clear History

Reindent

[Documentation](#)

[Bio Primitives](#)

[Browse BioFiles](#)

[Web Tools](#)

[Session Logs](#)

[Upload File](#)

[Files: Prev All](#)

[Lisp Hyperspec](#)

[Frame Browser](#)

[Feedback](#)

BioLingua Listener v4.1

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

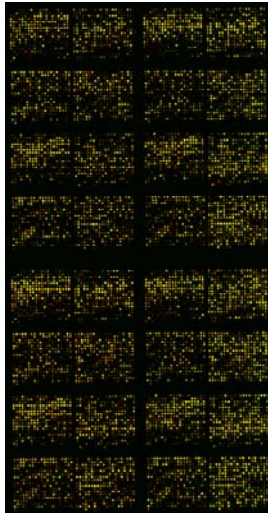
Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

# Knowledge-Based Microarray Casual Analysis

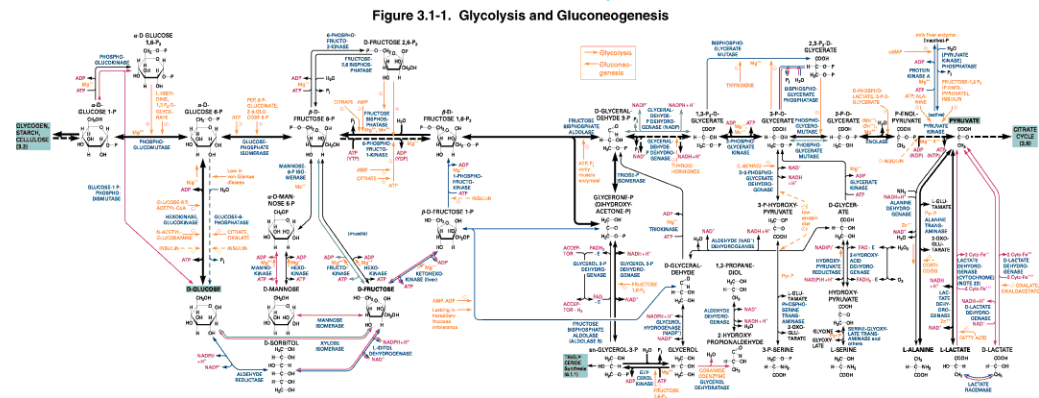
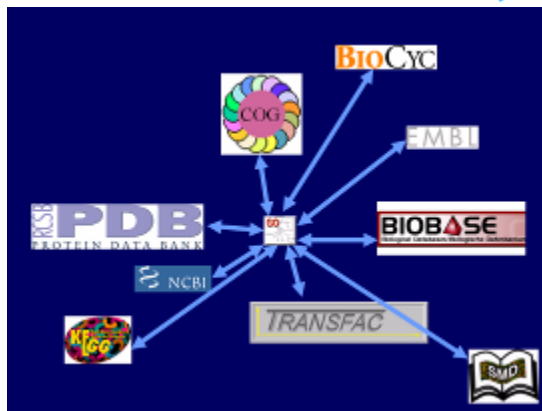


Statistics (R)

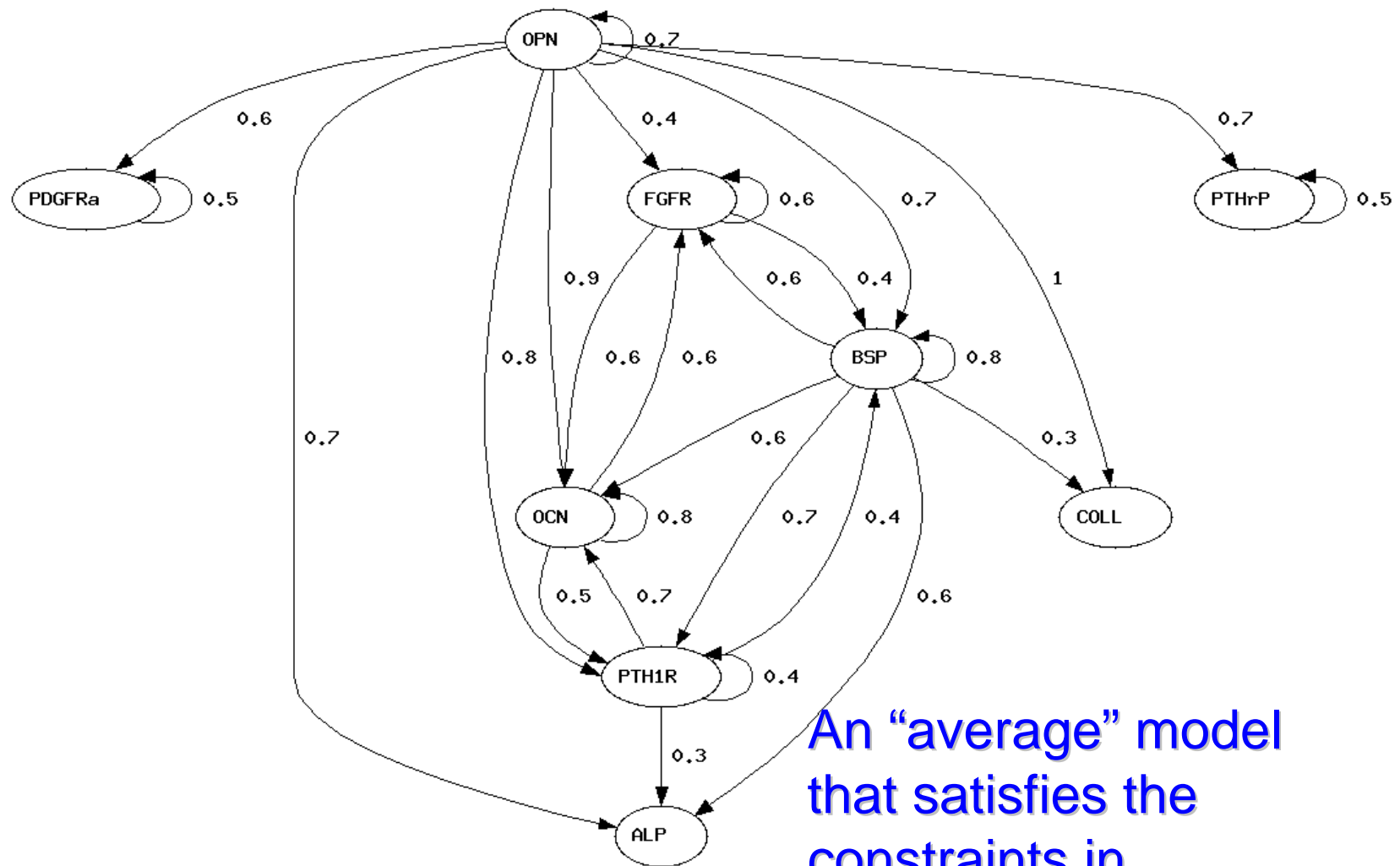
Constraints

Knowledge  
Summarization

Model Space  
Search



→ <11>> (solve-constraints "test.cons")



An “average” model  
that satisfies the  
constraints in  
the given constraint list

::

(solve-constraints "test.cons")

[Enter]

## Get the model in computable form.

➔ **<9>>** (solve-constraints "test.cons" :result-type :list)  
:: (("OPN" "OPN" 0.6) ("OPN" "PDGFra" 0.4) ("OPN" "FGFR" 0.7)  
("OPN" "PTH1R" 0.8) ("OPN" "PTHrP" 0.4) ("OPN" "COLL" 0.7)  
("OPN" "ALP" 0.7) ("OPN" "BSP" 0.7) ("OPN" "OCN" 0.6)  
("PDGFra" "PDGFra" 0.9) ("FGFR" "FGFR" 0.6) ("FGFR" "BSP" 0.3)  
("FGFR" "OCN" 0.6) ("PTH1R" "PTH1R" 0.9) ("PTH1R" "ALP" 0.5)  
("PTH1R" "BSP" 0.3) ("PTH1R" "OCN" 0.6) ("PTHrP" "PTHrP" 0.7)  
("BSP" "FGFR" 0.8) ("BSP" "PTH1R" 0.6) ("BSP" "COLL" 0.6)  
("BSP" "ALP" 0.6) ("BSP" "BSP" 0.8) ("BSP" "OCN" 0.8)  
("OCN" "FGFR" 0.6) ("OCN" "PTH1R" 0.5) ("OCN" "OCN" 0.8))

➔ **<10>>** (setf model (->frames \*))  
:: ((#\$OPN #\$OPN 0.6) (\$OPN \$PDGFra 0.4) (\$OPN \$FGFR 0.7)  
(\$OPN \$PTH1R 0.8) (\$OPN \$PTHrP 0.4) (\$OPN \$COLL 0.7)  
(\$OPN \$ALP 0.7) (\$OPN \$BSP 0.7) (\$OPN \$OCN 0.6)  
(\$PDGFra \$PDGFra 0.9) (\$FGFR \$FGFR 0.6) (\$FGFR \$BSP 0.3)  
(\$FGFR \$OCN 0.6) (\$PTH1R \$PTH1R 0.9) (\$PTH1R \$ALP 0.5)  
(\$PTH1R \$BSP 0.3) (\$PTH1R \$OCN 0.6) (\$PTHrP \$PTHrP 0.7)  
(\$BSP \$FGFR 0.8) (\$BSP \$PTH1R 0.6) (\$BSP \$COLL 0.6)  
(\$BSP \$ALP 0.6) (\$BSP \$BSP 0.8) (\$BSP \$OCN 0.8)  
(\$OCN \$FGFR 0.6) (\$OCN \$PTH1R 0.5) (\$OCN \$OCN 0.8))

(setf model (->frames \*))

[Enter]



# Are two genes *indirectly* connected?

```

:: (#$PTH1R #$OCN 0.6)
→ <15>> (defun indirectly-connected? (from to model)
  (let ((table (make-hash-table)))
    (loop for (from to n) in model
      do (push to (gethash from table)))
    (idc2 from to table)))

```

```

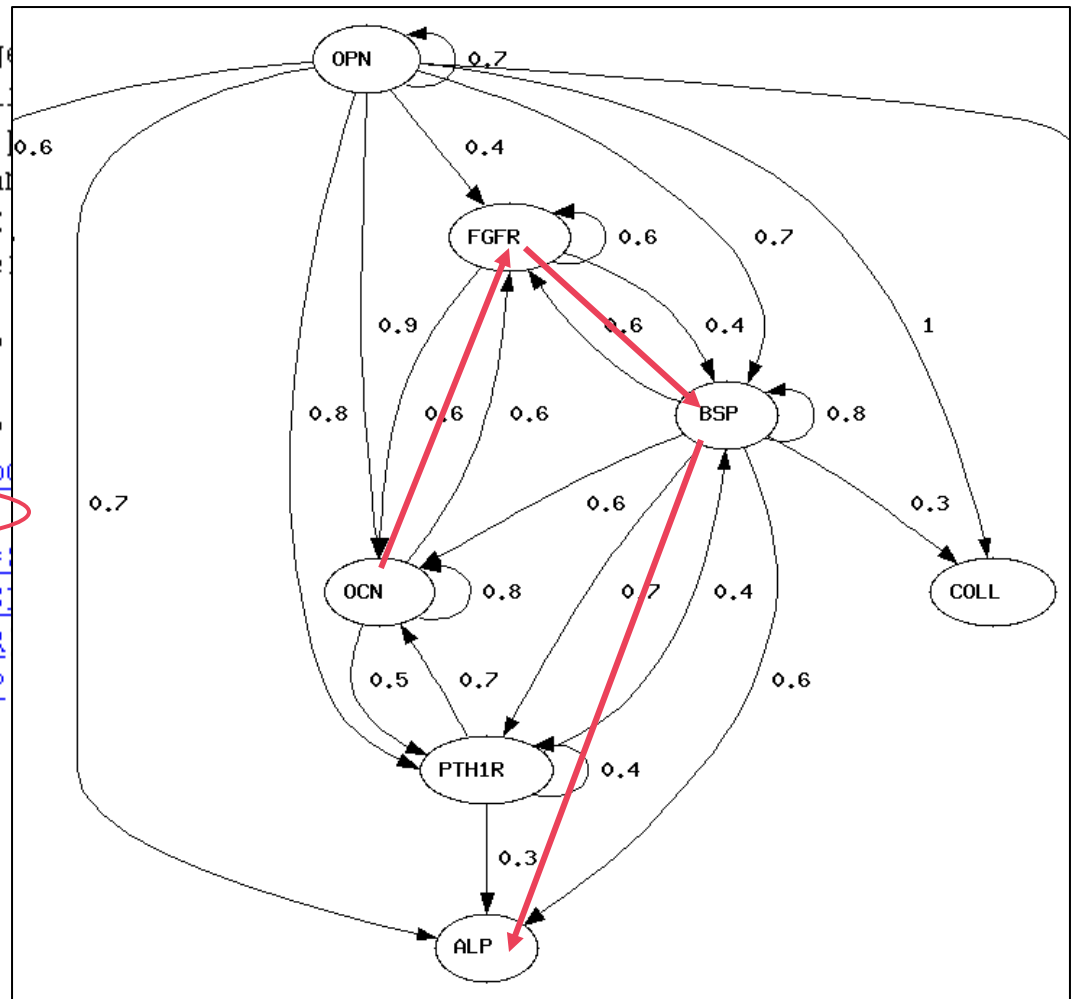
:: INDIRECTLY-CONNECTED?
→ <16>> (defun idc2 (here target)
  (cond ((eq here target) 1)
        ((member here (rest target)) 1)
        (:else (mapcan #'(lambda (target)
                           (idc2 here target))
                        target))))

```

```

:: IDC2
<17>> (indirectly-connected?
:: NIL
→ <18>> (indirectly-connected?
:: ((#$ALP #$BSP #$FGFR #$OCN)
  (#$ALP #$BSP #$FGFR #$OCN)
  (#$ALP #$PTH1R) (#$ALP #$BSP)
  (#$ALP #$PTH1R #$OCN #$FGFR)
  (#$ALP #$BSP #$FGFR) (#$ALP

```



# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

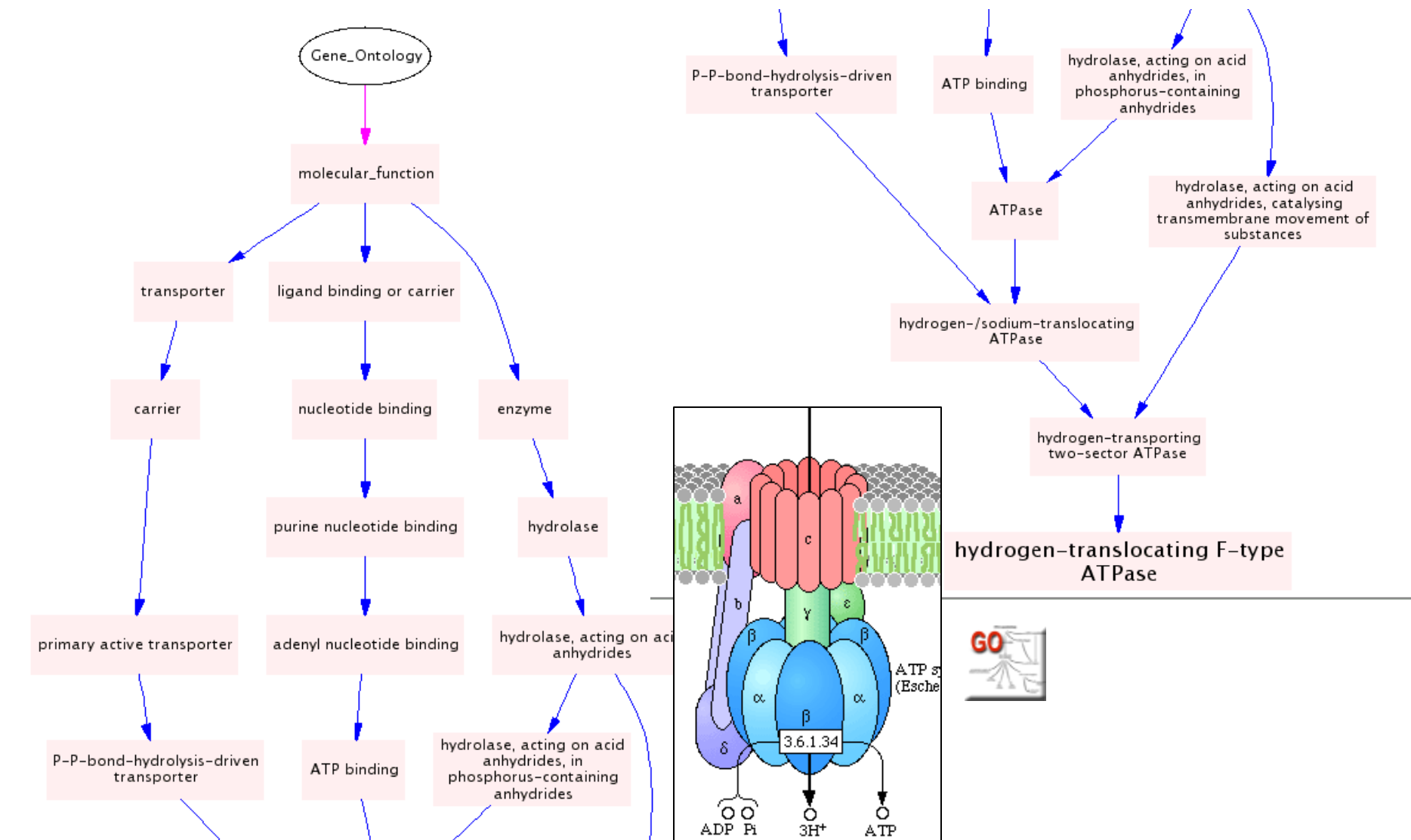
Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

# Current representational practice:



From GenNav, the NIH Gene Ontology Browser

# Goal: “Runnable” Biochemistry

```
(photosynthesis isa process with
  inputs (chloroplast-inside.water everywhere.light chloroplast-outside.nadph+
          chloroplast-outside.adp chloroplast-outside.pi)
  outputs (chloroplast-outside.atp chloroplast-outside.nadph everywhere.o2)
  implemented-by photosystem)

(photosystem composition (psii antenna-array atpase pq-pool))

(light-absorption isa process with
  inputs (everywhere.light)
  outputs (chlorophyll.energy)
  function absorption
  implemented-by chlorophyll)

(light-energy-concentration isa process with
  outputs psii.energy
  driver chlorophyll.energy
  function concentration
  implemented-by antenna-array)

(psii-water-breakdown isa process with
  inputs (chloroplast-inside.water)
  driver psii.energy
  outputs (psii.e- psii.e- chloroplast-inside.h+ chloroplast-inside.o2)
  function molecular-splitting
  implemented-by psii)

(psii-pq-reduction isa process with
  inputs (psii.e- chloroplast-membrane.h+ chloroplast-membrane.plastoquinone)
  outputs (chloroplast-membrane.plastoquinol)
  function reduction
  implemented-by psii
  inhibited-by dcmu)
```

# Frame #Glycolysis

Assume-Unique-Enzymes	no
Common-Name	glycolysis
definition	The breakdown of a monosaccharide (generally glucose) into simpler components, including pyruvate. definition_
Deltag0	-8.5d0
fName	Glycolysis
Goid	6096
In-Pathway	( <a href="#">#\$Glucfermen-Pwy</a> <a href="#">#\$Glycolysis+Citric-Acid-Pwy</a> )
IsaLevel	5
Net-Reaction-Equation	Glucose + 2 Pi + 2 ADP + 2 NAD = 2 pyruvate + 2 ATP + 2 NADH + 2 H + 2 H(2)O
Pathway-Links	(( <a href="#">#\$Pyruvate</a> <a href="#">#\$Fermentation-Pwy</a> <a href="#">#\$Amino-Acid-Biosynthesis</a> ) ( <a href="#">#\$Glc-6-P</a> <a href="#">#\$Pentose-P-Pwy</a> ))
PREDECESSORS	(( <a href="#">#\$Pglucisom-Rxn</a> ) ( <a href="#">#\$6pfructphos-Rxn</a> <a href="#">#\$Pglucisom-Rxn</a> ) ( <a href="#">#\$F16aldolase-Rxn</a> <a href="#">#\$6pfructphos-Rxn</a> ) ( <a href="#">#\$Trioisomerization-Rxn</a> <a href="#">#\$F16aldolase-Rxn</a> ) ( <a href="#">#\$Gapoxnphosphn-Rxn</a> <a href="#">#\$Trioisomerization-Rxn</a> ) ( <a href="#">#\$F16aldolase-Rxn</a> ) ( <a href="#">#\$Phosglyphos-Rxn</a> <a href="#">#\$Gapoxnphosphn-Rxn</a> ) ( <a href="#">#\$3pgarearr-Rxn</a> <a href="#">#\$Phosglyphos-Rxn</a> ) ( <a href="#">#\$2pgadehydrat-Rxn</a> <a href="#">#\$3pgarearr-Rxn</a> ) ( <a href="#">#\$Pepdephos-Rxn</a> <a href="#">#\$2pgadehydrat-Rxn</a> ))
Prettyname	glycolysis
Reaction-List	( <a href="#">#\$Pglucisom-Rxn</a> <a href="#">#\$6pfructphos-Rxn</a> <a href="#">#\$F16aldolase-Rxn</a> <a href="#">#\$Trioisomerization-Rxn</a> <a href="#">#\$Gapoxnphosphn-Rxn</a> <a href="#">#\$Phosglyphos-Rxn</a> <a href="#">#\$3pgarearr-Rxn</a> <a href="#">#\$2pgadehydrat-Rxn</a> <a href="#">#\$Pepdephos-Rxn</a> )
source	ocelot
Super-Pathways	( <a href="#">#\$Glucfermen-Pwy</a> <a href="#">#\$Glycolysis+Citric-Acid-Pwy</a> )
SYNONYMS	Embden-Meyerhof pathway

# Frame #Pglucisom-Rxn

Slots	
Common-Name	Glucose-6-phosphate isomerase
DBLINKS	(( <a href="#">#P T51720</a> () pkarp 3199817462) ( <a href="#">#P T46970</a> () pkarp 3199817462) ( <a href="#">#P T44843</a> () pkarp 3199817462) ( <a href="#">#P T43196</a> () pkarp 3199817462) ( <a href="#">#P T14631</a> () pkarp 3199817462) ( <a href="#">#P T09153</a> () pkarp 3199817462) ( <a href="#">#P T05572</a> () pkarp 3199817462) ( <a href="#">#P T03948</a> () pkarp 3199817462) ( <a href="#">#P T02094</a> () pkarp 3199817462) ( <a href="#">#P S75607</a> () pkarp 3199817462) ( <a href="#">#P S73908</a> () pkarp 3199817462) ( <a href="#">#P S58164</a> () pkarp 3199817462) ( <a href="#">#P S57831</a> () pkarp 3199817462) ( <a href="#">#P S57830</a> () pkarp 3199817462) ( <a href="#">#P S41808</a> () pkarp 3199817462) ( <a href="#">#P S41807</a> () pkarp 3199817462) ( <a href="#">#P S41806</a> () pkarp 3199817462) ( <a href="#">#P S235</a> pkarp 3199817462) ( <a href="#">#P NUZQF</a> () pkarp 3199817462) ( <a href="#">#P NUVKL</a> () pkarp 3199817462) ( <a href="#">#P NUPG</a> () pkarp 3199817462) ( <a href="#">#P NUMS</a> () pkarp 3199817462) ( <a href="#">#P NUBY</a> () pkarp 3199817462) ( <a href="#">#P NUBSSA</a> () pkarp 3199817462) ( <a href="#">#P I48073</a> () pkarp 3199817462) ...15 more... )
Deltag0	0.4d0
ec-number	5.3.1.9
Enzymatic-Reaction	<a href="#">#PEnzrxn-256</a>
fName	Pglucisom-Rxn
In-Pathway	( <a href="#">#PSucsyn-Pwy</a> <a href="#">#P341-Pwy</a> <a href="#">#PGlycolysis</a> <a href="#">#PGlucone-Pwy</a> <a href="#">#P124-Pwy</a> <a href="#">#P125-Pwy</a> )
LEFT	<a href="#">#PGlc-6-P</a>
RIGHT	<a href="#">#PFructose-6p</a>
source	ocelot
SYNONYMS	(Hexose monophosphate isomerase Phosphohexoisomerase Phosphoglucoisomerase Phosphosaccharom Hexosephosphate isomerase Oxoisomerase Phosphohexomutase Phosphohexose isomerase Phosphoglu isomerase)



## Find the genes involved in glycolysis, and their reactions.

```
➡ <38>> (remove-if-not #'(lambda (g) (member #\$glycolysis (^go-frames g))) (^genes  
:: (#\$sll10587 #\$sll10593 #\$slr1096 #\$sll10018 #\$sll11196 #\$sll10745 #\$slr0752  
#\$slr0783 #\$sll11275 #\$slr1945 #\$slr0884 #\$sll11342 #\$slr1349 #\$slr0943  
#\$slr0394)  
<39>> (first *)  
:: #\$sll10587  
<40>> (^product *)  
:: #\$sll10587-Monomer  
<41>> (^catalyzes *)  
:: #\$Enzrxn-491  
<42>> (^reaction *)  
:: #\$Pepdephos-Rxn  
<43>> (^left *)  
:: (#\$Pyruvate \$Atp)  
➡ <44>> (defun extract-reaction (gene)  
  (ignore-errors (let ((reaction (^reaction (^catalyzes (^product gene)))))  
    (list gene (^left reaction) (^right reaction)))))  
:: EXTRACT-REACTION  
➡ <45>> (mapcar #'extract-reaction (hop 38))  
:: ((#\$sll10587 (#\$Pyruvate \$Atp)  
  (#\$Phospho-Enol-Pyruvate \$Adp \$PROTON))  
NIL  
  (#\$slr1096 (#\$Red-Dihydrolipoamide \$NAD)  
  (\$Ox-Lipoamide \$Nadh \$PROTON))  
  (#\$sll10018 #\$Fructose-16-Diphosphate  
  (#\$Dihydroxy-Acetone-Phosphate \$GAP))  
NIL NIL (#\$slr0752 #\$2-Pg (#\$Phospho-Enol-Pyruvate \$WATER))  
  (#\$slr0783 \$GAP #\$Dihydroxy-Acetone-Phosphate)  
  (#\$sll11275 (#\$Pyruvate \$Atp)  
  (#\$Phospho-Enol-Pyruvate \$Adp \$PROTON))  
  (#\$slr1945 \$G3p #\$2-Pg)
```

# “Runnable” Biochemistry

```
<98>> (defun run-chemistry (reactions env)
  (loop with old-env = env
    as new-env = (new-environment reactions old-env)
    until (equal new-env old-env)
    do (setq old-env new-env)
    (print new-env)
    finally (return new-env)))
:: RUN-CHEMISTRY
<99>> env
:: (#$G3p #$Atp)
<100>> rs
:: (#$s110587-reaction #$slr1096-reaction #$s110018-reaction
  #$slr0752-reaction #$slr0783-reaction #$s111275-reaction
  #$slr1945-reaction #$slr0884-reaction #$s111342-reaction
  #$slr1349-reaction #$slr0943-reaction #$slr0394-reaction)
<101>> (run-chemistry rs env)
::
(#$Adp #$DPG #$2-Pg #$G3p #$Atp)
(#$WATER #$Phospho-Enol-Pyruvate #$Adp #$DPG #$2-Pg #$G3p #$Atp)
(#$WATER #$Phospho-Enol-Pyruvate #$Adp #$DPG #$2-Pg #$G3p #$Atp)
```



# Additional KnowOS Features

- Multi-headed multi-threaded architecture
- Built-in (nascent) electronic lab notebook
- Easily access other web-based DBs
- Integrated file system
- Easily integrate any new (linux based) tools
- Easily integrate new knowledge or data
- Complete OO programming model
- Fully compiled language
- Program development tools (editors, debuggers, etc.)
- Easy to deploy functions as new web services
- Online “Live Tutorials” provide a complete biocomputing course!
- Totally open source freeware

# Live Tutorials (A Course in Symbolic BioComputing)

[::](#)

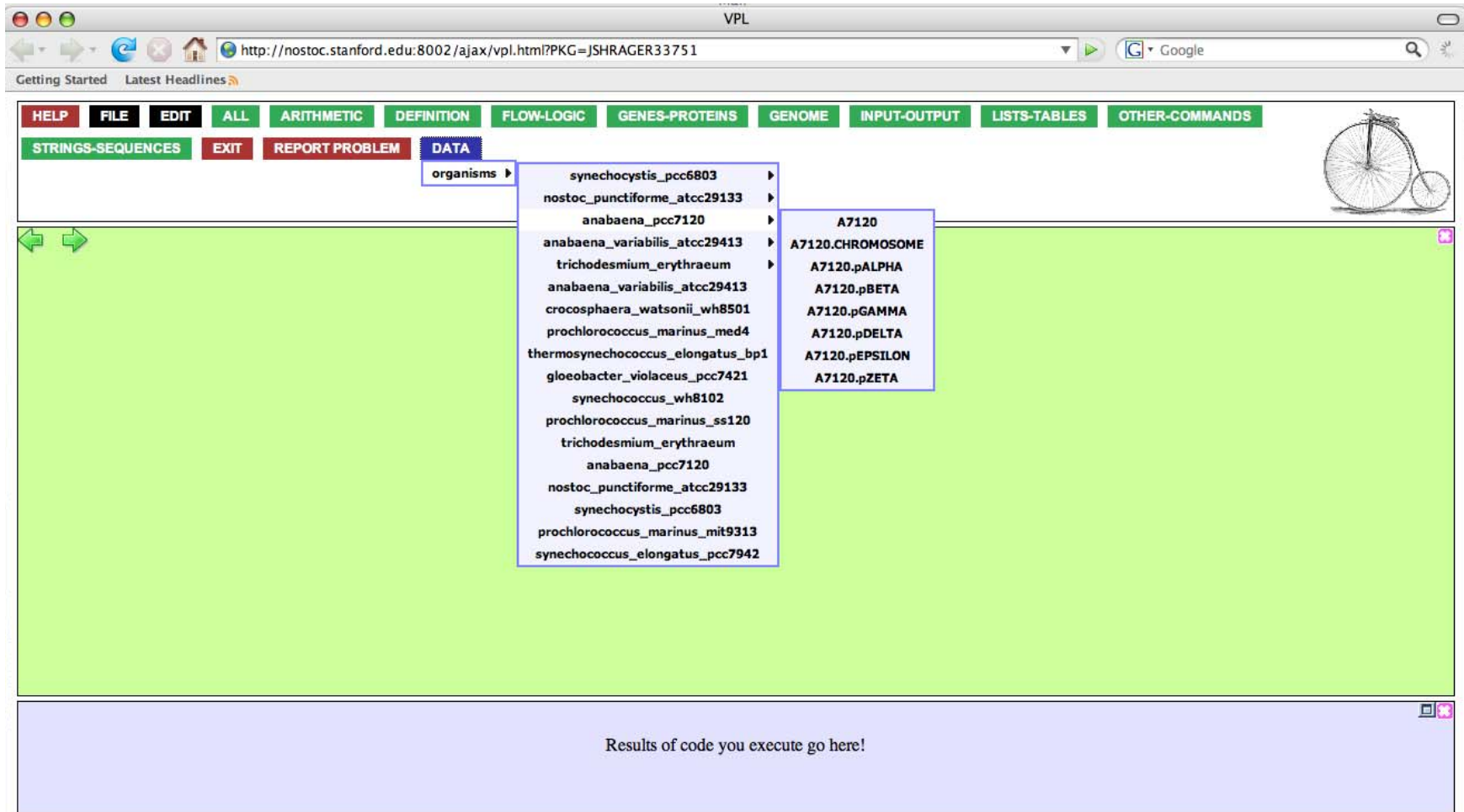
Name	Description
<a href="#"><u>A Tour of BioBike</u></a>	Getting one's feet wet with BioLisp and the BioBike Listener
<a href="#"><u>Lisp I: Evaluation</u></a>	The basics of Lisp evaluation.
<a href="#"><u>Lisp II: Data and Iteration</u></a>	Lisp data structures and iteration (loops).
<a href="#"><u>Functions</u></a>	An important little Lisp lesson.
<a href="#"><u>Tables</u></a>	Working with data tables
<a href="#"><u>Biocomputing I</u></a>	Compring gene function across organisms using annotation and microarrays
<a href="#"><u>Graph R&amp;R</u></a>	Representation and Reasoning about Graphs.
<a href="#"><u>GO Analysis I</u></a>	Simple symbolic biocomputing based on the Gene Ontology
<a href="#"><u>Microarray Analysis</u></a>	Simple microarray data clustering
<a href="#"><u>Metabolic Simulation I</u></a>	Introduction to dynamic analysis of metabolic pathways
<a href="#"><u>Metabolic Simulation II</u></a>	Dynamic pathway analysis using the Gene Ontology knowledge base
<a href="#"><u>Regulation Simulation</u></a>	Simulating the cell cycle (based upon a 2004 PNAS publication)
<a href="#"><u>Natural Language I</u></a>	Extracting biological relationships from PubMed using simple statistics
<a href="#"><u>Functional Phylogeny</u></a>	Exploring phylogeny based upon gene function
<a href="#"><u>Model Discovery I</u></a>	Reverse engineering of regulatory networks
<a href="#"><u>Model Discovery II</u></a>	Model discovery through partial correlations in microarray data
<a href="#"><u>Reasoning Intro</u></a>	Introduction to automated reasoning using Snark
<a href="#"><u>More BioReasoning</u></a>	Re-understanding the Elhai problem in reasoning terms
<a href="#"><u>Complex Regulatory Reasoning</u></a>	Using Snark to reason about the cell cycle

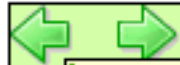





# On Beyond Programming...

- Visual Programming
- Deductive biocomputing
- Natural language biocomputing

# BioBike Visual Programming Language



**HELP****FILE****EDIT****ALL****ARITHMETIC****DEFINITION****FLOW-LOGIC****STRINGS-SEQUENCES****EXIT****REPORT PROBLEM****DATA****CODING-GENES-OF** **biolisp:s6803**  **1>** (S6803.slr0612 S6803.slr0613 S6803.s110558 S6803.s111214 S  
S6803.s111209 S6803.slr1315 S6803.slr1316 S6803.slr1317 S  
S6803.s111204 S6803.s111203 S6803.s111202 S6803.s111409 S  
S6803.s111404 S6803.slr1484 S6803.slr1485 S6803.slr1488 S  
S6803.slr1493 S6803.slr1494 S6803.s111401 S6803.s111400 S

HELP

FILE

EDIT

ALL

ARITHMETIC

DEFINITION

FLOW-LOGIC

GENES-PROTEINS

GENO

STRINGS-SEQUENCES

EXIT

AGGREGATE-ARITHMETIC ▶

BASIC-ARITHMETIC ▶

NUMERICAL-TYPE-CHECKS ▶

STATISTICS ▶

ITEM-OF-RANK

←

→

▶ MEAN

▶ LENGTH-OF

▶

token

EACH

▶ CODING-GENES-OF

▶ biolisp:s6803

▶ 4> 936.86926

▶ 3>

(693 522 702 1077 939 1089 1083 1980 2010 609 1032 1050 807



HELP

FILE

EDIT

EXIT

PROBLEM

ALL

ARITHMETIC

DEFINITION

FLOW-LOGIC

GENES-PROTEINS

GENOME

INPUT-OUTPUT

LISTS-TABLES

OTHER-COMMANDS

STRINGS-SEQUENCES

DATA

FUNCTIONS

> Forget all  
MEAN-GENES

DEFINE-FUNCTION

mean-genes

required

org

...

df-options

df-options

body

MEAN

LENGTH-OF

EACH

token

CODING-GENES-OF

org

LOOP

initializations

controls

for

org

in

\*ALL-CYANOBACTERIA\*

by (opt)

cond (opt)

action (opt)

collect

LIST

org

MEAN-GENES

org

finally (opt)

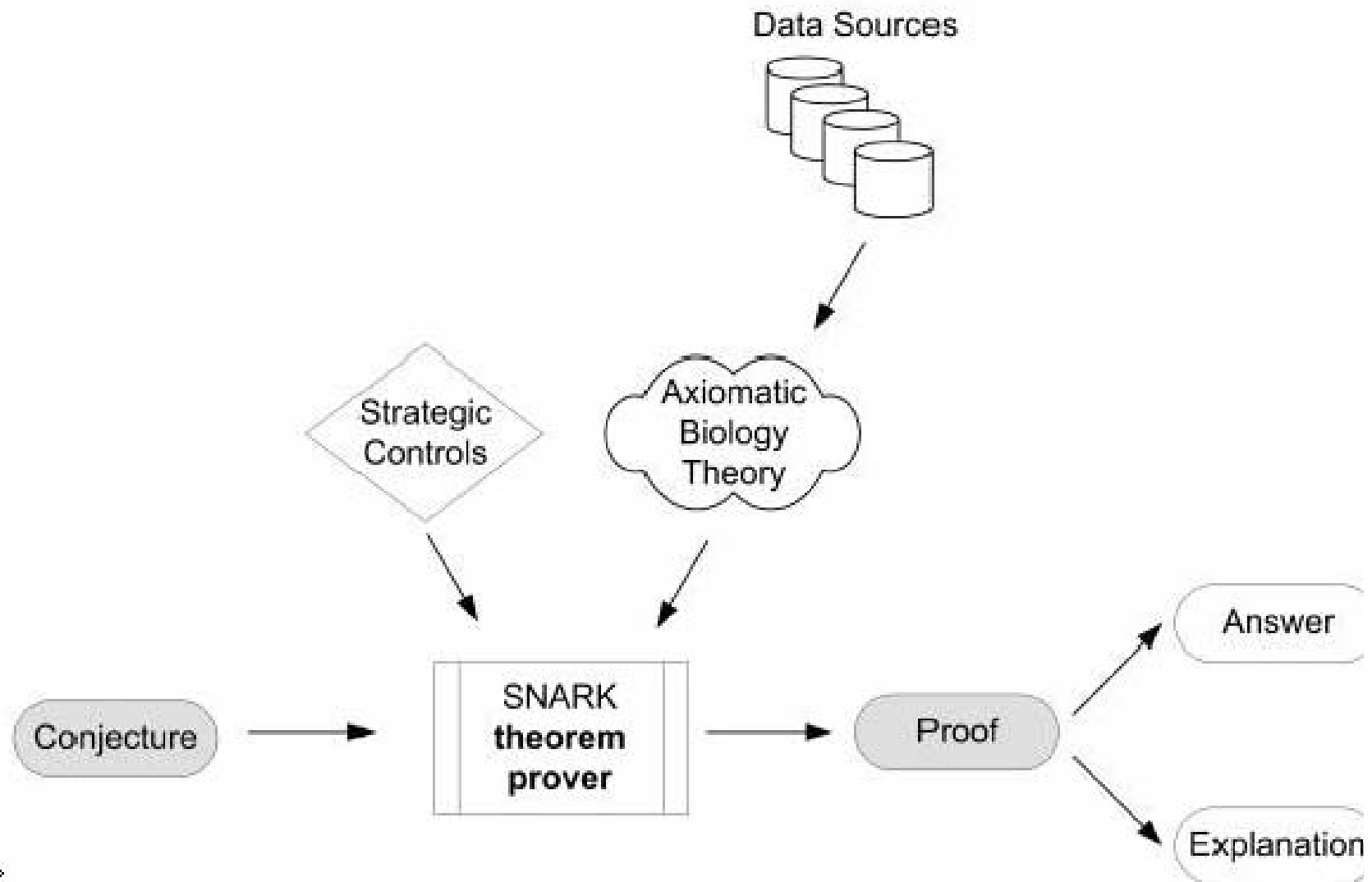
9>

((trichodesmium\_erythraeum 639.9852) (anabaena\_variabilis\_atcc29413 875.13464)  
(anabaena\_pcc7120 966.6737) (nostoc\_punctiforme\_atcc29133 997.52435) (synechocystis\_pcc6803  
936.86926) )



# Toward semi-automatic programming

(BioDeducta)



-->

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

Model Identification: What models fit this data?

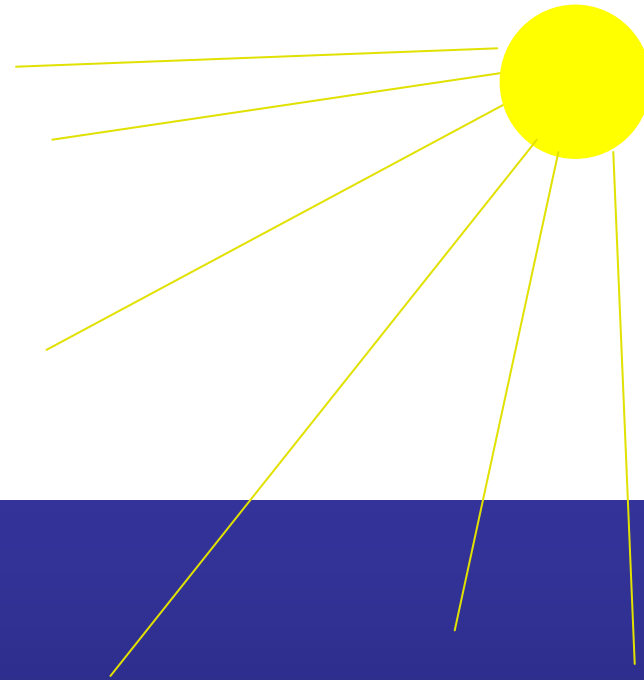
Model Analysis: What does this model predict?

Annotation: What does this gene do?

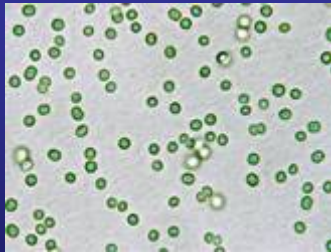
Use and Share Results.

# How do cells control response to light?

I.e., What genes are related to the adaptation to high light?

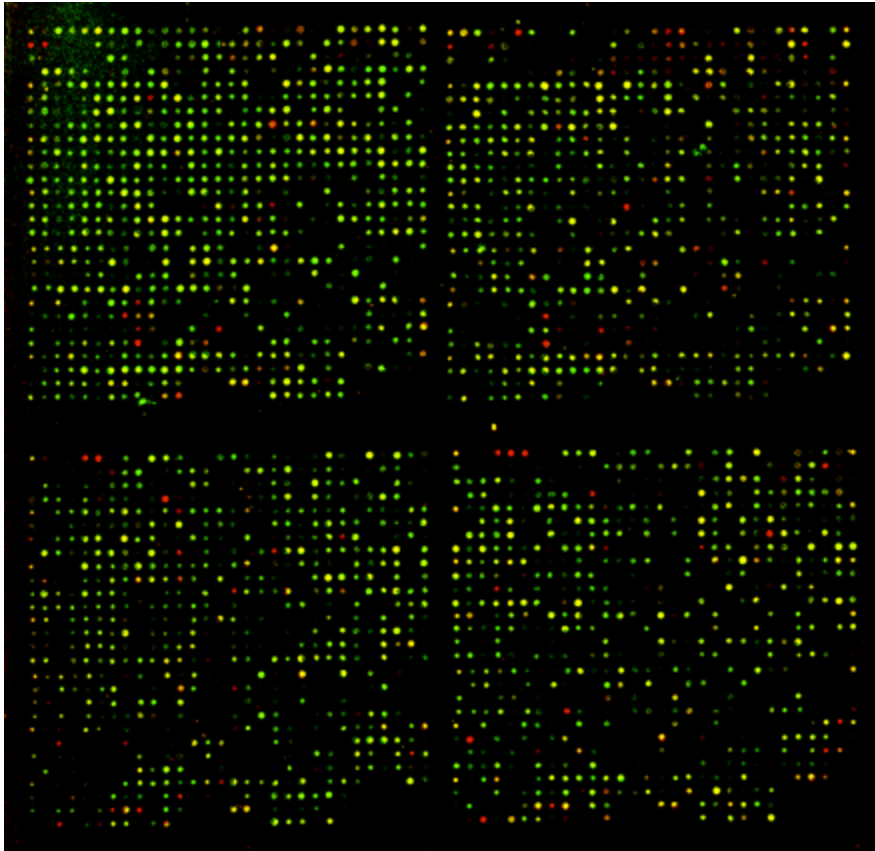


Prochlorococcus MED4

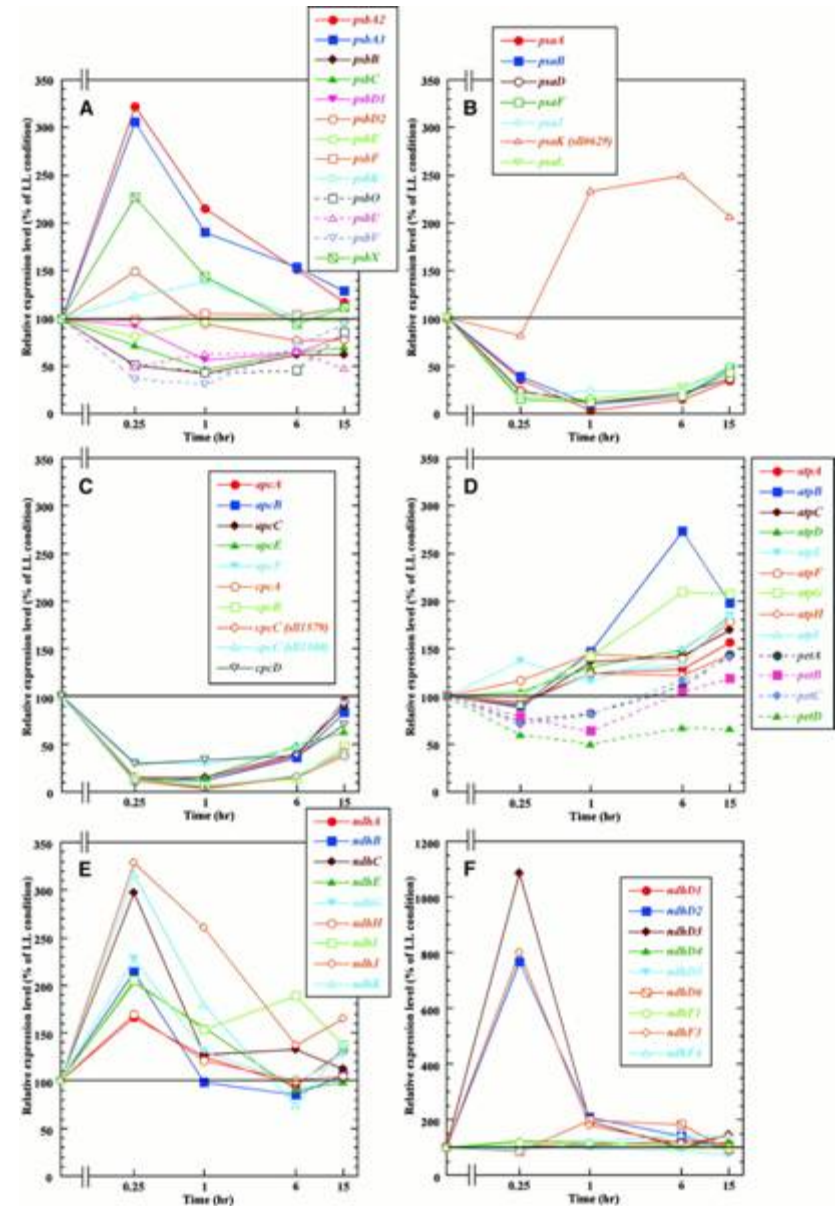


Prochlorococcus MIT9313

Hihara, Kamei, Kanehisa, Kaplan, and Ikeuchi (2001) DNA microarray analysis of cyanobacterial gene expression during acclimation to high light. Plant Cell, 13(4)



*Synechocystis* PCC 6803



# How do cells control response to light?

I.e., What genes are related to the adaptation to high light?

## Outline Protocol

Look for:

- Gene present in *Prochlorococcus* MED4  
MED4 is naturally adapted to grow in high light.
- Ortholog absent in *Prochlorococcus* MIT9313  
MIT9313 is naturally adapted to grow in low light
- Ortholog present in *Synechocystis* PCC 6803  
In order to make contact with annotation and microarray data
- *Synechocystis* PCC 6803 ortholog responds to high light  
Gene turns on by factor  $> 2$  in response to high light

The Prime Directive (first clause):

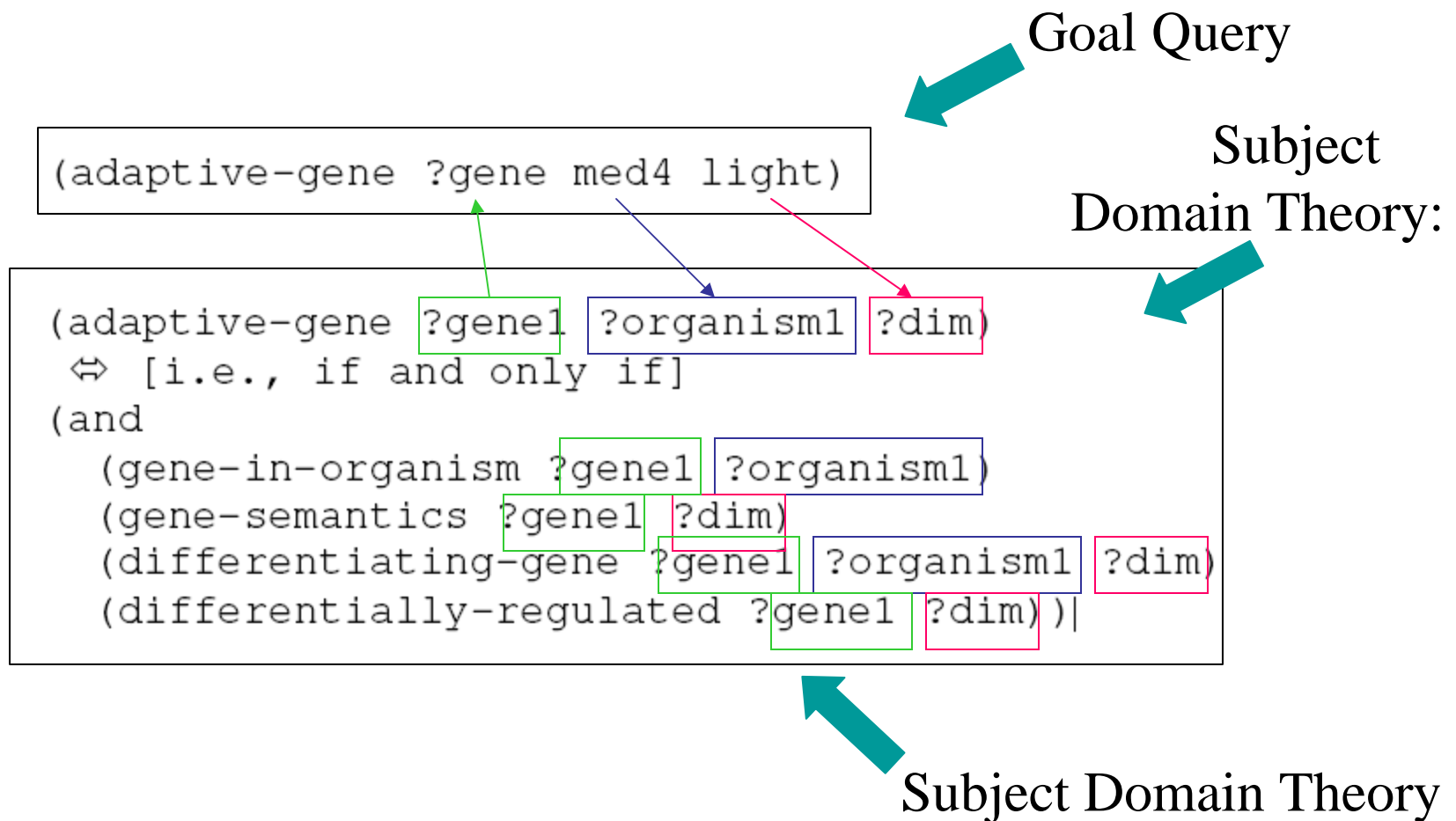
All data and knowledge can be manipulated by user-written programs that approximate user's natural protocols.

BioBike program:

```
(loop for pm4gene in (#^Genes ProcMed4)
  as all-orthologous = (all-blast-orthologs pm4gene)
  as 6803ortholog = (intersect (#^Genes Syny6803) all-orthologous)
  when (and (not-any #'member-geneid
                    (#^Genes slotv Proc9313) all-orthologous))
        (any #'member-geneID 6803ortholog)
        (>= ma-ratio (ma-select 6803ortholog Hihara1) 2)))
collect light-specific-genes 6803ortholog)
```

# Language for Expressing Conjectures, and Platform for Analysis

- A. First Order Logic (FOL) representation
- B. Subject Domain Theory
- C. Biological Process (and entities) Ontology
- D. Visual query language.



(differentially-regulated ?gene ?dim)

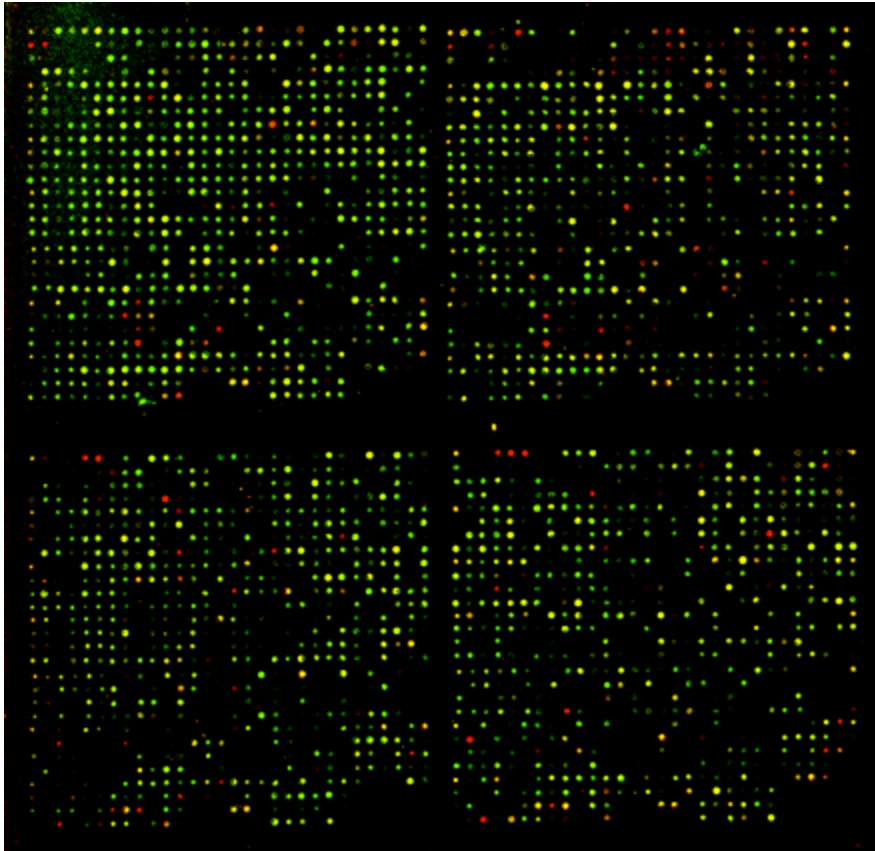
⇔

```
(exists
  ((?experiment :sort experiment)
   (?organism3 :sort species)
   (?gene3 :sort gene))
  (and
    (experiment ?experiment :dimension ?dim
                  :organism ?organism3)
    (gene-has-ortholog-in-organism ?gene
                                     ?gene3 ?organism3)
    (> (regulation-ratio ?gene3 ?experiment) 2.0)
  ))
```

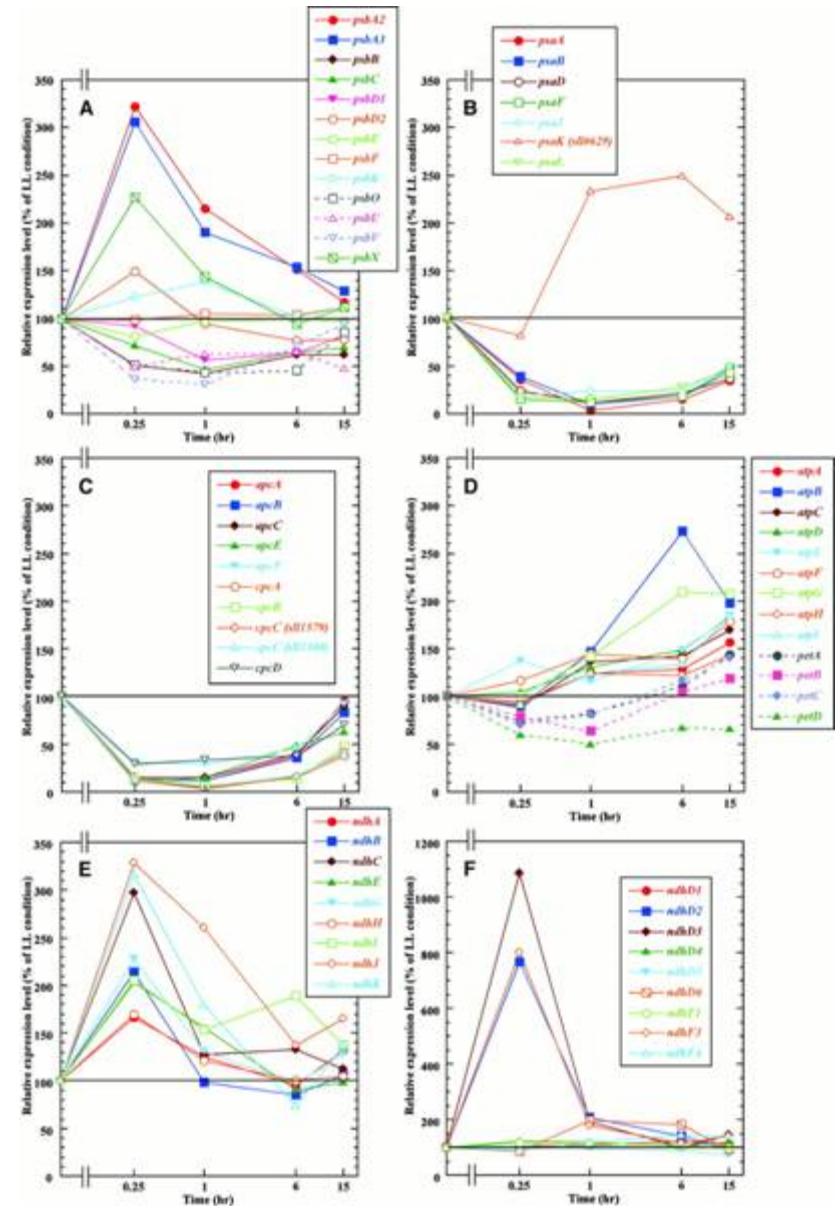
(experiment hihara :dimension light :organism s6803)



Hihara, Kamei, Kanehisa, Kaplan, and Ikeuchi (2001) DNA microarray analysis of cyanobacterial gene expression during acclimation to high light. Plant Cell, 13(4)



*Synechocystis* PCC 6803



Goal Query:

(adaptive-gene ?gene med4 light)

**Result:**

?gene: #*\$P*MED4.PMM0817

?organism2: #*\$p*rochlorococcus\_marinus\_mit9313

?experiment: HIHARA

?organism3: #*\$s*ynechocystis\_pcc6803

?gene3: #*\$S*6803.ssr2595

I.e., A low-light organism that has no ortholog to ?gene is *prochlorococcus marinus* pcc. 9313. Experiments were performed by Hihara on the organism *synechocystis* pcc 6803, and a high regulation ratio was discovered in those experiments on gene *S6803.ssr2595*, which is an ortholog of PMM0817. The annotation for PMM0817 reads: “possible high-light inducible protein”.

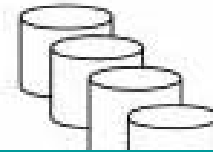
(Matches the results from: Bhaya, Dufresne, Vaultot, and Grossman: Analysis of the hli gene family in marine and freshwater cyanobacteria. FEMS Letters, 2002, 205(2). PMM0817 is called hli17 in this paper.)

# Natural Language Biocomputing

“Does pmm0226 not have an ortholog in mit9313?”

“What is the Hihara mean regulation ratio of pmm0226?”

Data Sources



*List the genes that pertain to med4 and that have an ortholog in s6803 that has a hihara ratio greater than 2 and that do not have orthologs in mit9313.*

*What genes confer differential adaptation to light in promed4 versus pro9313?*

Gemini  
parser

Controls

Biology  
Theory

Conjecture

SNARK  
theorem  
prover

Proof

Answer

Explanation

-->

# What Biologists Want To Compute

Knowledge Search: What do we know about this  
Gene, Pathway, System, Organism, ...?

Comparison: How are these organisms similar?

Model Identification: What models fit this data?

Model Analysis: What does this model predict?

Annotation: What does this gene do?

Use and Share Results.

Goal Query:

(adaptive-gen

Result:

?gene: # \$PMED4.PMM081

?organism2: # \$prochlo

?experiment: HIHARA

?organism3: # \$synecho

?gene3: # \$S6803.ssr25

+ "Explanation"

```
(resolve 5411 definition-of-differentiating-gene)
Answer (or (snark::answer-- (list |hashdollar-PMED4.PMM0817|))
  (defn-of-differentially-regulated
    |hashdollar-PMED4.PMM0817| light
    |hashdollar-S6803.ssr2595|
    |hashdollar-synechocystis_pcc6803|)))

(Row 5416
  (or (not (organism ?species :environment light :quality low))
    (not
      (gene-in-organism |hashdollar-PMED4.PMM0817|
        |hashdollar-prochlorococcus_marinus_med4|))
      (gene-has-ortholog-in-organism |hashdollar-PMED4.PMM0817|
        (snark-user::gene-skolemmqlk2 |hashdollar-PMED4.PMM0817| ?species)
        ?species))
    (rewrite (resolve 5413 defn-of-differentially-ecotyped) 8)
    Answer (or (snark::answer-- (list |hashdollar-PMED4.PMM0817|))
      (defn-of-differentially-regulated
        |hashdollar-PMED4.PMM0817| light
        |hashdollar-S6803.ssr2595|
        |hashdollar-synechocystis_pcc6803|)))

(Row 5422
  (or
    (not
      (gene-in-organism |hashdollar-PMED4.PMM0817|
        |hashdollar-prochlorococcus_marinus_med4|))
      (gene-has-ortholog-in-organism |hashdollar-PMED4.PMM0817|
        (snark-user::gene-skolemmqlk2 |hashdollar-PMED4.PMM0817|
          |hashdollar-prochlorococcus_marinus_mit9313|)
          |hashdollar-prochlorococcus_marinus_mit9313|))
      (resolve 5416 7)
      Answer (or (snark::answer-- (list |hashdollar-PMED4.PMM0817|))
        (defn-of-differentially-regulated
          |hashdollar-PMED4.PMM0817| light
          |hashdollar-S6803.ssr2595|
          |hashdollar-synechocystis_pcc6803|)))

(Row 5425
  (or
    (not
      (gene-in-organism-code |hashdollar-PMED4.PMM0817|
        |hashdollar-prochlorococcus_marinus_med4|))
      (gene-has-ortholog-in-organism |hashdollar-PMED4.PMM0817|
        (snark-user::gene-skolemmqlk2 |hashdollar-PMED4.PMM0817|
          |hashdollar-prochlorococcus_marinus_mit9313|)
          |hashdollar-prochlorococcus_marinus_mit9313|))
      (resolve 5422 definition-of-gene-in-organism-code)
      Answer (or (snark::answer-- (list |hashdollar-PMED4.PMM0817|))
        (defn-of-differentially-regulated
          |hashdollar-PMED4.PMM0817| light
          |hashdollar-S6803.ssr2595|
          |hashdollar-synechocystis_pcc6803|)))

(Row 5427
  (gene-has-ortholog-in-organism |hashdollar-PMED4.PMM0817|
    (snark-user::gene-skolemmqlk2 |hashdollar-PMED4.PMM0817|
```

Scientists can “promote” hypotheses as if they were results, and other scientists can import these. The system automatically tracks provenance (code+params, or BioDeducta “explanations”) to build a network of support.

user: Shrager:

Top Matrix	Weight:	3301930422: Foreign Terrorist	33
Operations:			
3301930131: Terrorists sniping	MEDIUM	CC	
3301917159: Accent sounded Caribbean, Jamaican?	MEDIUM	C	
3301917788: TV profile: white, male, military background	MEDIUM	I	
Support:		0.5	-1.

user: Heuer:

Top Matrix	Weight:	3301930556: Terrorists sniping	3301929911: Kids out joyriding with guns	Diagnosticity:
Operations:				
3301917397: Forensics shows all shot at long range with .223 bullets	MEDIUM	Delete Promote	II	1 confirm, and 1 disconfirm (0)
3301917333: 2 Oct shot fired thru Michael's store window	MEDIUM	I	CC	1 confirm, and 1 disconfirm (0)
Support:		0.5	0.0	

When the support for linked results changes, results that depend upon those are likewise changed in level of belief, or are flagged for reconsideration.

user: Shrager:

Top Matrix	Weight:	3301930422: Foreign Terrorist	3301930422: Foreign Terrorist
Operations:			
3301930131: Terrorists sniping	MEDIUM	CC	
3301917159: Accent sounded Caribbean, Jamaican?	MEDIUM	C	
3301917788: TV profile: white, male, military background	MEDIUM	I	
Support:		0.5	-1

user: Heuer:

Top Matrix	Weight:	3301930556: Terrorists sniping	3301929911: Kids out joyriding with guns	Diagnosticity:
Operations:				
3301917397: Forensics shows all shot at long range with .223 bullets	MEDIUM	Delete Promote	II	1 confirm, and 1 disconfirm (0)
3301917333: 2 Oct shot fired thru Michael's store window	MEDIUM	I	CC	1 confirm, and 1 disconfirm (0)
Support:		0.5	0.0	

# “Bayes Community”

Client/server architecture permits collaboration among scientists through “publication” of hypothesis and linking in as evidence

New Results

Incoming theories are distributed to the scientists according to the hypotheses they are working, and heuristically knitted into the ongoing model development process

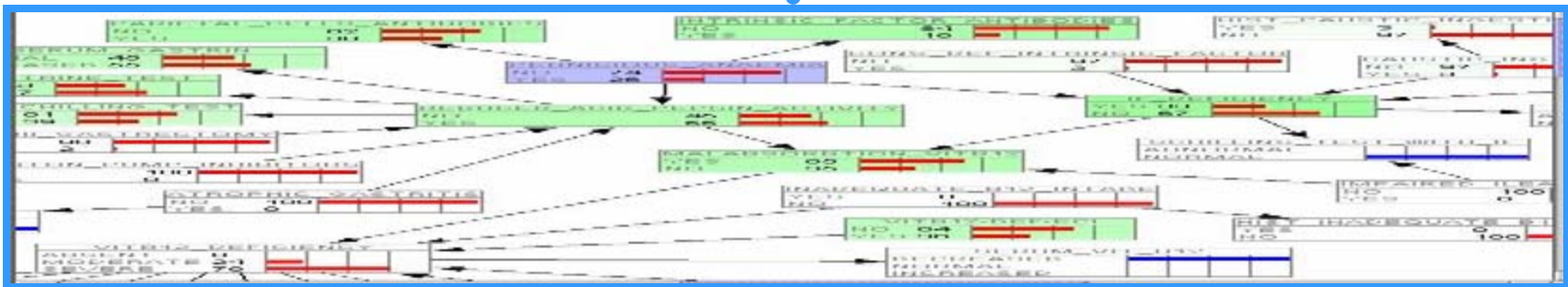
Analysis  
of  
Competing  
Hypotheses

Analysis  
of  
Competing  
Hypotheses

Analysis  
of  
Competing  
Hypotheses

Analysis  
of  
Competing  
Hypotheses

Linked matrices project a Bayesian influence network





Goal: Enable biologists to carry out novel, complex, knowledge-based computations themselves.

Approach: Make complex biocomputing easier for biologists.

How?

- Provide a uniform programming interface...

- Using a simple, interactive, paradigm...

- To a programmable data/knowledge base...

- With all the relevant data/knowledge preloaded...

- And all the relevant tools (incl. advanced ones)...

- And intelligent semi-automatic programming tools...

- In a biological community...

- All through the web.

Goal: Enable biologists to carry out novel, complex, knowledge-based computations themselves.

How: KnowOS/BioBike/BioDeducta/CACHE

Who?

Primarily: JP Massar, Mike Travers, Jeff Elhai,  
Richard Waldinger, & John K. Myers

With: Bob Haxo, Andrew Pohorille, Mark Slupesky, Mark Stickel,  
The labs of Devaki Bhaya, Pat Langley & Arthur Grossman,  
Numerous students.

Funding: NASA, NSF, ONR, ARDA, & sweat equity

Goal: Enable biologists to carry out novel, complex, knowledge-based computations themselves.

How: KnowOS/BioBike/BioDeducta/CACHE

Where?

www.biobike.org, [www.knowos.org](http://www.knowos.org), biolinguas.sourceforge.net  
Massar, et al. (2005) Bioinformatics, 21(20)

Also:

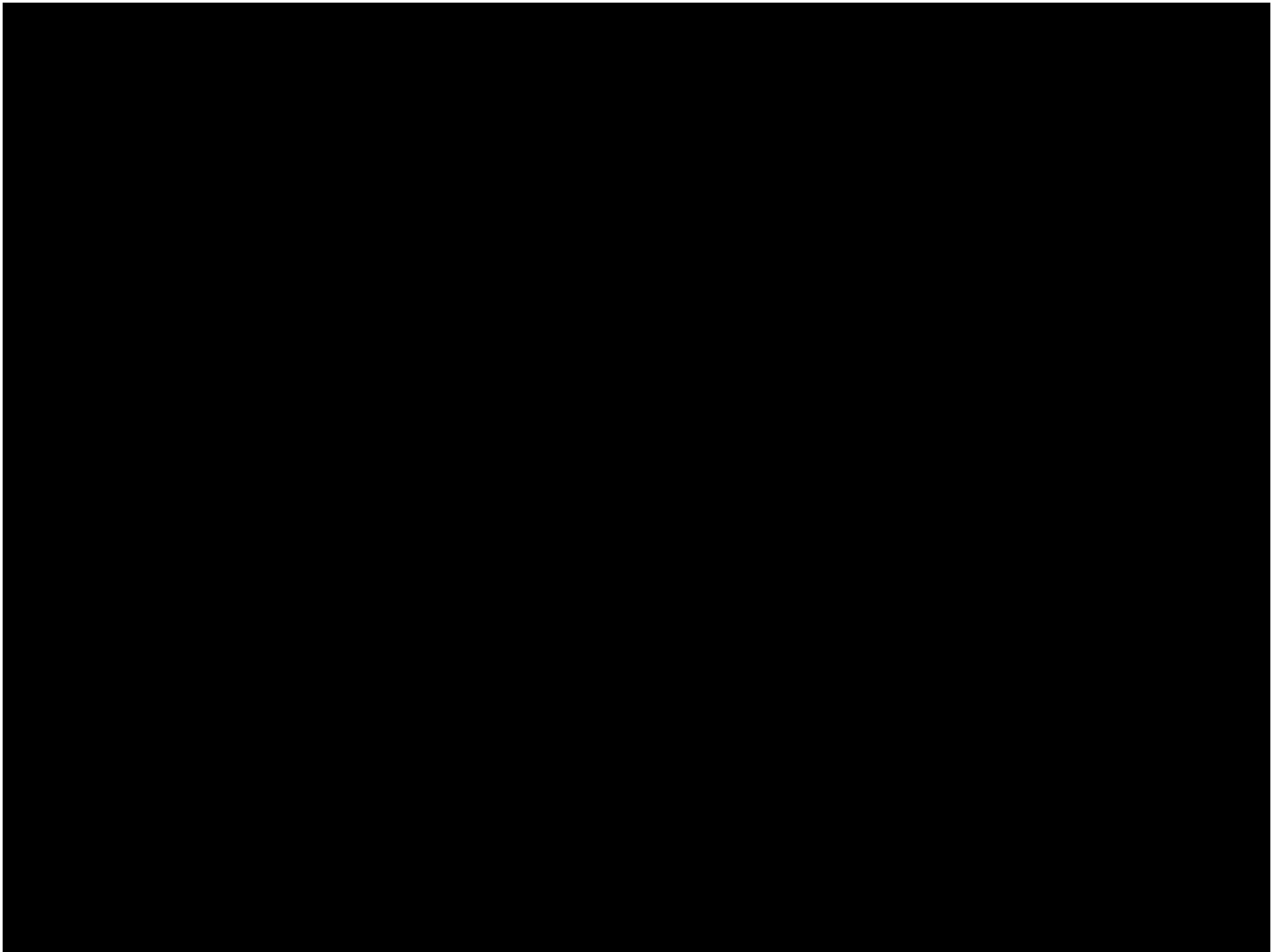
Fedoroff, et al. (2006) The Scientist, 19(11)

Travers, et al. (2005) ILC (KnowOS)

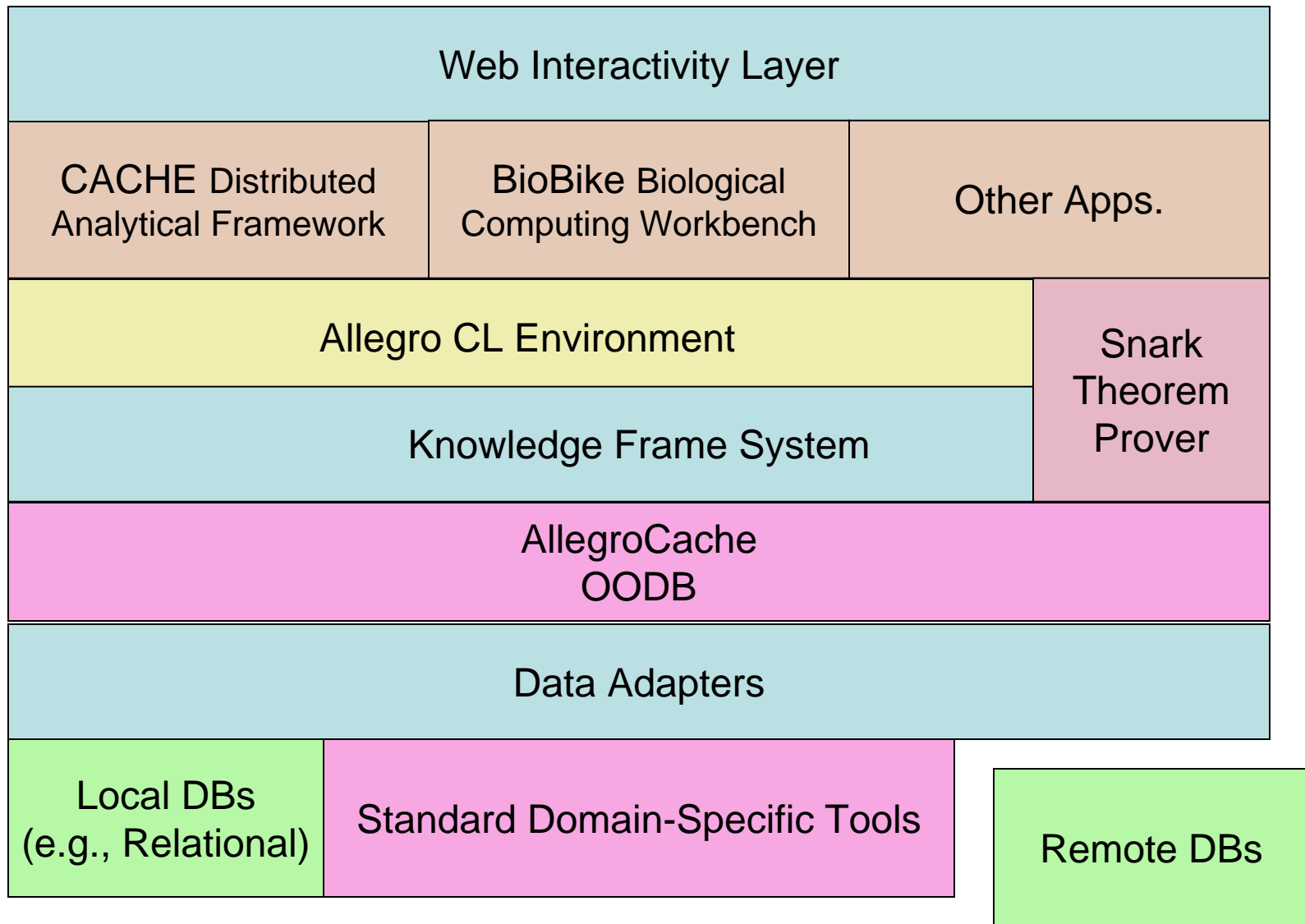
Shrager, et al. (2002) PSB

Shrager, (in press) J. Hist. and Phil. of Science

Shrager, et al. (in press) PLoSOne (BioDeducta)



# KnowOS Computational Architecture



# BioBike v. Bio{ PERL|Python|Java|... }

- Local Knowledge Warehousing (in memory or local databases).
- Knowledge Integrated through a Frame System.
- Intended to be programmed by biologists themselves.
- Tools and commonly used KBs are locally pre-integrated.
- Complete Web-based programming environment.
- Data accessed remotely or in random local files.
- Data integration usually haphazard (per application).
- Intended to be programmed by engineers.
- Tools and required KBs must be hand installed.
- Piecemeal local programming tools.

# BioBike v. BioPerl (in The SEED)

## SEED Version:

```
for i in `pegs $1`  
do  
    (echo "$i"; echo "$i" | similar_to 1.0e-50 | is_prokaryotic | head -n  
40 )  
| upstream upstream=100 plus=10 | tr -d A-Z >  
"Output-intergenic.$1/$i.fasta"  
    cd Output-intergenic.$1; clustalw -infile=$i.fasta -align > /dev/null  
    cd ..  
    echo $i  
done
```

Challenge problem:

Find 100 bp of sequence upstream from a set of orthologs for all genes in an organism and align them.

*(by Rick Stevens, co-developer of The Seed)*

---

## BioLingua-Lite version:

```
(FOR-EACH gene IN (GENES-OF Npun)  
  AS orthologs = (ORTHOLOGS-OF gene)  
  AS upstream-seqs = (SEQUENCES-UPSTREAM-OF orthologs LENGTH 100)  
  COLLECT (ALIGNMENT-OF upstream-seqs))
```

*(by Jeff Elhai, developer of BioLite)*