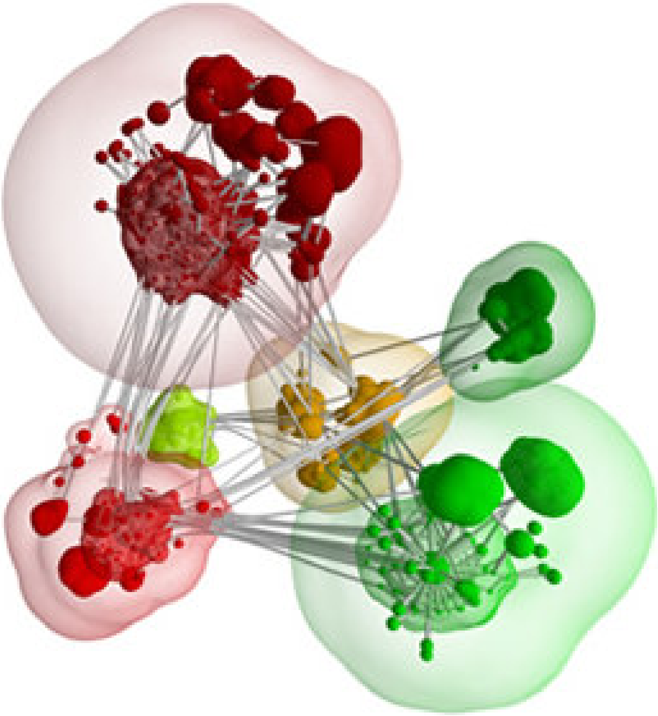
Large Life Science Datasets with SPARQL or Prolog



Jans Aasman, Ph.D.
CEO Franz Inc
ja@franz.com

You work in the life sciences:

How do you use AllegroGraph?

This talk

- AllegroGraph in a few slides
- Using the Science Commons datasets
- Gruff: A rich client for data exploration, prolog and sparql
- AGWebview: a webbrowser
- Some observations of Science Commons dataset users
- Prolog or SPARQL

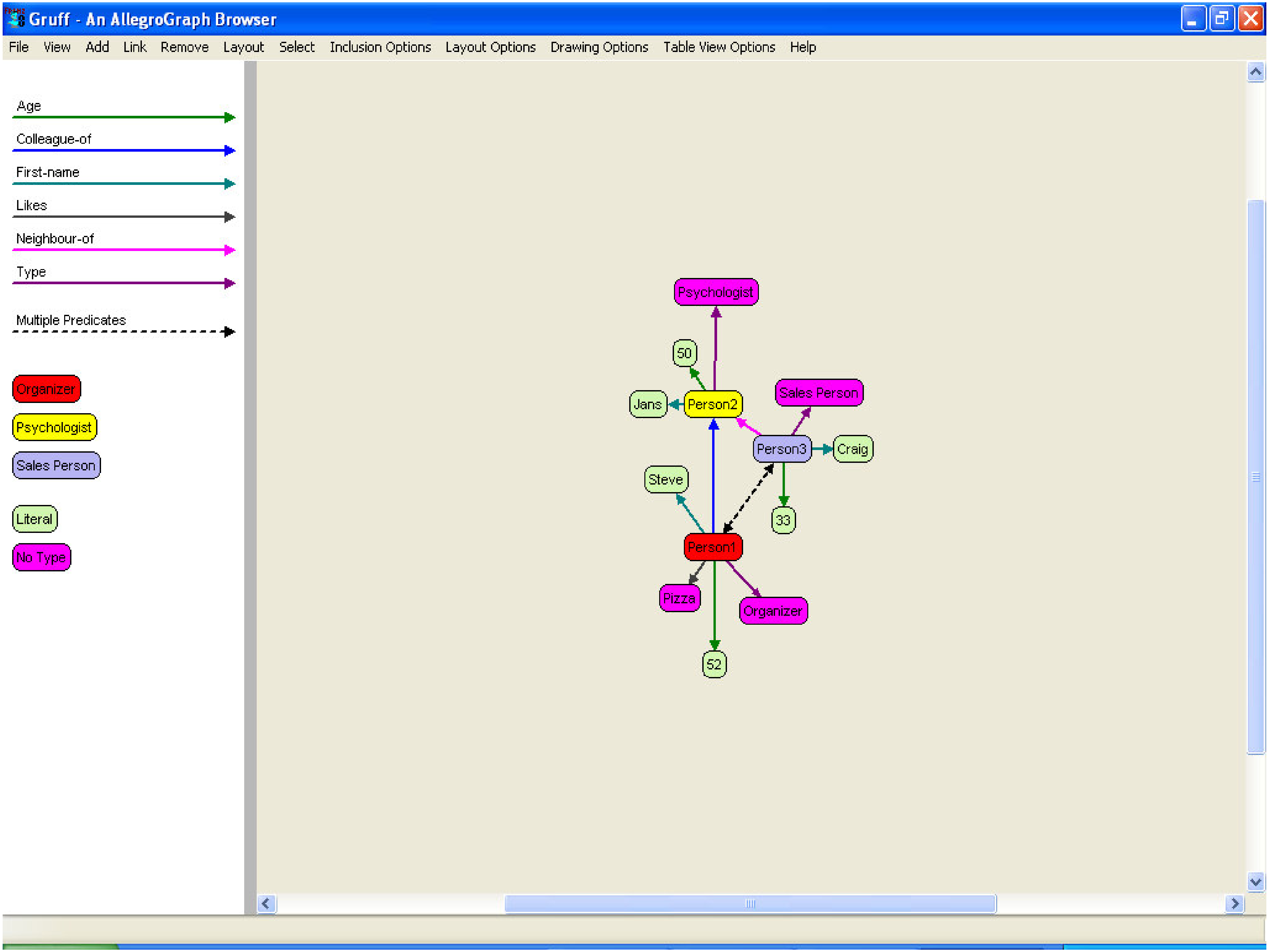
Graphs, triples, triple-store?

```
createTripleStore(“seminar.db" )

addTriple (Person1 first-name Steve)
addTriple (Person1 isa  Organizer)
addTriple (Person1 age 52)
addTriple (Person2 first-name Jans)
addTriple (Person2 isa Psychologist)
addTriple (Person2 age 50)
addTriple (Person3 first-name Craig)
addTriple (Person3 isa SalesPerson)
addTriple (Person3 age 32)

addTriple (Person1 colleague-of Person2)
addTriple (Person1 colleague-of Person3)
addTriple ( Person3 neighbor-of Person1)
addTriple ( Person3 neighbor-of Person2)

addTriple (Person1 likes Pizza)
```


Gruff - An AllegroGraph Browser
File
View
Add
Link
Remove
Layout
Select
Inclusion Options
Layout Options
Drawing Options
Table View Options
Help
Age
Colleague-of
First-name
Likes
Neighbour-of
Type
Multiple Predicates
Organizer
Psychologist
Sales Person
Literal
No Type
Psychologist
50
Jans
Person2
Sales Person
Person3
Craig
Steve
33
Person1
Pizza
Organizer
52

And now you can query in Prolog or Sparql

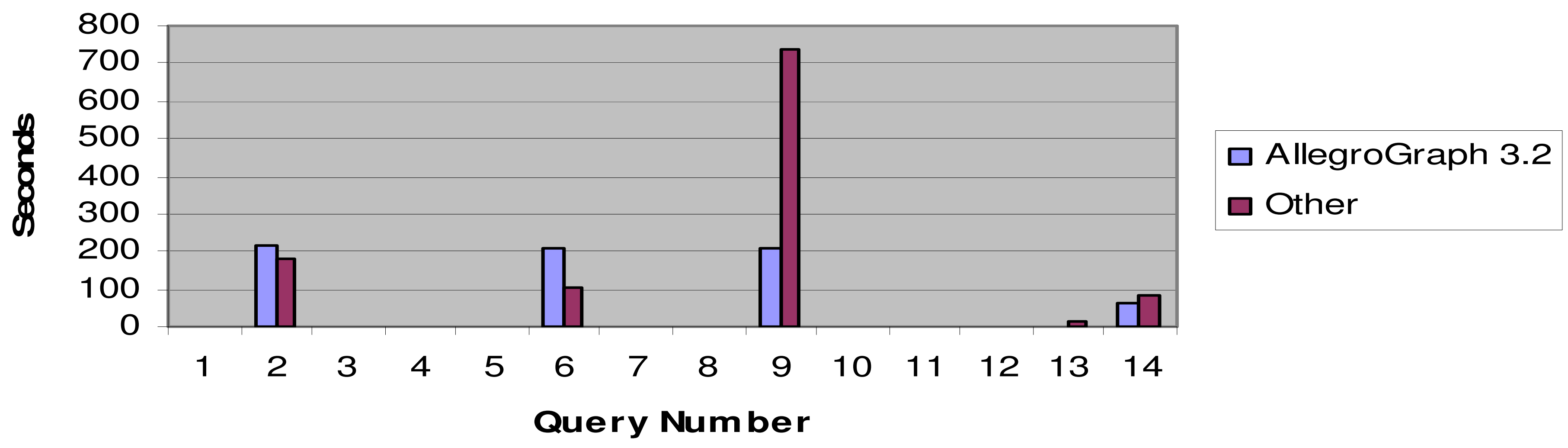
```
(select (?xname ?yname)
  (q ?x colleague-of ?y)
  (q ?y neighbor-of ?x)
  (q ?x first-name ?xname)
  (q ?y first-name ?yname))
```

```
SELECT ?xname ?yname WHERE {
    ?x ex:colleague-of ?y .
    ?y ex:neighbor-of ?x .
    ?x ex:first-name ?xname .
    ?y ex:first-name ?yname . }
```

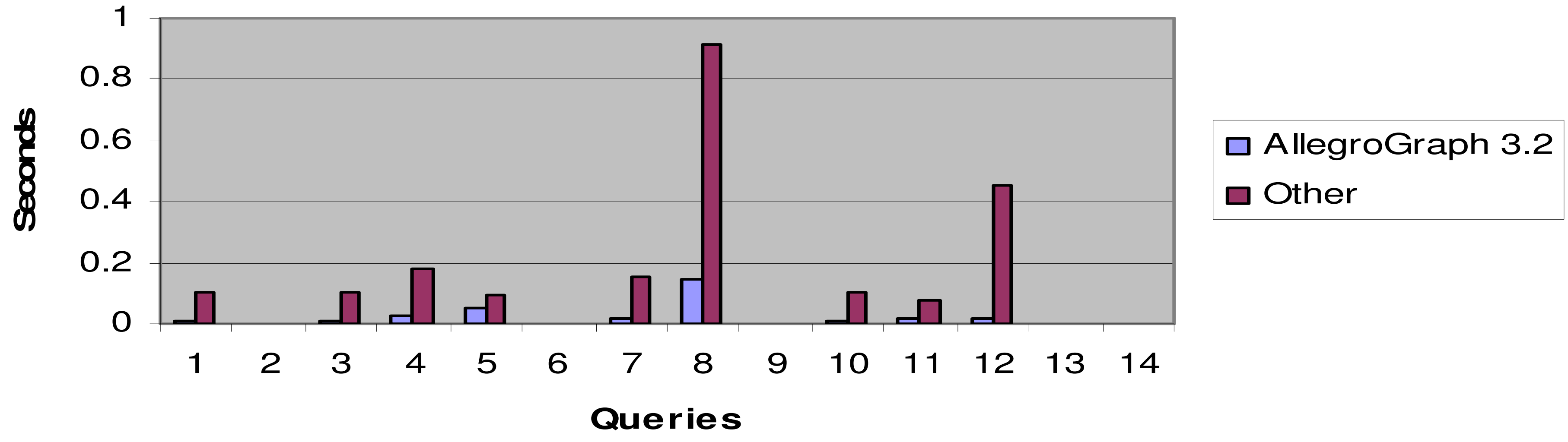
AllegroGraph

- Scalable and persistent Triple (Quad) Store
  - Load and query over Billions of RDF triples
  - The only fast reasoner that doesn't need materializing
- Compliant with standards
  - RDF, RDFS, OWL, SPARQL, Named Graphs, ISO Prolog, OWL-lite reasoning
- RDFS++ Reasoning
  - All of RDFS + owl:sameAs, owl:transitiveProperty, owl:inverseOf, owl:hasValue
- Full text indexing
- Spatial, Temporal and Social

LUBM(8000) queries



LUBM(8000) with long queries zeroed



AllegroGraph

- Scalable and persistent Triple (Quad) Store
  - Load and query over Billions of RDF triples
  - The only fast reasoner that doesn’t need materializing
- Compliant with standards
  - RDF, RDFS, OWL, SPARQL, Named Graphs, ISO Prolog, OWL-lite reasoning
- RDFS++ Reasoning
  - All of RDFS + owl:sameAs, owl:transitiveProperty, owl:inverseOf, owl:hasValue
- Full text indexing
- Spatial, Temporal and Social

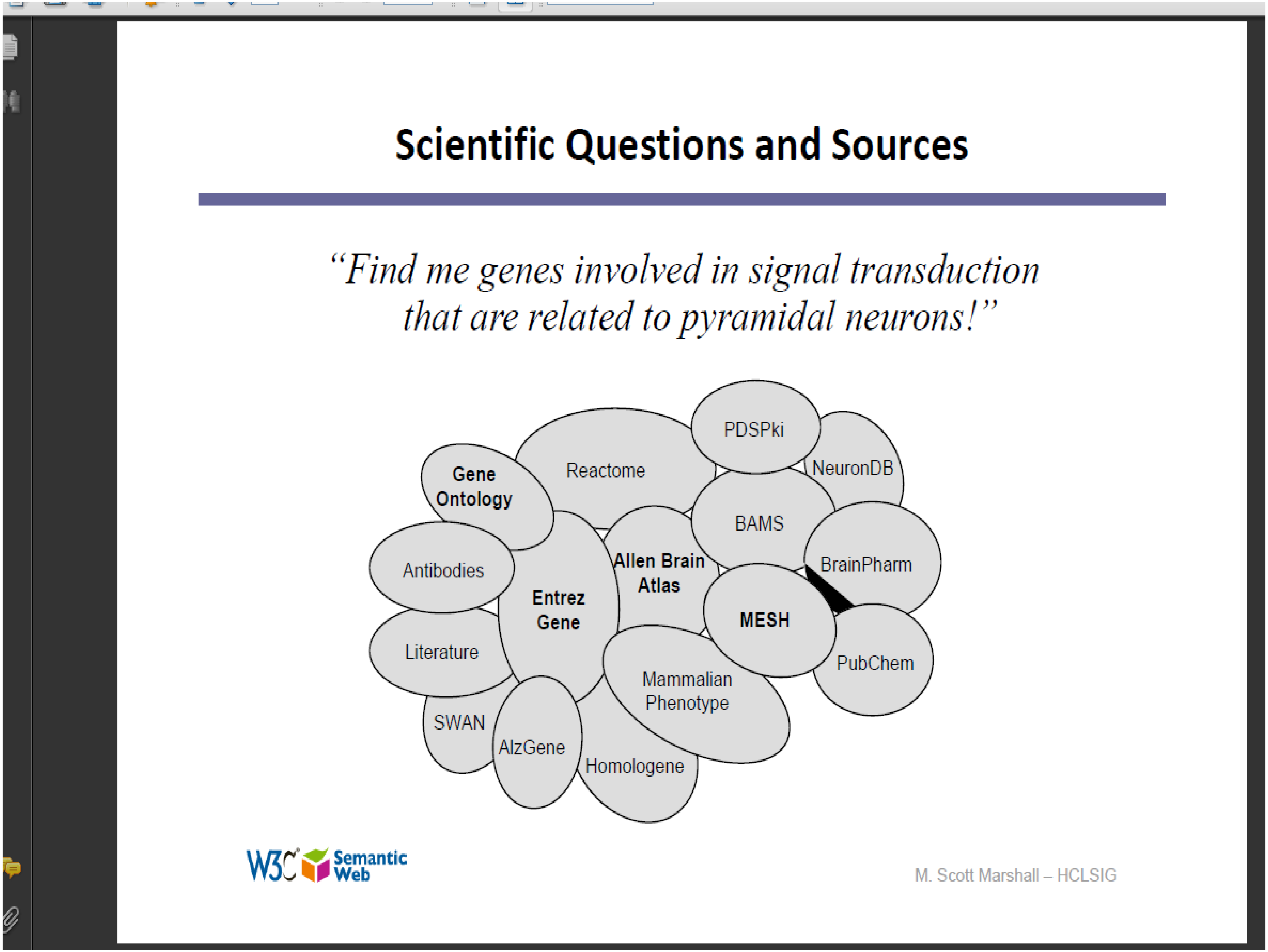
Harnessing the Semantic Web to Answer Scientific Questions:

A Health Care and Life Sciences Interest Group demo

Alan Ruttenberg, Science Commons

science commons

Accelerating the
Scientific Research Cycle



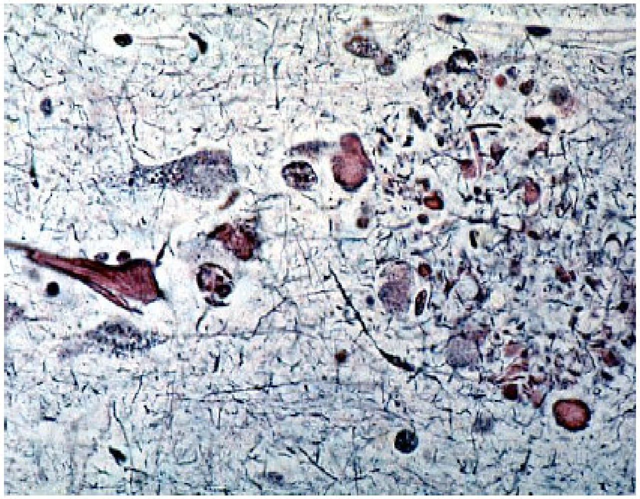
Scientific Questions and Sources

"Find me genes involved in signal transduction that are related to pyramidal neurons!"

M. Scott Marshall – HCLSIG

Looking for Alzheimer Disease targets

Signal transduction pathways are considered to be rich in “druggable” targets - proteins that might respond to chemical therapy



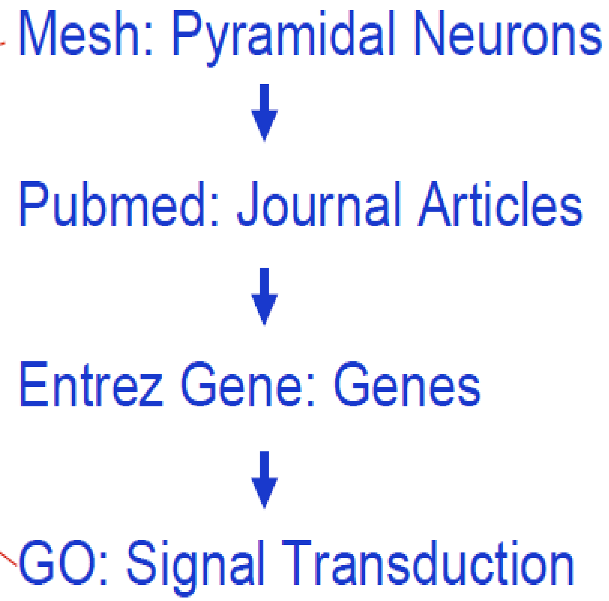
CA1 Pyramidal Neurons are known to be particularly damaged in Alzheimer’s disease.

Casting a wide net, can we find candidate genes known to be involved in signal transduction and active in Pyramidal Neurons?

A SPARQL query spanning 4 sources

```
prefix go: <http://purl.org/obo/owl/GO#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix owl: <http://www.w3.org/2002/07/owl#>
prefix mesh: <http://purl.org/commons/record/mesh/>
prefix sc: <http://purl.org/science/owl/sciencecommons/>
prefix ro: <http://www.obofoundry.org/ro/ro.owl#>

select ?genename ?processname
where
{  graph <http://purl.org/commons/hcls/pubmesh>
    { ?paper ?p mesh:D017966 .
      ?article sc:identified_by_pmid ?paper.
      ?gene sc:describes_gene_or_gene_product_mentioned_by ?article.
    }
  graph <http://purl.org/commons/hcls/goa>
    { ?protein rdfs:subClassOf ?res.
      ?res owl:onProperty ro:has_function.
      ?res owl:someValuesFrom ?res2.
      ?res2 owl:onProperty ro:realized_as.
      ?res2 owl:someValuesFrom ?process.
  graph <http://purl.org/commons/hcls/20070416/classrelations>
    {{?process <http://purl.org/obo/owl/obo#part_of> go:GO_0007166}
      union
     {?process rdfs:subClassOf go:GO_0007166 }}
      ?protein rdfs:subClassOf ?parent.
      ?parent owl:equivalentClass ?res3.
      ?res3 owl:hasValue ?gene.
     }
  graph <http://purl.org/commons/hcls/gene>
    { ?gene rdfs:label ?genename }
  graph <http://purl.org/commons/hcls/20070416>
    { ?process rdfs:label ?processname}
}
```


Mesh: Pyramidal Neurons

Pubmed: Journal Articles

Entrez Gene: Genes

GO: Signal Transduction

Inference required

AllegroGraph and NC dataset

- Loading 100,000,000 triples, including text indexing for rdf:comment and rdfs:label

– Loading	1:30:23
– Indexing:	15:19
– Total time:	1:45:43

Demo

- Gruff and NC
- AGWebview and NC

Some Observations

Issue [1] - Graphs

In which Graph(s) are my triples?

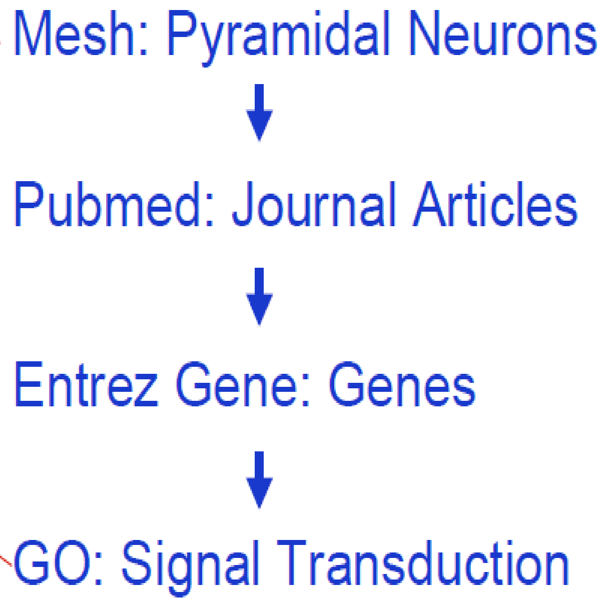
- Researchers are forced to partition the data through graphs (the fourth argument of a triple) at load time
- Researchers are forced to remember which graph knows about what predicates (or risk severe performance penalties)

- AllegroGraph supports federation: you can partition your data through graphs in one db, or you can have your data in different dbs on different machines…

A SPARQL query spanning 4 sources

```
prefix go: <http://purl.org/obo/owl/GO#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix owl: <http://www.w3.org/2002/07/owl#>
prefix mesh: <http://purl.org/commons/record/mesh/>
prefix sc: <http://purl.org/science/owl/sciencecommons/>
prefix ro: <http://www.obofoundry.org/ro/ro.owl#>

select ?genename ?processname
where
{  graph <http://purl.org/commons/hcls/pubmesh>
    { ?paper ?p mesh:D017966 .
      ?article sc:identified_by_pmid ?paper.
      ?gene sc:describes_gene_or_gene_product_mentioned_by ?article.
    }
  graph <http://purl.org/commons/hcls/goa>
    { ?protein rdfs:subClassOf ?res.
      ?res owl:onProperty ro:has_function.
      ?res owl:someValuesFrom ?res2.
      ?res2 owl:onProperty ro:realized_as.
      ?res2 owl:someValuesFrom ?process.
  graph <http://purl.org/commons/hcls/20070416/classrelations>
    {{?process <http://purl.org/obo/owl/obo#part_of> go:GO_0007166}
      union
     {?process rdfs:subClassOf go:GO_0007166 }}
      ?protein rdfs:subClassOf ?parent.
      ?parent owl:equivalentClass ?res3.
      ?res3 owl:hasValue ?gene.
    }
  graph <http://purl.org/commons/hcls/gene>
    { ?gene rdfs:label ?genename }
  graph <http://purl.org/commons/hcls/20070416>
    { ?process rdfs:label ?processname}
}
```



Mesh: Pyramidal Neurons

Pubmed: Journal Articles

Entrez Gene: Genes

GO: Signal Transduction

Inference required

Issue [1] - Graphs

In which Graph(s) are my triples?

- Researchers are forced to partition the data through graphs (the fourth argument of a triple) at load time
- Researchers are forced to remember which graph knows about what predicates (or risk severe performance penalties)

- AllegroGraph supports <u>federation</u>: you can partition your data through graphs in one db, or you can have your data in different dbs on different machines…

Issue [2] – Materializing is pain

An amazing 3.4 M subclass relationships, sometimes to 10 levels deep,

  - Reasoning without materialization is painfully slow
  - But: Materializing takes hours
  - Multiplies the number of triples
  - Any serious change to the ontology forces re-materialize

- AllegroGraph we do not need to materialize
- We optimize Prolog queries
  - Statistics based
  - Predicates are indexed on the fly
  - Industry Leading LUBM results *without* materializing

FRANZ INC.
"WEB 3.0's DATABASE"

Issue [3] - Numbers

Range queries on numbers and dates is slow if data doesn't fit in memory

- Find every subject S for measurement M where the certainty values are between 0.7 and 0.9
- Millions of numbers in NeuroCommons datasets
- In lab data more numbers than symbols

- In AllegroGraph numbers are *not* in string table but natively encoded. We support nearly all XML Schema data types.

Issue [4] - Abstractions

- Interesting SPARQL Queries are usually far too long because SPARQL doesn’t support Abstractions

- AllegroGraph supports full Prolog and Prolog functors
- Franz is considering Common Logic as a more user friendly, and more declarative way to do queries and rules

SPARQL or Prolog

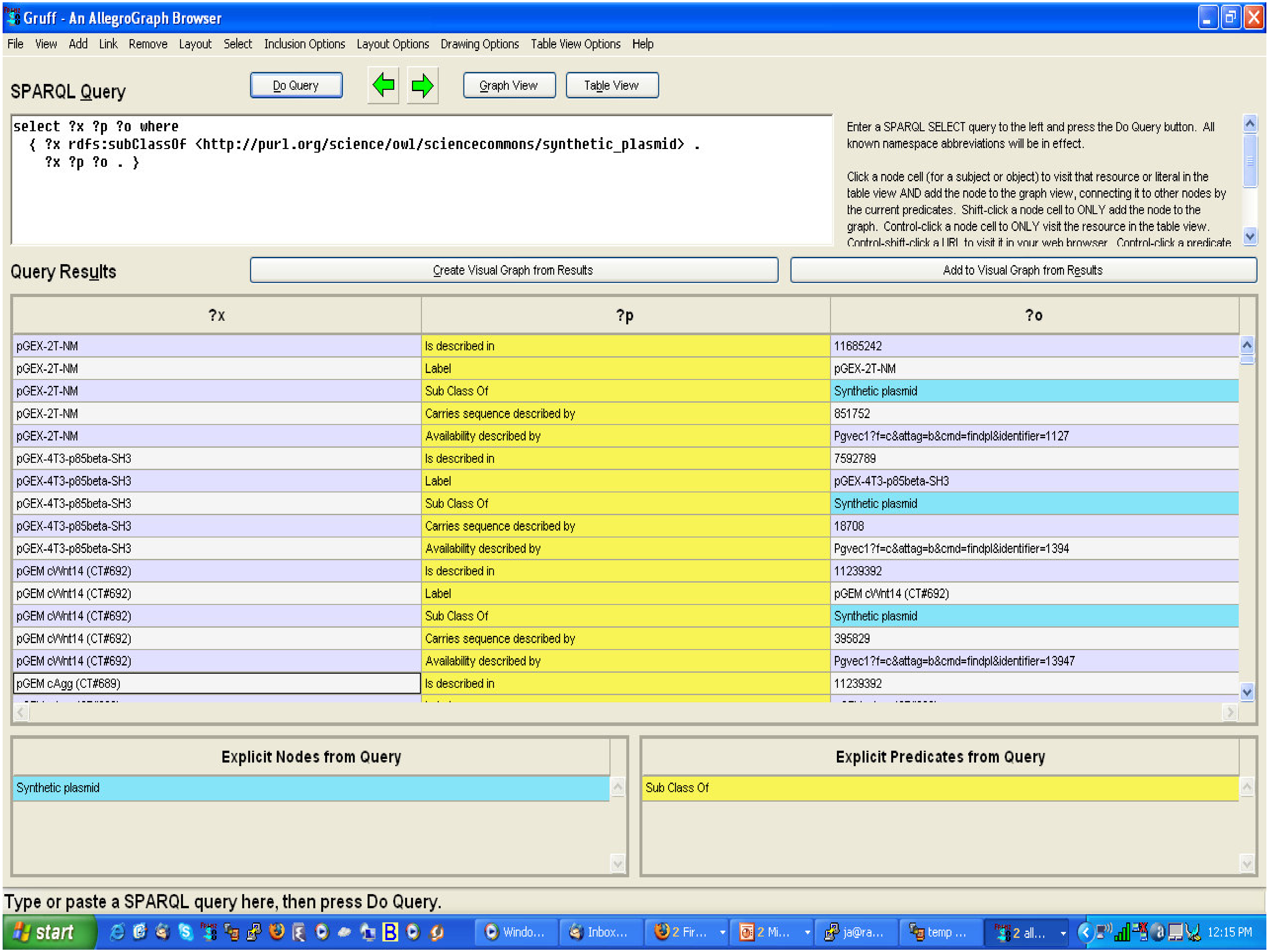
- 70 % of our users use SPARQL only
  - It is the standard QL, good descriptions on the web, quickly growing community that can help.., many SPARQL end points
- 30 % use Prolog
  - Not limited to two arguments
  - Range queries are naturally encoded
  - Use rules and build layer of abstractions
  - Has already query optimizer
    - Statistics based, indices on the fly
    - No need for static materializing
    - Reasoner integrated
  - Will be important in the future if rule-ML or Common Logic take off

Thank You

Jans Aasman

Franz Inc.

www.franz.com



```
select ?x ?p ?o where
  { ?x rdfs:subClassOf <http://purl.org/science/owl/sciencecommons/synthetic_plasmid> .
    ?x ?p ?o . }
```

Enter a SPARQL SELECT query to the left and press the Do Query button. All known namespace abbreviations will be in effect.

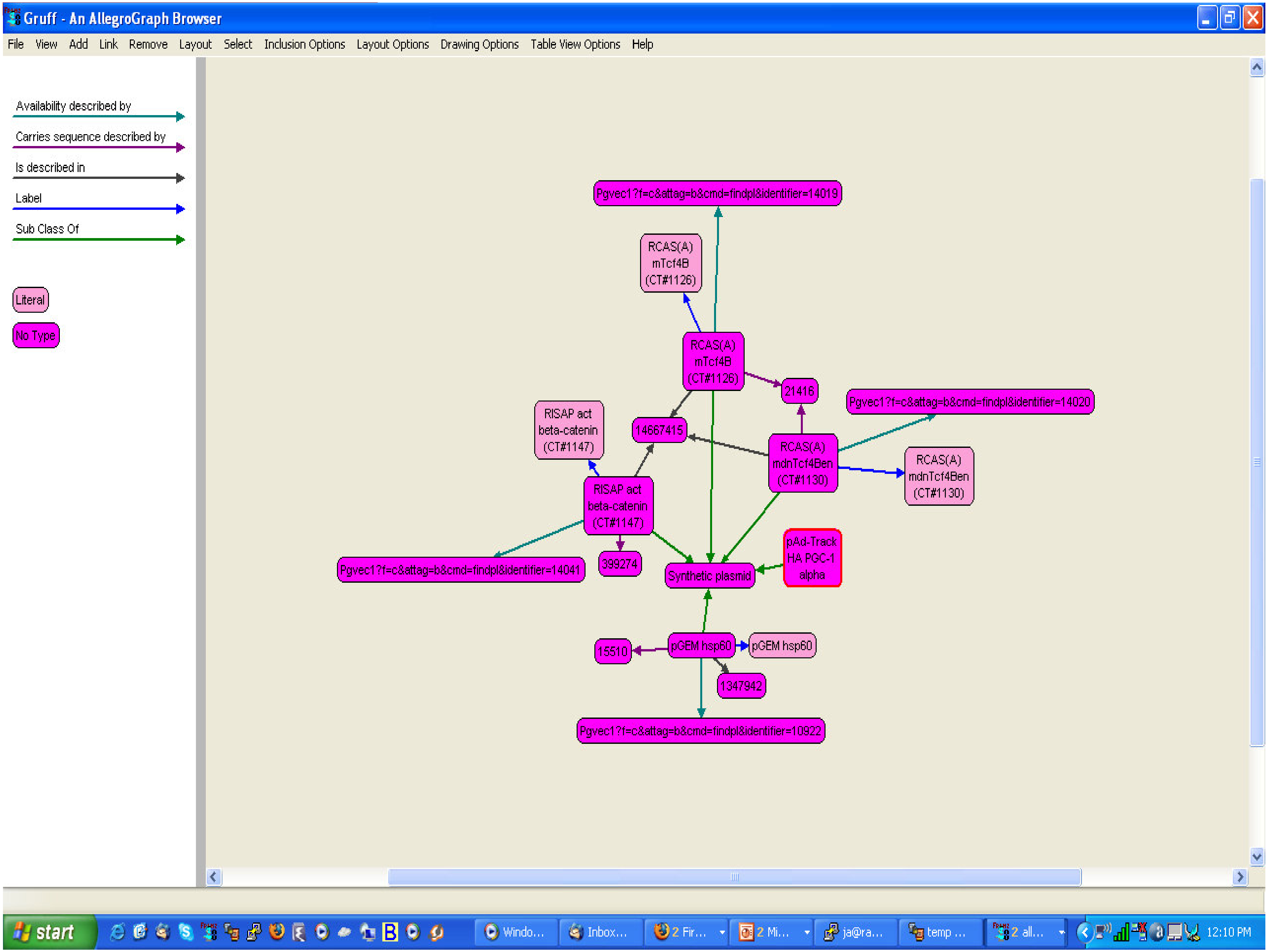
Click a node cell (for a subject or object) to visit that resource or literal in the table view AND add the node to the graph view, connecting it to other nodes by the current predicates. Shift-click a node cell to ONLY add the node to the graph. Control-click a node cell to ONLY visit the resource in the table view. Control-shift-click a URL to visit it in your web browser. Control-click a predicate

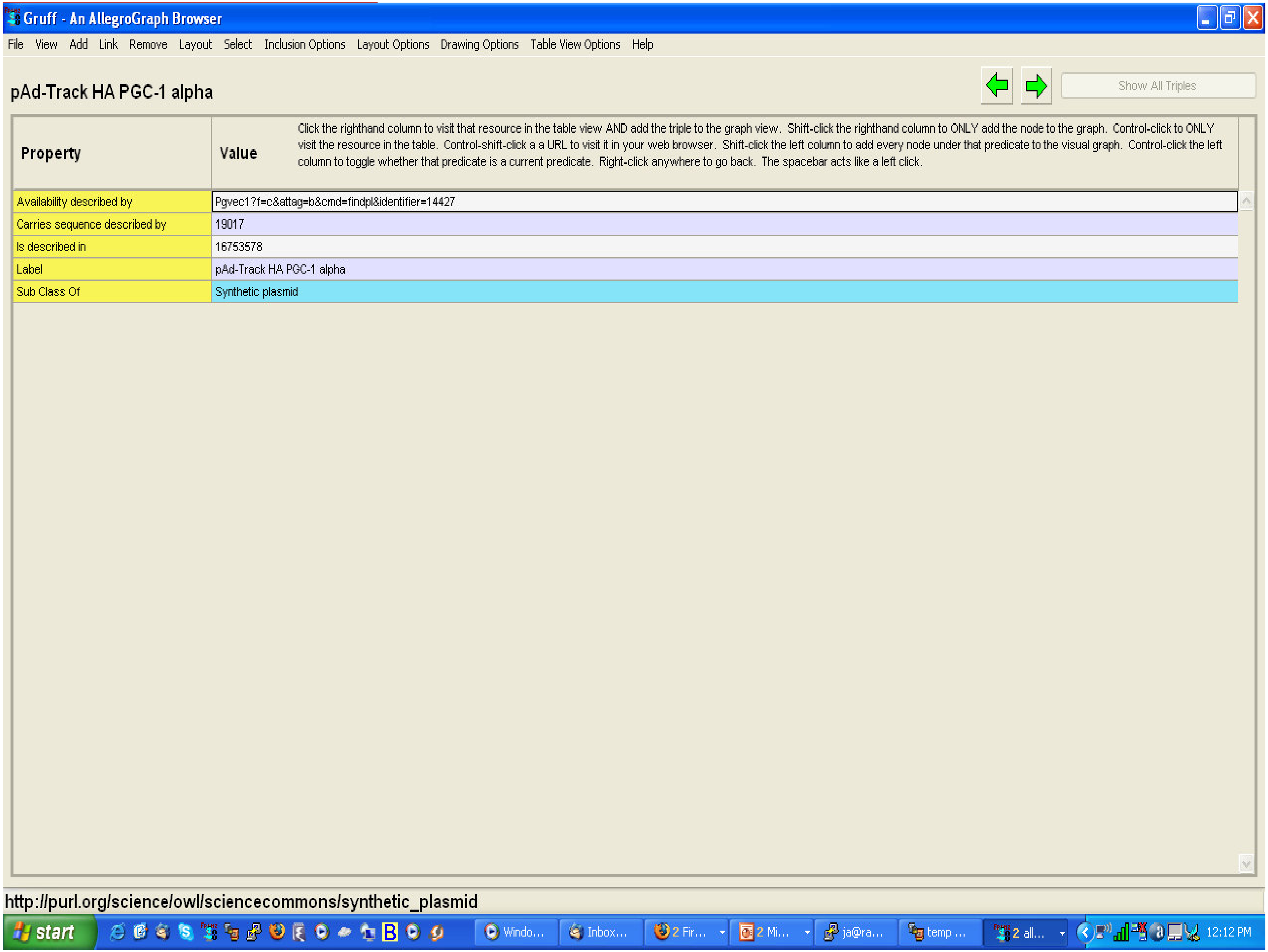
?x	?p	?o
pGEX-2T-NM	Is described in	11685242
pGEX-2T-NM	Label	pGEX-2T-NM
pGEX-2T-NM	Sub Class Of	Synthetic plasmid
pGEX-2T-NM	Carries sequence described by	851752
pGEX-2T-NM	Availability described by	Pgvec1?f=c&attag=b&cmd=findpl&identifier=1127
pGEX-4T3-p85beta-SH3	Is described in	7592789
pGEX-4T3-p85beta-SH3	Label	pGEX-4T3-p85beta-SH3
pGEX-4T3-p85beta-SH3	Sub Class Of	Synthetic plasmid
pGEX-4T3-p85beta-SH3	Carries sequence described by	18708
pGEX-4T3-p85beta-SH3	Availability described by	Pgvec1?f=c&attag=b&cmd=findpl&identifier=1394
pGEM cWnt14 (CT#692)	Is described in	11239392
pGEM cWnt14 (CT#692)	Label	pGEM cWnt14 (CT#692)
pGEM cWnt14 (CT#692)	Sub Class Of	Synthetic plasmid
pGEM cWnt14 (CT#692)	Carries sequence described by	395829
pGEM cWnt14 (CT#692)	Availability described by	Pgvec1?f=c&attag=b&cmd=findpl&identifier=13947
pGEM cAgg (CT#689)	Is described in	11239392

Explicit Nodes from Query
Synthetic plasmid

Explicit Predicates from Query
Sub Class Of

Type or paste a SPARQL query here, then press Do Query.


Gruff - An AllegroGraph Browser
File View Add Link Remove Layout Select Inclusion Options Layout Options Drawing Options Table View Options Help
Availability described by
Carries sequence described by
Is described in
Label
Sub Class Of
Literal
No Type
Pgvec1?f=c&attag=b&cmd=findpl&identifier=14019
RCAS(A) mTcf4B (CT#1126)
RCAS(A) mTcf4B (CT#1126)
21416
Pgvec1?f=c&attag=b&cmd=findpl&identifier=14020
RISAP act beta-catenin (CT#1147)
14667415
RCAS(A) mdnTcf4Ben (CT#1130)
RCAS(A) mdnTcf4Ben (CT#1130)
RISAP act beta-catenin (CT#1147)
Pgvec1?f=c&attag=b&cmd=findpl&identifier=14041
399274
Synthetic plasmid
pAd-Track HA PGC-1 alpha
15510
pGEM hsp60
pGEM hsp60
1347942
Pgvec1?f=c&attag=b&cmd=findpl&identifier=10922
start
Windo...
Inbox...
2 Fir...
2 Mi...
ja@ra...
temp ...
2 all...
12:10 PM

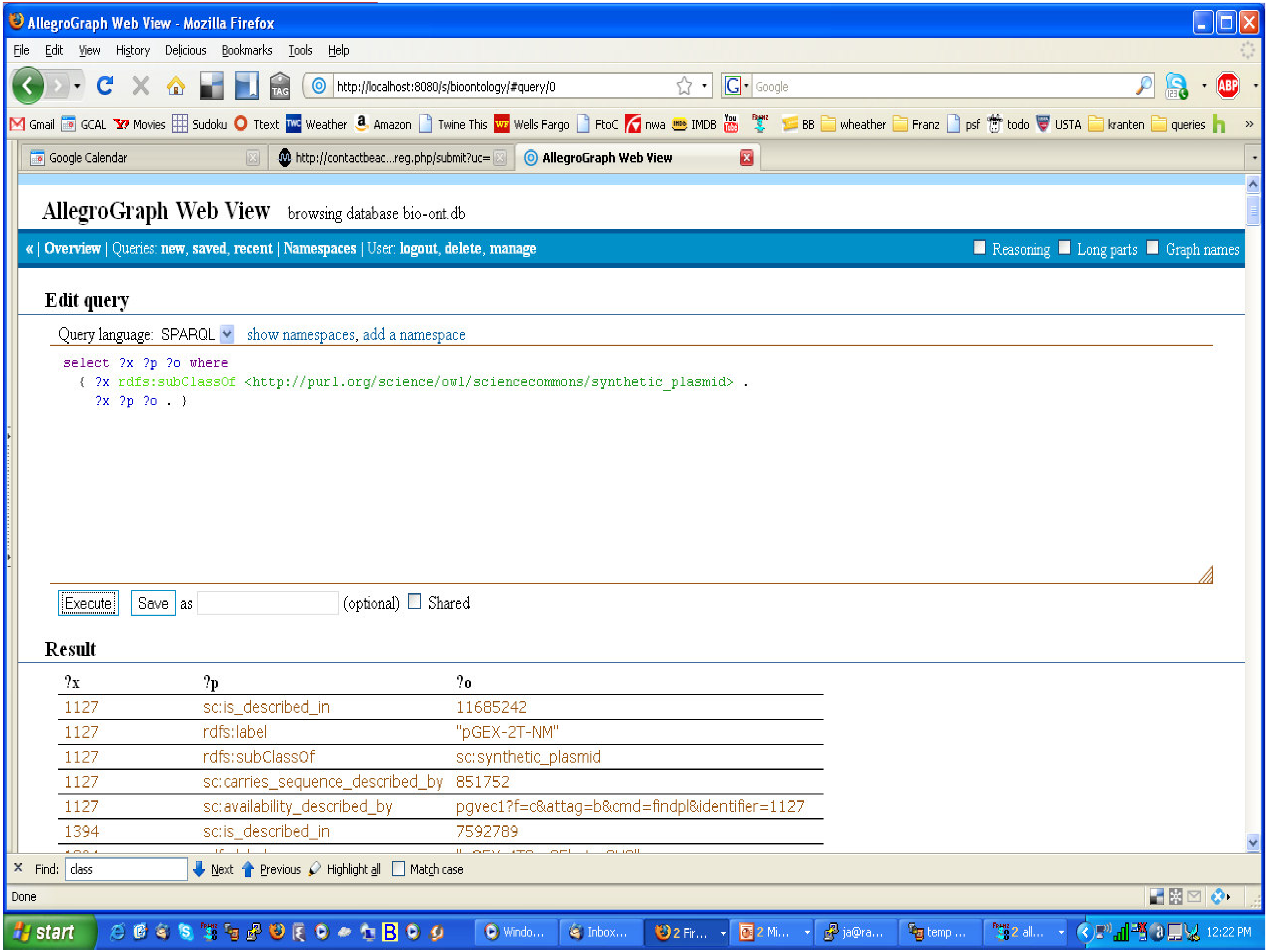


pAd-Track HA PGC-1 alpha

Click the righthand column to visit that resource in the table view AND add the triple to the graph view. Shift-click the righthand column to ONLY add the node to the graph. Control-click to ONLY visit the resource in the table. Control-shift-click a a URL to visit it in your web browser. Shift-click the left column to add every node under that predicate to the visual graph. Control-click the left column to toggle whether that predicate is a current predicate. Right-click anywhere to go back. The spacebar acts like a left click.

Property	Value
Availability described by	Pgvec1?f=c&attag=b&cmd=findpl&identifier=14427
Carries sequence described by	19017
Is described in	16753578
Label	pAd-Track HA PGC-1 alpha
Sub Class Of	Synthetic plasmid

http://purl.org/science/owl/sciencecommons/synthetic_plasmid


AllegroGraph Web View - Mozilla Firefox
File Edit View History Delicious Bookmarks Tools Help
http://localhost:8080/s/bioontology/#query/0
Google
Gmail GCAL Movies Sudoku Ttext Weather Amazon Twine This Wells Fargo FtoC nwa IMDB BB wheather Franz psf todo USTA kranten queries
Google Calendar
http://contactbeac...reg.php/submit?uc=
AllegroGraph Web View
« | Overview | Queries: new, saved, recent | Namespaces | User: logout, delete, manage
□ Reasoning □ Long parts □ Graph names
Find: class Next Previous Highlight all □ Match case
Done
start
Windo... Inbox... 2 Fir... 2 Mi... ja@ra... temp ... 2 all...
12:22 PM

AllegroGraph Web View browsing database bio-ont.db

Edit query

Query language: SPARQL show namespaces, add a namespace

```
select ?x ?p ?o where
  { ?x rdfs:subClassOf <http://purl.org/science/owl/sciencecommons/synthetic_plasmid> .
    ?x ?p ?o . }
```

Execute Save as (optional) □ Shared

Result

?x	?p	?o
1127	sc:is_described_in	11685242
1127	rdfs:label	"pGEX-2T-NM"
1127	rdfs:subClassOf	sc:synthetic_plasmid
1127	sc:carries_sequence_described_by	851752
1127	sc:availability_described_by	pgvec1?f=c&attag=b&cmd=findpl&identifier=1127
1394	sc:is_described_in	7592789