a programmatic interface for querying pathogen genomics data

Giles Velarde, Pathogen Genomics

VAPORWARE ALERT!

- Rapid prototypes
- Proofs of concept
- Requirements gathering

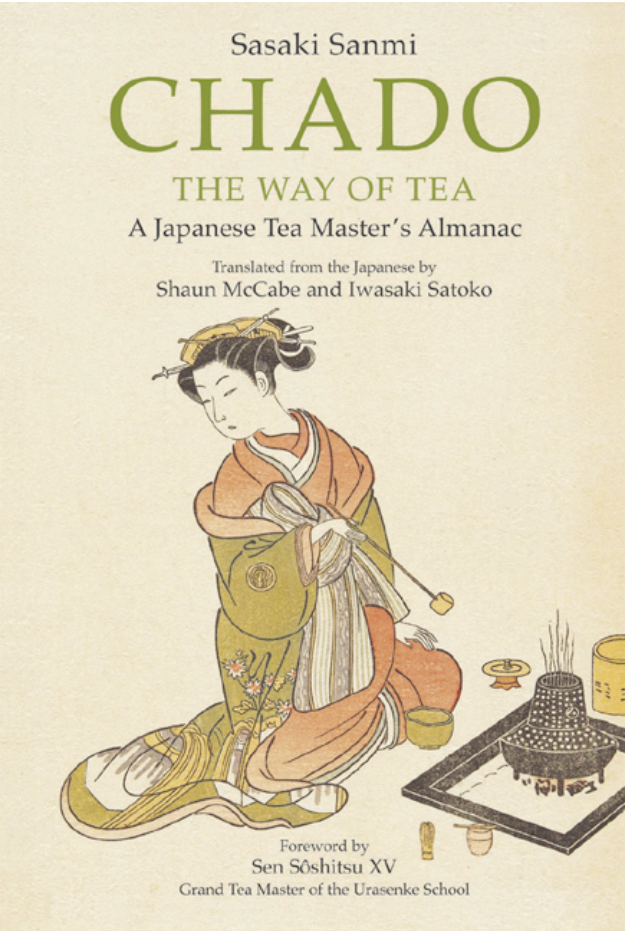
- Conditions apply

THE PATH

- Chado
- GeneDB
- Cooperation with EupathDB
- Web services & APIs
- The Way of the CRAWL

Hiroshige's Upright Tōkaidō

CHADO



- Chado is a relational database schema that underlies many GMOD installations. It is **capable of representing many of the general classes** of data frequently encountered in modern biology such as ***sequence, sequence comparisons, phenotypes, genotypes, ontologies, publications, and phylogeny***. It has been designed to handle **complex representations of biological knowledge** and should be considered one of the most sophisticated relational schemas currently available in molecular biology. The **price of this capability** is that the **new user must spend some time becoming familiar** with its fundamentals.

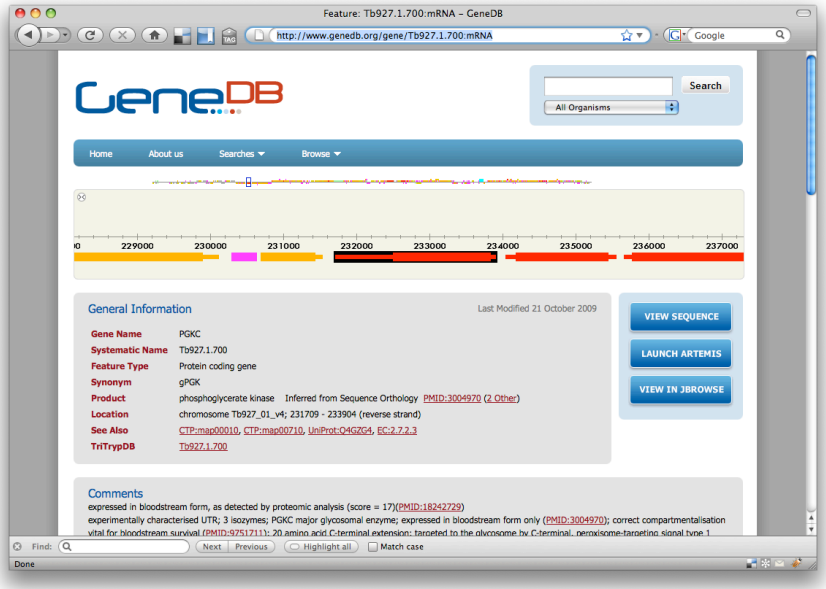
  - A database for very deep curation
  - An integrated database
  - A database that is generic enough to use for any organism

PATHOGEN DB - UNIQUE RESOURCE!

- Built on Chado
- 45 organisms
  - Apicomplexan Protozoa
  - Kinetoplastid Protozoa
  - Parasitic Helminths
  - Bacteria
- Cross-organism computed data
  - Orthologues
  - domains

GeneDB

- Pathogen DB web front-end
  - Hibernate
    - DAO caches
    - Lucene index search
- Weekly data updates



EUPATHDB COOPERATION

- EupathDB
  - Functional genomics integrative resources
- Collaboration
  - Annotation team
    - Sanger
    - Seattle SBRI
    - UGA
  - TriTrypDB / PlasmoDB
    - Data integration
      - GeneDB genomics data
      - Functional genomics data sets

THE CHALLENGE

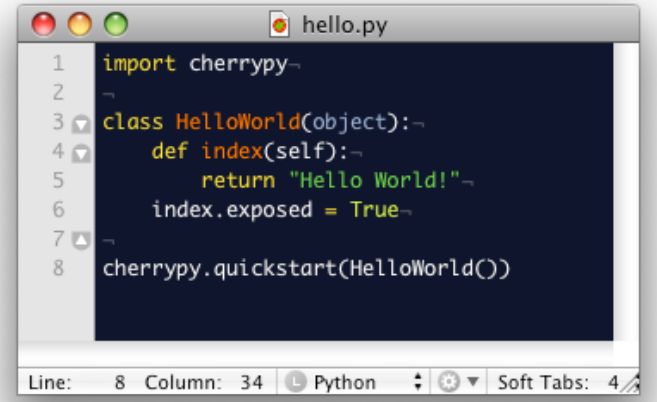
- EupathDB
  - Need to know what has changed
  - Need to be able to get the data
- Remote annotation
    - DB-Artemis via VPN
    - Consistency
  - Need to build Rich Internet Applications
- Sanger
  - Need to exploit our own data as well!
  - Chado-complexity - SQL hard

THE NEEDS

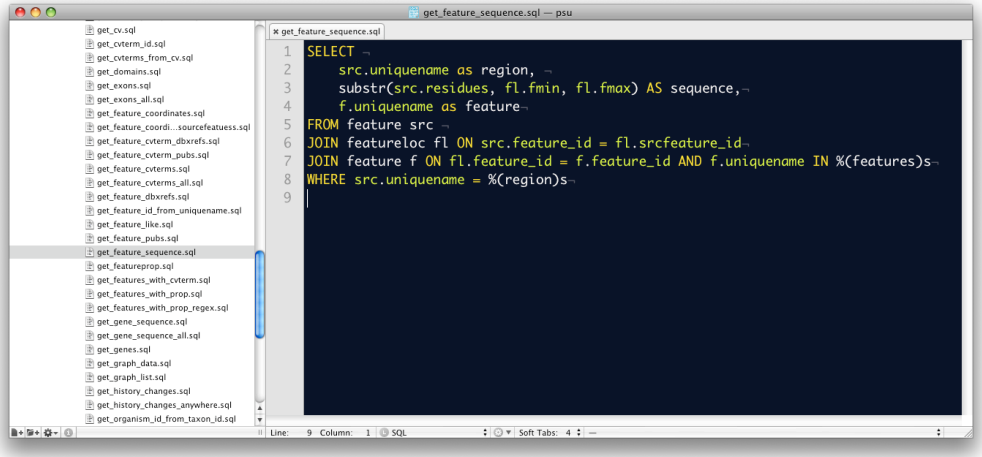
Hiroshige – Bowl of Sushi

- Rapid prototype
  - Quick to implement new queries
  - Must run directly off DB
    - No time to rebuild caches
    - Pure SQL
- Lightweight
  - Must not tax existing website
    - Bioinformatics PhD students…
  - JS,CURL/WGET,PERL,R…
    - REST
- Respond to user's specific needs

THE TECHNOLOGY



- Python, Jython
- CherryPy
  - Multi-threaded web app server
  - Simple to reuse controller classes in different contexts
- Ropy
  - SQL files
  - XML/JSON



THE FORM

- Library
  - Unit testing…
- Standalone app server
  - MVC … there is no V
  - enforced decoupling of the data layer from the view
- Command line app
  - Direct DB access
    - WS not suitable for LSF jobs
  - PERL wrapper module

THE PURPOSE

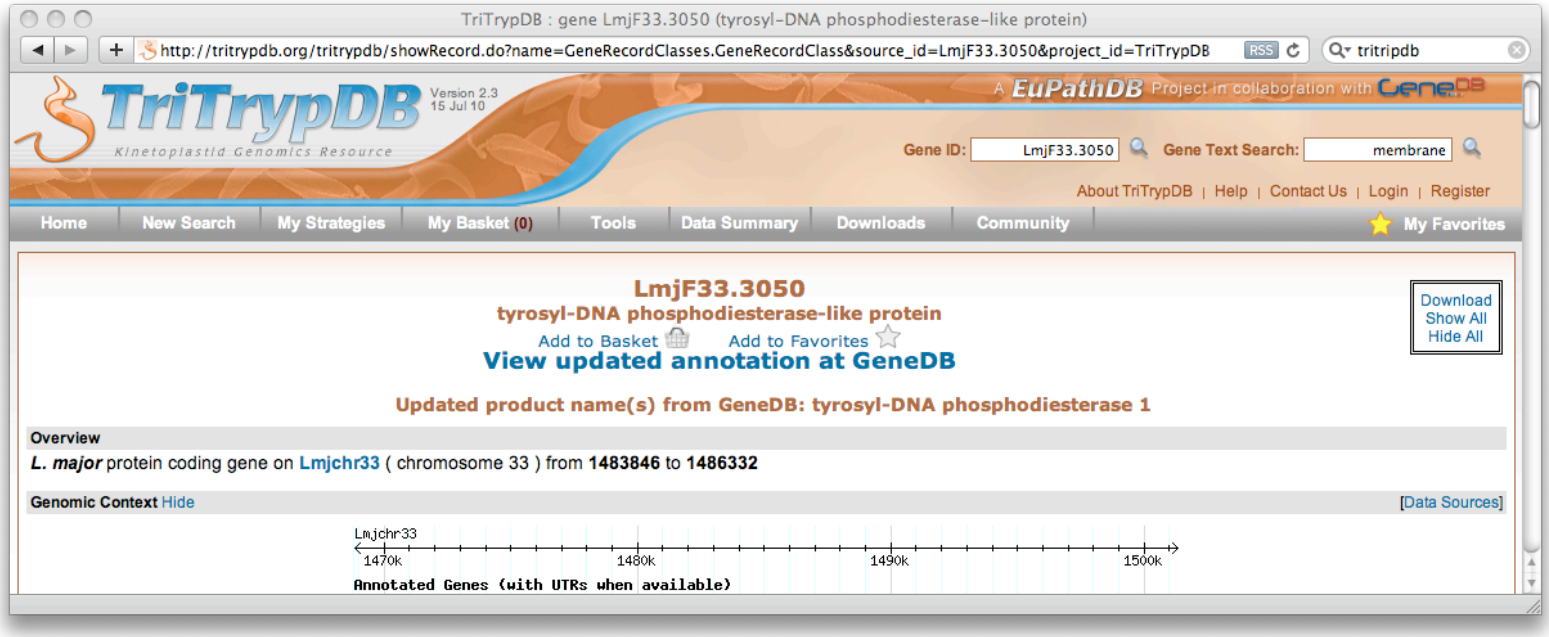
Mask the **complexity** of the SQL as much as possible,

& allow you to get on with **data analysis & development.**

USE CASES

WHAT'S NEW WEB-SERVICE

- EupathDB
  - Queries our services daily
    - recent annotation changes
  - Displays on their gene page a link to GeneDB



EXPLOITING PATHOGENS DB LOCALLY

Using the PERL command-line wrapper

WHAT'S NEW

(FLORA LOGAN, ULRIKE BÖHME & MATT ROGERS)

get_changes_in_chromosome_9.pl

```perl
use strict;
use warnings;
use Crawl::Crawler;

my $crawler = Crawl::Crawler->new();

# Store the annoation_changes in a temporary hashtable (returned by crawler as a collection).
my %annotation_changes;
foreach my $annotation_change (@{$crawler
                                  ->query("histories/annotation_changes", {taxonomyID => "185431", since=> "2009-01-01"})
                                  ->{response}->{results}}) {
    $annotation_changes{$annotation_change->{gene}} = $annotation_change;
}

# Check the genes in the chromosome you're interested in to see which ones have changed.
print "gene\tfeature\tfeature_type\ttype\tuser\tdate\tchange\n";
foreach my $gene (@{$crawler
                        ->query("genes/inregion", {region => "Tb927_09_v4"})
                        ->{response}->{genes}}) {
    my $gene_name = $gene->{gene};
    if (exists($annotation_changes{$gene_name})) {
        my $gene_result = $annotation_changes{$gene_name};
        print $gene_result->{gene}          . "\t";
        print $gene_result->{feature}       . "\t";
        print $gene_result->{feature_type}  . "\t";
        print $gene_result->{type}          . "\t";
        print $gene_result->{user}          . "\t";
        print $gene_result->{date}          . "\t";
        print $gene_result->{change}        . "\n";
    }
}
```

Line: 5 Column: 37 Perl Soft Tabs: 4

PFAM DOMAINS -> CODA

(ADAM REID)

get_pfam_domains.pl

```perl
#!/usr/local/bin/perl -w
use strict;
use Crawl::Crawler;
use Crawl::Util;

my $crawler = Crawl::Crawler->new();

# Get list of genes for Plasmodium falciparum (NCBI Id: 5833)
my $result = $crawler->query("genes/inorganism", {taxonID => "5833"});

# Loop through genes
foreach my $gene (@{$result->{response}->{genes}}) {
    # Get domains for that gene
    my $dom_res = $crawler->query("genes/domains", {genes=> $gene->{gene}});

    #Print out Pfam domains
    foreach my $dom (@{$dom_res->{response}->{results}})
    {
            print "$gene->{gene}\t$dom->{domain}\n" if $dom->{domain} =~ /Pfam/;
    }
}
```

Line: 1 Column: 1 Perl Soft Tabs: 4

A QUERY THAT IS USEFUL FOR A BIOLOGIST LOCALLY COULD WELL BE USEFUL FOR A BIOINFORMATICIAN REMOTELY

COMMAND LINE EXAMPLES

COMMAND LINE - GET ORGANISMS

gv1@ubuntu: ~/code/test — ssh — 101×20

```
gv1@ubuntu:~/code/test$ crawler.py -query organisms/list -database localhost/pathogens?pathdb \
>     | jsawk "return this.response.organisms" \
>     | jsawk -q "[?name.match('Yersinia')]"
Password:
[{"name":"Yersinia enterocolitica","organism_id":"72","taxonomyid":"630"},{"name":"Yersinia pestis","
organism_id":"71","taxonomyid":"632"}]
gv1@ubuntu:~/code/test$
```

Uses SpiderMonkey, Jsawk & JSONQuery

COMMAND LINE - FORMAT GENES

gv1@ubuntu: ~/code/test — ssh — 101×20

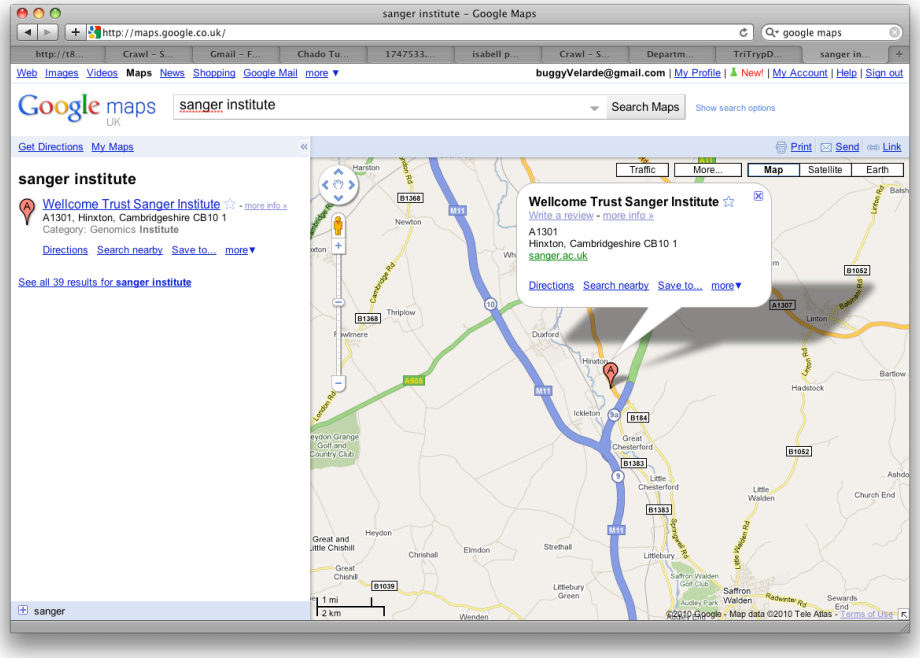
```
gv1@ubuntu:~/code/test$ crawler.py -query genes/inregion -database localhost/pathogens?pathdb \
>     -region Pf3D7_01 \
>     | jsawk "return this.response.genes" \
>     | jsawk -a 'return this.join("\n")' "return this.gene + '\t' + this.fmin + '\t' + this.fmax"
Password:
PFA0170c        148148  153011
PFA0315w        272692  273641
PFA0380w        315773  320559
PFA0440w        364470  365145
PFA0515w        406892  413323
PFA0370w        304090  304552
PFA0045c        62419   63633
PFA0115w        104935  105441
PFA0560c        443694  444441
PFA0310c        265446  269412
PFA0240w        211107  213243
PFA0025c        53391   53503
PFA0570w        447176  450433
PFA0705c        562298  563256
PFA0335w        287166  289134
```

COLLABORATIVE INTERFACES

Rich internet applications

AJAX TECHNIQUES

- Data refresh without page refresh
- Conservative
  - Autocomplete
- Advanced
  - Rich internet applications e.g. Google maps



Fundamentally depends on web services

REMINDER :
VAPORWARE ALERT!

Very early prototypes…

Anything can go wrong!

A PROTOTYPE WEB-APP

(TIM CARVER)

Good example of an AJAX driven application.

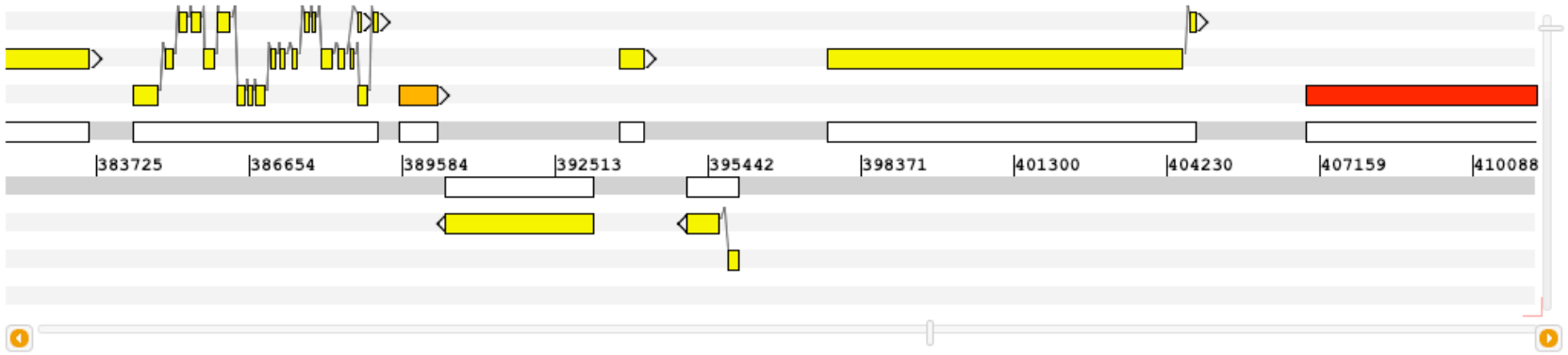
Some people call it Web-Artemis.

Pf3D7_01

http://localhost:8888/web-artemis/#

Most Visited | Getting Started | Latest Headlines | Sanger Intweb | Sanger Personnel Se... | Agresso | jQuerify

View | Graph | Plasmodium falciparum 3D7 | Sequence:



Name	Type	Feature Start	Feature End	Properties
PFA0480w	gene	381850	383578	
PFA0480w:mRNA	mRNA	381850	383578	
PFA0480w:exon:1	exon	381850	383578	
PFA0480w:pep	polypeptide	381850	383578	
PFA0485w	gene	384437	389094	
PFA0485w.2	mRNA	384437	388779	
PFA0485w.2:exon:2	exon	385040	385166	
PFA0485w.2:exon:1	exon	384437	384876	
PFA0485w.2:exon:18	exon	388749	388779	
PFA0485w.2:exon:3	exon	385315	385424	
PFA0485w.2:exon:4	exon	385553	385692	
PFA0485w.2:exon:5	exon	385780	385961	
PFA0485w.2:exon:6	exon	386050	386253	
PFA0485w.2:exon:7	exon	386417	386539	
PFA0485w.2:pep	polypeptide	384437	388779	comment=alternate spliced form revealed by splice bridging read pair transcriptome sequence. Confirmed by 4 reads (PMID:20141604).;
PFA0485w.2:exon:8	exon	386614	386674	

Done

Pf3D7_01

http://localhost:8888/web-artemis/#

Most Visited | Getting Started | Latest Headlines | Sanger Intweb | Sanger Personnel Se... | Agresso | jQuerify

View | Graph | Plasmodium falciparum 3D7 | Sequence:

```
  #  #  #  #  *  I  L  Q  K  I  K  R  *  E  R  F  R  C  T  Y  K  *  I  Y  N  I  L  K  D  K  #  E  I  #  Y  *  S  #  #  S  N  G  Y  D  #  K
 D  N  N  N  N  E  Y  Y  K  K  L  K  D  E  K  G  L  D  V  L  I  N  E  Y  T  T  S  L  K  I  S  K  K  Y  N  I  D  H  N  K  V  M  G  M  I  K  K
  I  I  I  I  M  N  I  T  K  N  #  K  M  R  K  V  +  M  Y  L  #  M  N  I  Q  H  P  #  R  #  V  R  N  I  I  L  I  I  I  K  #  W  V  *  L  K  N
GATAATAATAATAATGAATATTACAAAAAATTAAAAGATGAGAAAGGTTTAGATGTACTTATAAATGAATATACAACATCCTTAAAGATAAGTAAGAAATATAATATTGATCATAATAAAGTAATGGGTATGATTAAAAA
  |382004        |382018        |382032        |382046        |382060        |382074        |382088        |382102        |382116        |382130
CTATTATTATTATTACTTATAATGTTTTTTAATTTTCTACTCTTTCCAAATCTACATGAATATTTACTTATATGTTGTAGGAATTTCTATTCATTCTTTATATTATAACTAGTATTATTTCATTACCCATACTAATTTTT
  L  L  L  L  S  Y  #  L  F  N  F  S  S  F  P  K  S  T  S  I  F  S  Y  V  V  D  K  F  I  L  L  F  Y  L  I  S  *  L  L  T  I  P  I  I  L  F
 I  I  I  I  I  F  I  V  F  F  #  F  I  L  F  T  #  I  Y  K  Y  I  F  I  C  C  G  #  L  Y  T  L  F  I  I  N  I  M  I  F  Y  H  T  H  N  F  F
  Y  Y  Y  Y  H  I  N  C  F  I  L  L  H  S  L  N  L  H  V  #  L  H  I  Y  L  M  R  L  S  L  Y  S  I  Y  Y  Q  D  Y  Y  L  L  P  Y  S  #  F  V
```

Name	Type	Feature Start	Feature End	Properties
PFA0480w	gene	381850	383578	
PFA0480w:mRNA	mRNA	381850	383578	
PFA0480w:exon:1	exon	381850	383578	
PFA0480w:pep	polypeptide	381850	383578	

Done

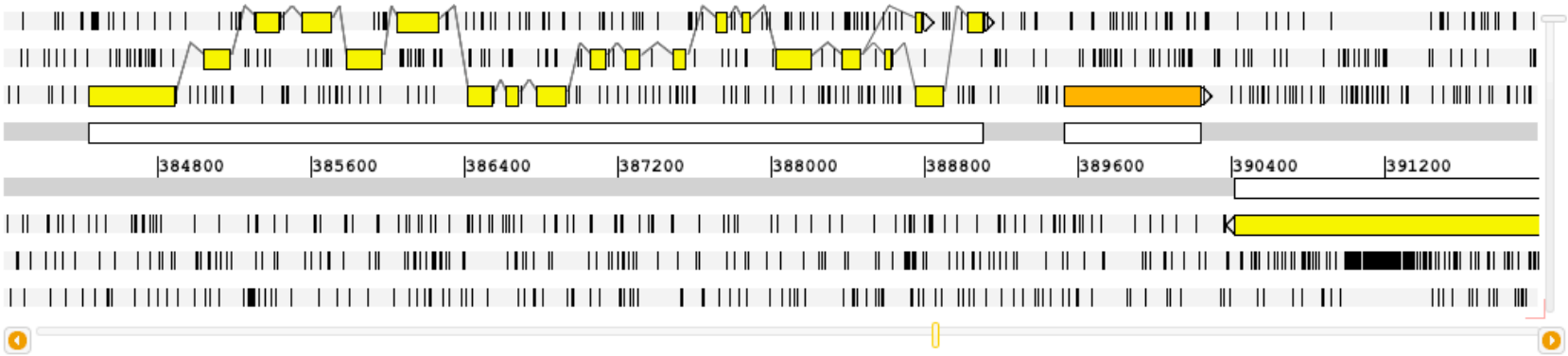
Pf3D7_01

http://localhost:8888/web-artemis/

Most Visited Getting Started Latest Headlines Sanger Intweb Sanger Personnel Se... Agresso jQuerify

View Graph

Plasmodium falciparum 3D7 Sequence:



Name	Type	Feature Start	Feature End	Properties
PFA0485w	gene	384437	389094	
PFA0485w.2	mRNA	384437	388779	
PFA0485w.2:exon:6	exon	386050	386253	
PFA0485w.2:exon:11	exon	387242	387304	
PFA0485w.2:exon:12	exon	387484	387547	
PFA0485w.2:pep	polypeptide	384437	388779	comment=alternate spliced form revealed by splice bridging read pair transcriptome sequence. Confirmed by 4 reads (PMID:20141604).;
PFA0485w.2:exon:13	exon	387711	387760	
PFA0485w.2:exon:14	exon	387844	387876	
PFA0485w.2:exon:15	exon	388024	388196	
PFA0485w.2:exon:16	exon	388367	388453	
PFA0485w.2:exon:1	exon	384437	384876	
PFA0485w.2:exon:2	exon	385040	385166	
PFA0485w.2:exon:3	exon	385315	385424	
PFA0485w.2:exon:4	exon	385553	385692	
PFA0485w.2:exon:5	exon	385780	385961	
PFA0485w.2:exon:18	exon	388749	388779	

Done

Pf3D7_01

http://localhost:8888/web-artemis/

Most Visited | Getting Started | Latest Headlines | Sanger Intweb | Sanger Personnel Se... | Agresso | jQuerify

View | Graph

Plasmodium falciparum 3D7 | Sequence:

384800 385600 386400 387

Name	Type	Feature Start	Feature End
PFA0485w	gene	384437	389094
PFA0485w.2	mRNA	384437	388779
PFA0485w.2:exon:6	exon	386050	386253
PFA0485w.2:exon:11	exon	387242	387304
PFA0485w.2:exon:12	exon	387484	387547
PFA0485w.2:pep	polypeptide	384437	388779
PFA0485w.2:exon:13	exon	387711	387760
PFA0485w.2:exon:14	exon	387844	387876
PFA0485w.2:exon:15	exon	388024	388196
PFA0485w.2:exon:16	exon	388367	388453
PFA0485w.2:exon:1	exon	384437	384876
PFA0485w.2:exon:2	exon	385040	385166
PFA0485w.2:exon:3	exon	385315	385424
PFA0485w.2:exon:4	exon	385553	385692
PFA0485w.2:exon:5	exon	385780	385961
PFA0485w.2:exon:18	exon	388749	388779

PFA0485w.1

colour=7
PlasmoAP_score=2
comment=Barrell, October 2008, trimmed 5' to third Met in frame and deleted exon 13 in comparison with P. knowlesi and P. vivax
SignalP_prediction=Signal anchor
signal_peptide_probability=0.015
signal_anchor_probability=0.830
colour=7

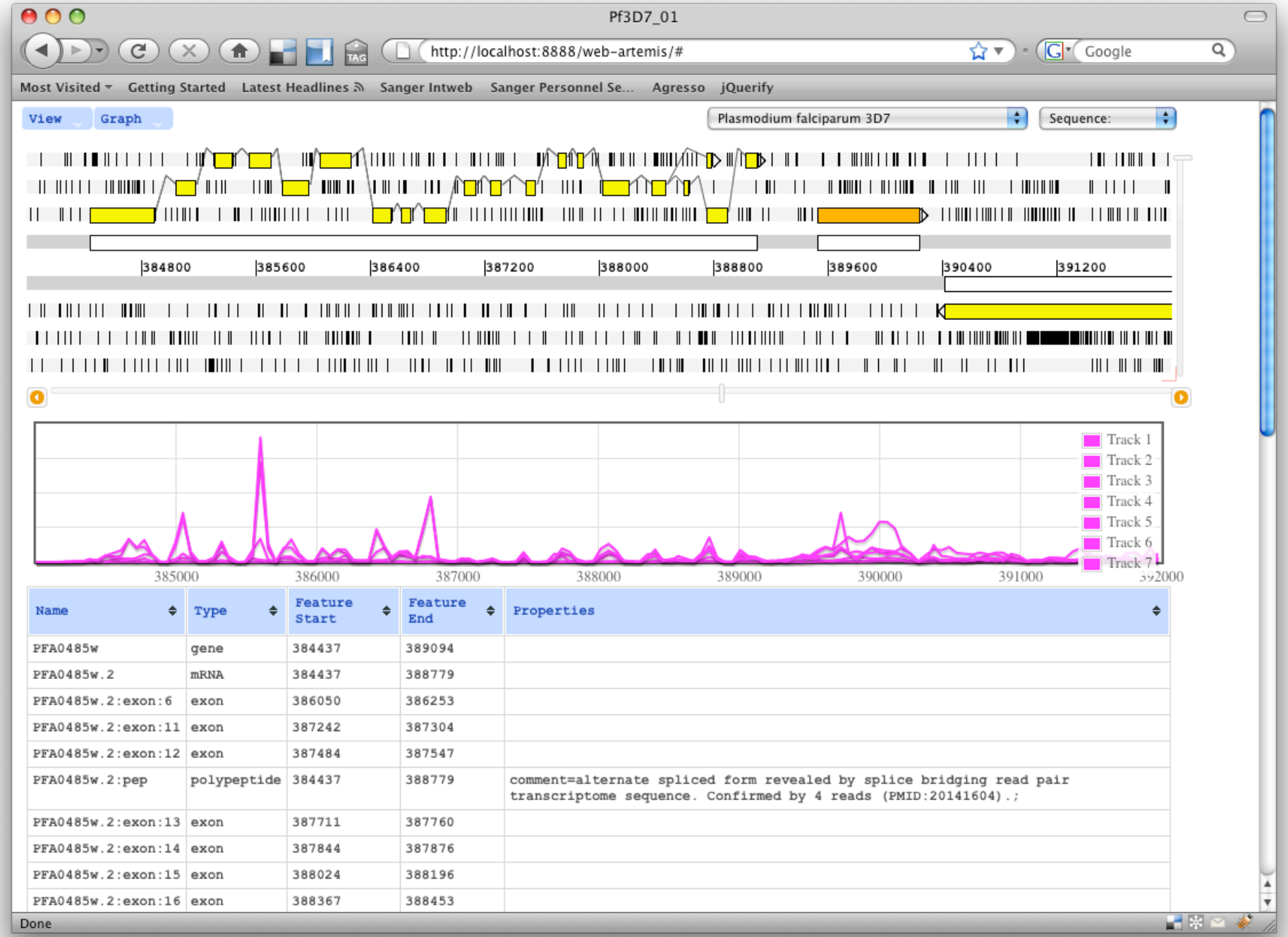
DbXRefs :
MPMP:dolicholmetpath.html; OPI:PFA0485w; OrthoMCLDB:PFA0485w; PlasmoDB:PFA0485w;

Product :
phosphatidate cytidylyltransferase, putative

Gene Ontology :
GO:0020011; aspect=C; apicoplast; evidence=inferred from Reviewed Computational Analysis; PMID:11738814;
GO:0016772; aspect=F; transferase activity, transferring phosphorus-containing groups; autocomment=From iprscan; evidence=Inferred from Electronic Annotation; date=20100131;
GO:0016020; aspect=C; membrane; autocomment=From iprscan; evidence=Inferred from Electronic Annotation; date=20100131;

Clusters :
PBANKA_020370:pep
PCAS_020210:pep
PFA0485w.1:pep
PFA0485w.2:pep

Done



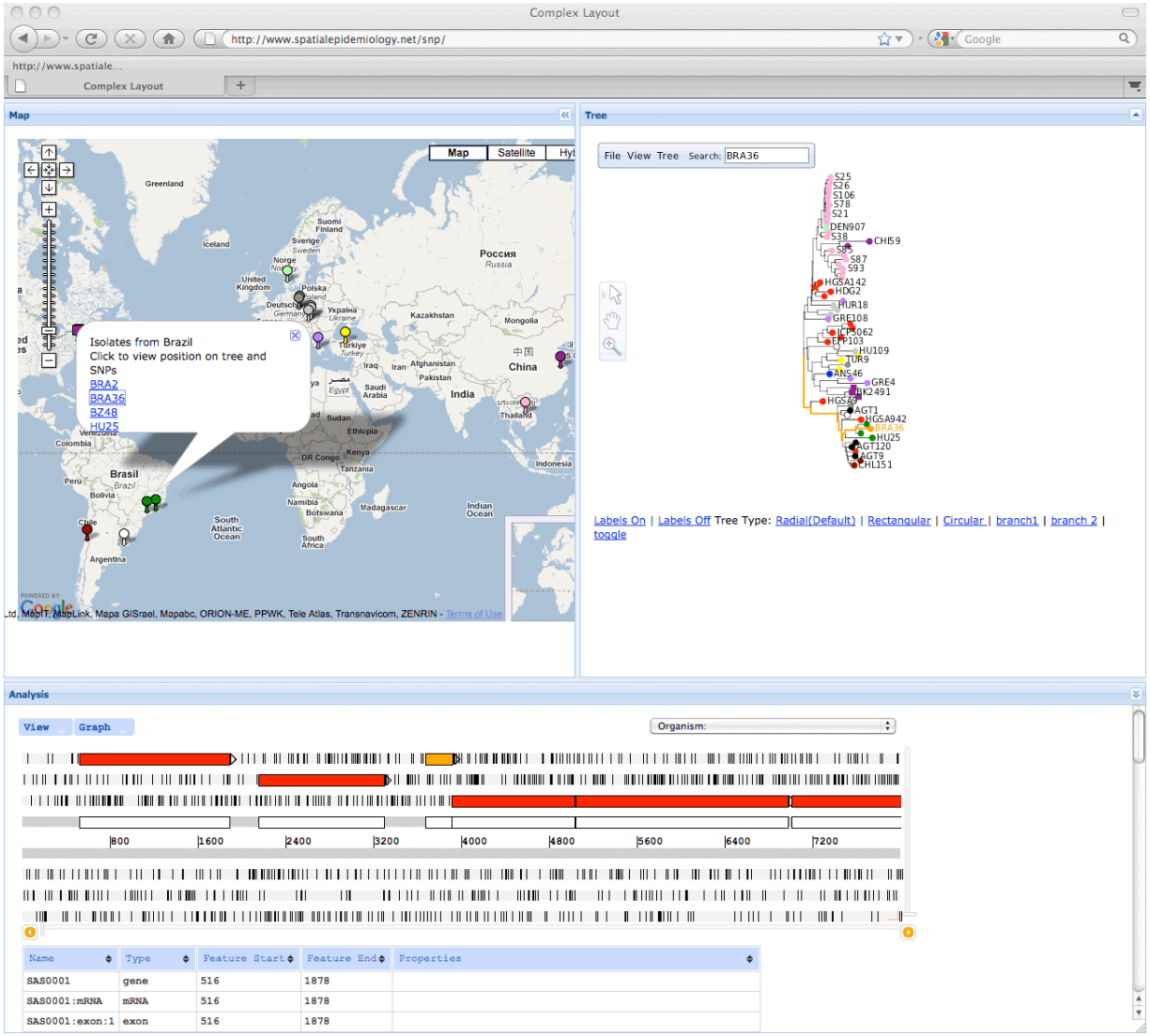
Name	Type	Feature Start	Feature End	Properties
PFA0485w	gene	384437	389094	
PFA0485w.2	mRNA	384437	388779	
PFA0485w.2:exon:6	exon	386050	386253	
PFA0485w.2:exon:11	exon	387242	387304	
PFA0485w.2:exon:12	exon	387484	387547	
PFA0485w.2:pep	polypeptide	384437	388779	comment=alternate spliced form revealed by splice bridging read pair transcriptome sequence. Confirmed by 4 reads (PMID:20141604).;
PFA0485w.2:exon:13	exon	387711	387760	
PFA0485w.2:exon:14	exon	387844	387876	
PFA0485w.2:exon:15	exon	388024	388196	
PFA0485w.2:exon:16	exon	388367	388453	

POTENTIAL USES

- Genomics visualisation tool
  - Including GeneDB of course
- Community annotation
  - Users won't have to use a desktop app via VPN
    - Simple annotations
- and …

SNP-MASHUPS

(TIM CARVER & DAVID AANENSEN)



SMASHUPS! (SNP-MASHUPS)

- Embedding as a widget
  - Combining
    - web services
    - independently written widgets
  - Integrate software that integrates data

THE TAKE HOME MESSAGE

- The Way of the CRAWL
  - Built as a library first
  - Deployed as
    - Standalone Web services app
    - Command line app
  - Used for
    - Collaborating with EupathDB
    - Query multi-organism data sets in house
      - without going through WS
    - Building RIAs and (s)mashups

GMODREST

- Currently
  - **Python** REST framework (Ropy)
  - Speaks **Chado**
- Time to implement the GMODREST interface?
- Caveats
  - GeneDB's Chado may have little differences
    - must test on other DBs

THANKS!

Adrian Tivey
Anne Pajon
Jacqueline McQuillan
Martin Aslett
Nishadi De Silva
Raeece Naem
Robin Houston
Tim Carver
Tina Eyre

Adam Reid
Flora Logan
Gary Dillon
Matt Holden
Matt Rogers
Thomas Dan Otto
Ulrike Böhme

Brian Brunk
Cary Pennington
Cristina Aurrecoechea
John Iodice
Omar Harb
Steve Fischer
Vishal Nayak

David Aanensen

Christiane Hertz-Fowler
Debbie Smith
Mark Carrington
Matt Berriman
David Roos