Data validation, storage, and visualization with GMOD tools for modENCODE

Nicole Washington
Lawrence Berkeley National Lab
modENCODE Data Collection Center

The modENCODE project

Background:

- Organisms
  - *Drosophila melanogaster*
  - *Caenorhabditis elegans*
- 11 NIH-funded groups
  - 5 fly & 5 worm groups
  - 1 data coordination center

The modENCODE project

Research projects:

- Find regulatory elements
  - ChIP-chip and ChIP-seq
- Find evidence for all gene structures & miRNAs
  - RACE, RT-PCR, cDNA seq, RNA-Seq
- Expression profiles of mRNAs under various conditions
  - Arrays, RNA-Seq
- Annotate UTR regulatory regions
- Examine origins of replication (fly)

The modENCODE project
Data coordination center (DCC):

- Develop specs for submissions
  - Metadata & data
  - Ensure uniformity across data types
- Data management
  - Collect & Validate
  - Store data/metadata
  - Provide statistics
- Serve to community
  - Interfaces for browsing & analysis

DCC Data management
Problems to tackle:

- Capture both data & experimental details
  - Store together in database
  - Utilize downstream for data analysis
- Basic responsibilities
  - Collection
  - Validation
  - Storage
  - Serving
- Provide links between results from different researchers, data types, organisms, etc.

DCC Data management
Solutions

- Extend existing gmod tools:
  - Data & metadata storage with **Chado**
  - Data visualization with **Gbrowse**
  - (Metadata visualization with **modMine**)

- Develop new tools:
  - Meta/Data validation
  - Track "finding" via data introspection
  - Submission & publishing pipeline

Storing metadata in Chado

- Requirements:
  - Strong data typing (ontologies)
  - Links between metadata and resulting features
  - Normalized, consistent with Chado schema
  - Methods to add/drop new data easily

Formalizing experimental metadata: BIR-TAB & MediaWiki extensions

- Two data files:
  - **IDF**: Investigation Design
    - Declare protocols and controlled vocabulary
  - **SDRF**: Sample-Data Relationship
    - Applications of protocols - inputs and outputs
    - Data and/or references to data
- Use Wiki Forms for additional controlled parameters
  - Define all protocols (incl. typing with ontologies)
  - Define additional reagents: Abs, Strains, Stages, etc.

BIR-TAB components (IDF)

Investigation Title	Demonstration UHTS Experiment	
Experimental Design	transcript_identification_design	
Experimental Factor Name	Kc167	
Experimental Factor Type	CellLine	
Experimental Factor Term Source R	MO	
Person Last Name	Celniker	Hoskins
Person First Name	Sue	Roger
Person Mid Initials		
Person Email	celniker@fruitfly.org	roger@fruitfly.org
Person Phone		314-286-0207
Person Address	1 Cyclotron Rd., MS64-121; Berkeley, CA 94720	1 Cyclotron Rd., MS64-121; Berkeley, CA 94720
Person Affiliation	LBNL	LCG, Washington Unversity
Person Roles	investigator	investigator
Person Roles Term Source REF	MO	MO
Quality Control Type	biological_replicate	
Quality Control Term Source REF	MO	
Replicate Type	biological_replicate	
Replicate Term Source REF	MO	
Date of Experiment	You can fill in or we can fill in	
Public Release Date	(we fill in)	
PubMed ID		
Experiment Description	ModencodeWiki	
Protocol Name	High throughput sequencing	
Protocol Type	sequencing_protocol	
Protocol Description	http://heartbroken.lbl.gov/project/index.php?title=BDGP_5%27_RLM-RACE&oldid=7703	
Protocol Parameters	BioSample	
Protocol Term Source REF	MO	
SDRF File	UHTS-SDRF.csv	
Term Source Name	MO	ME
Term Source File	http://www.berkeleybop.org/ontologies/obo-all/mged/mged.obo	http://wiki.modencode.org/project/extensions/DBFiel
Term Source Version	1.3.0.1	
Term Source Type	OBO	OBO

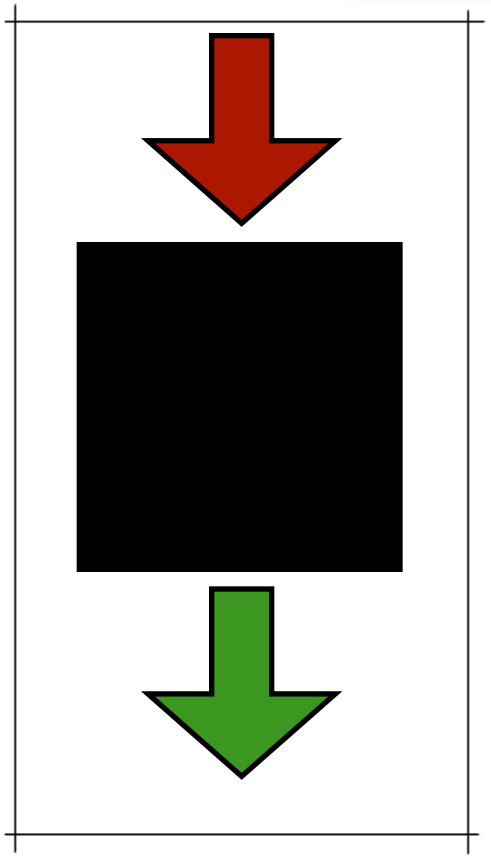
BIR-TAB components (SDRF)

- A superset of MAGE-TAB
- Applications of protocols - inputs and outputs
- Data and/or references to data
- Build an experimental graph
  - Can merge and split outputs and inputs of protocols - describe a DAG

Extract Name	Characteristics [Extract Materia	Term Source REF	Protocol REF	Result File [Raw Data File]	Result File [Derived Data File]
kc167-R1	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig
kc167-R1	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig
kc167-R2	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig
kc167-R2	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig
kc167-R3	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig
kc167-R3	Kc167	ModencodeWiki	High throughput sequencin	sequence.fasta	results.wig

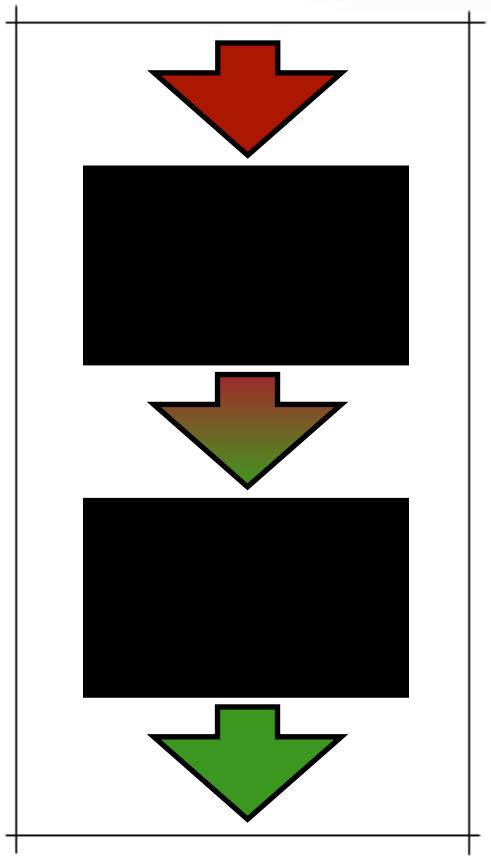
“Protocols”

- Protocols are “black boxes”
- Any input(s) can be transformed into any output(s)
- Can be as atomic – or not – as required

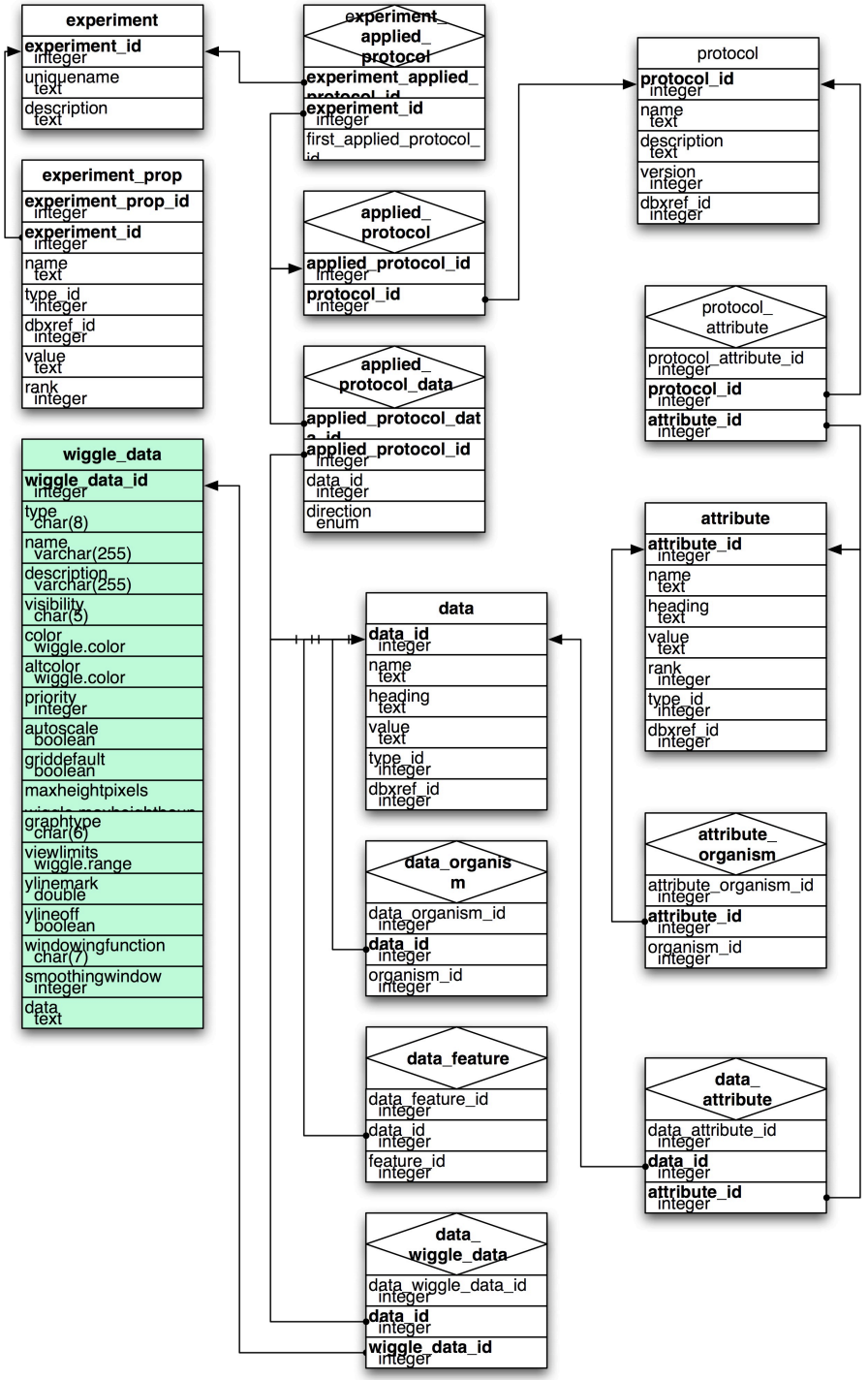


“Applied Protocols”

- Each protocol can be reused with different inputs/outputs as an “applied protocol”
- Applied protocols are chained
- When following the DAG of applied protocols, connections are made by shared data

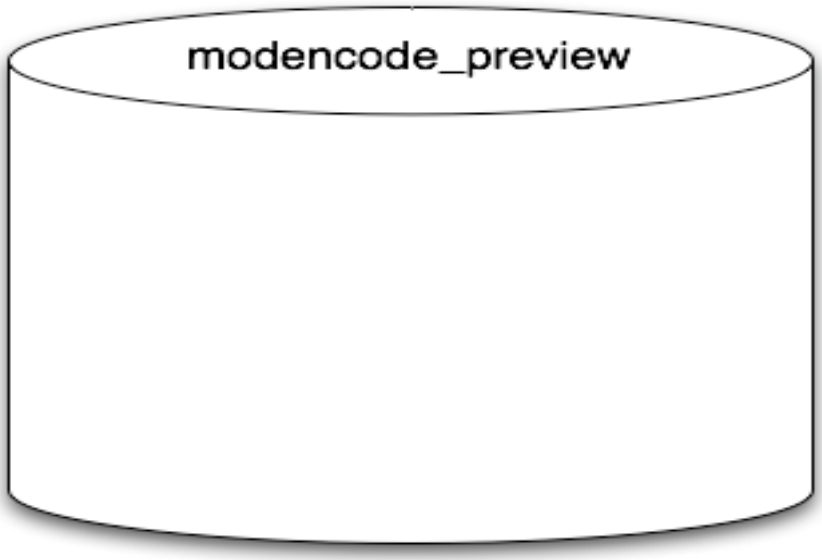


BIR-TAB in Chado SCHEMA



Chado in PostgreSQL:

Option 1: single database

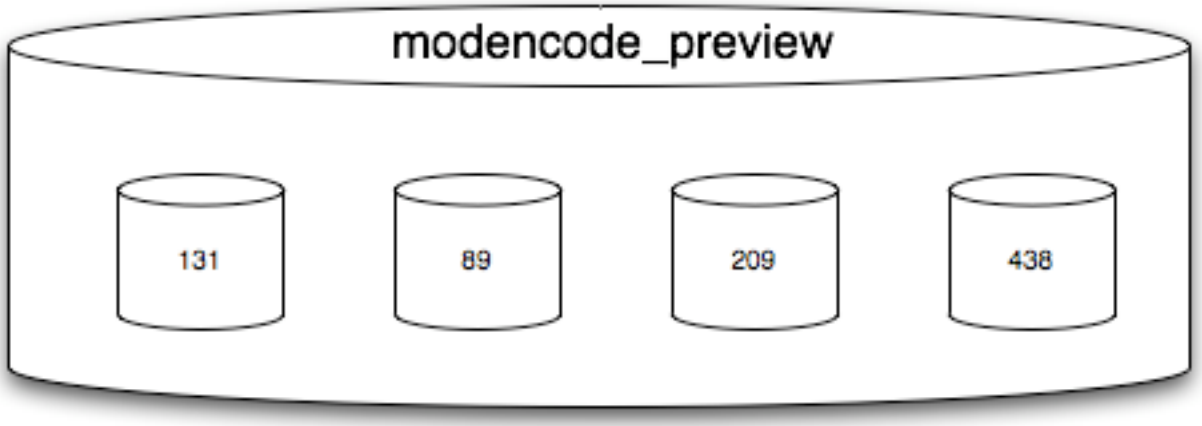


Chado in PostgreSQL:

Option 2: multiple databases
Each submission has its own DB

Chado in PostgreSQL:
combining data via namespaces

Option 3: single database with namespaces



Displaying data with GBrowse

- Requirements:
  - Everything GBrowse has, plus…
  - Need to easily add/drop data
  - Handling very large datasets
  - Want to use PostgreSQL for Bio::Seqfeature::Store

Displaying data with GBrowse

- Requirements:
  - Everything GBrowse has, plus…
  - Need to easily add/drop data
  - Handling very large datasets
  - Want to use PostgreSQL for Bio::Seqfeature::Store

- Solution:
  - Use GBrowse 2.0
  - Write Postgres adapter for Bio::Seqfeature::Store
  - Use multiple namespaces for Bio::Seqfeature::Store

Displaying data with GBrowse: stanza for v2.0


```
[modencode_preview_129:database]
db_adaptor    = Bio::DB::SeqFeature::Store
db_args       = -adaptor DBI::Pg
                -dsn     dbname=modencode_gffdb;host=localhost
                -user    '????????'
                -pass    '????????'
                -schema '129'

[white_-_dCT_WIG_130_129]
database = modencode_preview_129
feature = WIG:130
label = sub { return shift->name; }
glyph = wiggle_xyplot
max_score = 3
min_score = -1
category = Preview
pos_color = blue
neg_color = orange
label density = 100
key = white - dCTCF C-term signal intensity
```

NEW: Meta/Data validation

- Requirements:
  - Handle diverse data types
  - Modular components for maximal utility
  - Biologist user-friendly

NEW: Meta/Data validation

- Requirements:
  - Handle diverse experiment types
  - Modular components for maximal utility
  - Biologist user-friendly

- Solution
  - Wiki extension using forms for metadata entry (strains, antibodies, stages, etc.)
  - BIR-TAB metadata format directs validation pipeline
  - Validation modules invoked based on "type"
  - Output ChadoXML for max compatibility

Validation Form [edit]

(This section to be completed by Project Bioinformatics contact. Contact your DCC Liaison with questions.)

Antibody "Ab:BEAF-32:KW:1" (Version 2)

Official Name:	BEAF-32 ?
Short names/aliases:	BEAF-32 ?
Target Name:	BEAF-32 ?
Target gene product Fly/Worm name:	fly_genes:BEAF-32 ?
Species Target:	D. melanogaster ?
Species Host:	Rabbit ?
Antigenic Sequence:	MHVEKKSELRSLLKSGDKRCKLVEPRNTKSCVWRFFNLVQCDDHIEPYAC ?
Purified:	Crude - not purified ?
Poly/Monoclonal:	Polyclonal ?
Company/Lab:	Gift from Research Lab ?
Catalog number, database ID, laboratory, if applicable:	Ulrich Laemmli Lab ?
Lot Number, if applicable:	?
Short Description:	?
Reference:	?
Contributing Lab:	Russell, Steven ?

Update Restore

Please use this page's permanent link when referencing it in data submission (e.g. in the IDF):
http://wiki.modencode.org/project/index.php?title=Ab:BEAF-32:KW:1&oldid=14476
IE Users: Right-click and choose 'Copy Shortcut' to copy the permalink URL to the clipboard.

Validation Form

Protocol "Data Processing:KW:1" (Version 12)

Protocol Type: modencode-helper:peak_fi g_data_analysis_protocol ?

Input type: modencode-helper:CE ray Data File] ?

Ouput type: modencode-helper:GFF3 [Peaks], modencode-helper:WIG [signal intensitie ?

Short Description:

ChIP-chip data generated using Affymetrix Drosophila tilin
subjected to the analysis using MAT (Johnson et al 2006).
Control samples are hybridized on different arrays. The da

We specified the MAT parameters like Bandwidth, MaxGap and
be 200, 100 and 10 respectively. We used non-redundant dm3
generated Repeat Library files available for download at
http://liulab.dfci.harvard.edu/MAT/Download.htm

NEW: Track “finding”

- Requirements:
  - Introspect on a submission, find 1+ gbrowse-compatible tracks
  - Output standardized GFF3 for downstream use

NEW: Track "finding"

- Requirements:
  - Introspect on a submission, find 1+ gbrowse-compatible tracks
  - Output standardized GFF3 for downstream use

- Solution
  - Heuristics to produce different results depending on number and types of features found
  - Produce GFF3, WIG, or both, depending on input type
  - Reject for non-located features

NEW: Submission & Publishing pipeline

- Requirements:
  - Robust system for submission of data sets
  - Tracking & statistics for NIH management
  - User should control from submission to final browser
  - Public and private data sets available for different user

NEW: Submission & Publishing pipeline

- Requirements:
  - Robust system for submission of data sets
  - Tracking & statistics for NIH management
  - User should control from submission to final browser
  - Public and private data sets available for different user

- Solution
  - Built interface with Ruby
  - Dispatch perl validation modules
  - Track finding
  - Track configuration with Gbrowse session co-management
  - Statistics pages built using Google graph API

NEW: Track configuration

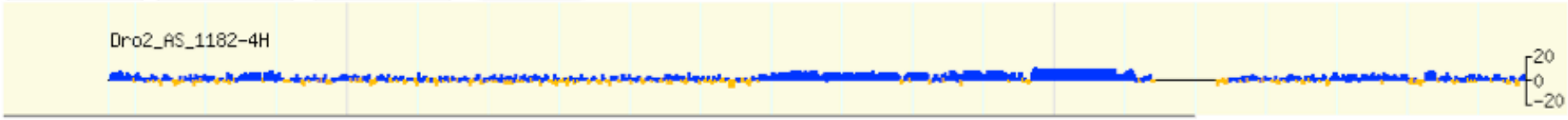
Configure tracks for submission #106

View in

Rese

[Demonstra_Signal_Graph_File_277_106]
feature = Signal_Graph_File:277
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **blue**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:277**
neg_color = **orange**

3L : 6066513 .. 6266513 Update



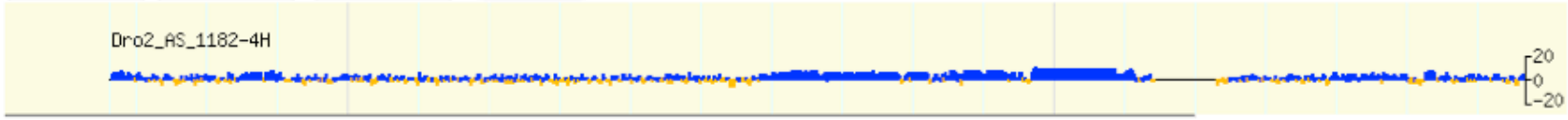
Configure tracks for submission #106

View in GBrowse

Reset All

[Demonstra_Signal_Graph_File_277_106]
feature = Signal_Graph_File:277
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **blue**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:277**
neg_color = **orange**

3L : 6066513 .. 6266513 Update



[Demonstra_Signal_Graph_File_279_106]
feature = Signal_Graph_File:279
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **blue**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:279**

modENCODE DCC Submission Pipeline

http://smaug.lbl.gov/submit/pipeline/configure_tracks/106

overflow:hide

(logout)
Logged In: **yostinso**

List [all | my group | my] submissions or create a new submission.

Stats | Report Bug | Administration

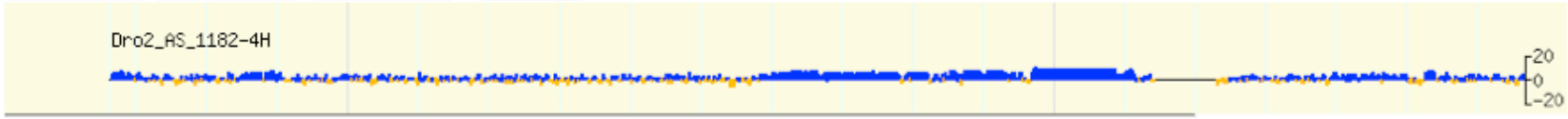
Configure tracks for submission #106

View in GBrowse

Reset All

[Demonstra_Signal_Graph_File_277_106]
feature = Signal_Graph_File:277
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = green
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:277**
neg_color = **orange**

3L : 6066513 .. 6266513 Update



[Demonstra_Signal_Graph_File_279_106]
feature = Signal_Graph_File:279
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **blue**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**

hidden Next Previous Highlight all Match case

Done

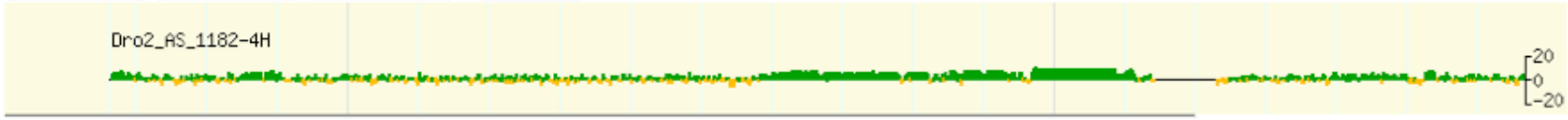
Configure tracks for submission #106

View in GBrowse

Reset All

[Demonstra_Signal_Graph_File_277_106]
feature = Signal_Graph_File:277
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **green**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:277**
neg_color = **orange**

3L : 6066513 .. 6266513 Update



[Demonstra_Signal_Graph_File_279_106]
feature = Signal_Graph_File:279
label = **sub { return shift->name; }**
category = Preview
bump density = **250**
glyph = **wiggle_xyplot**
max_score = **20**
pos_color = **blue**
fgcolor = **black**
connector = **solid**
database = modencode_preview_106
min_score = **-20**
label density = **100**
key = **106 Demonstra Signal_Graph_File:279**

Further information

- Pipeline & validation software available via svn:
  - **svn://public-svn.modencode.org/modencode**

Acknowledgements

- Most work done by: EO Stinson
- modENCODE PIs: Suzi Lewis & Lincoln Stein
- NIH funded