Ensembl Funcgen: A Database and API for Epigenomics and Gene Regulation Data.

Thomas Juettemann

Ensembl Regulation

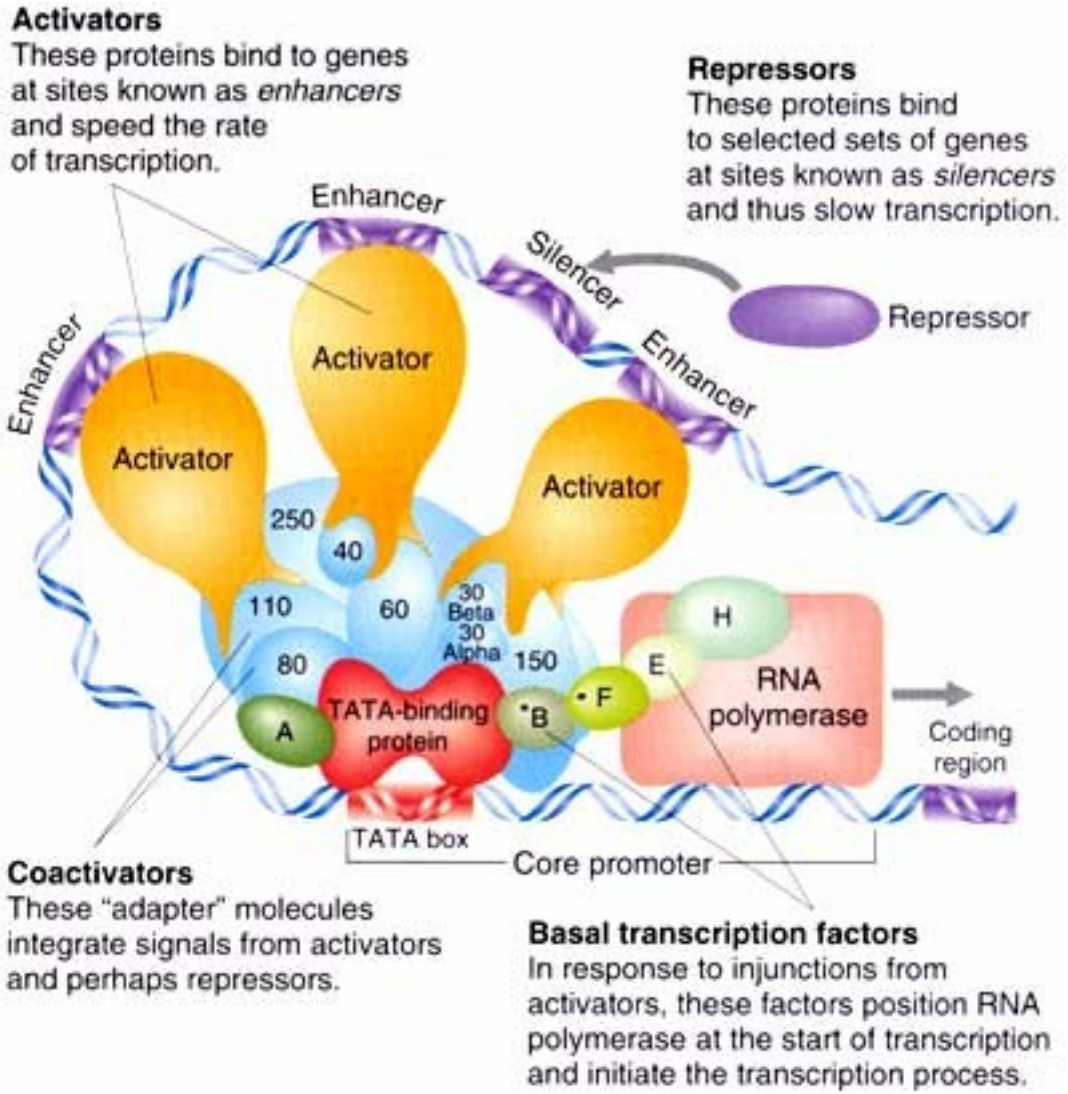
EMBL-EBI

wellcome trust sanger institute

Workshop Overview

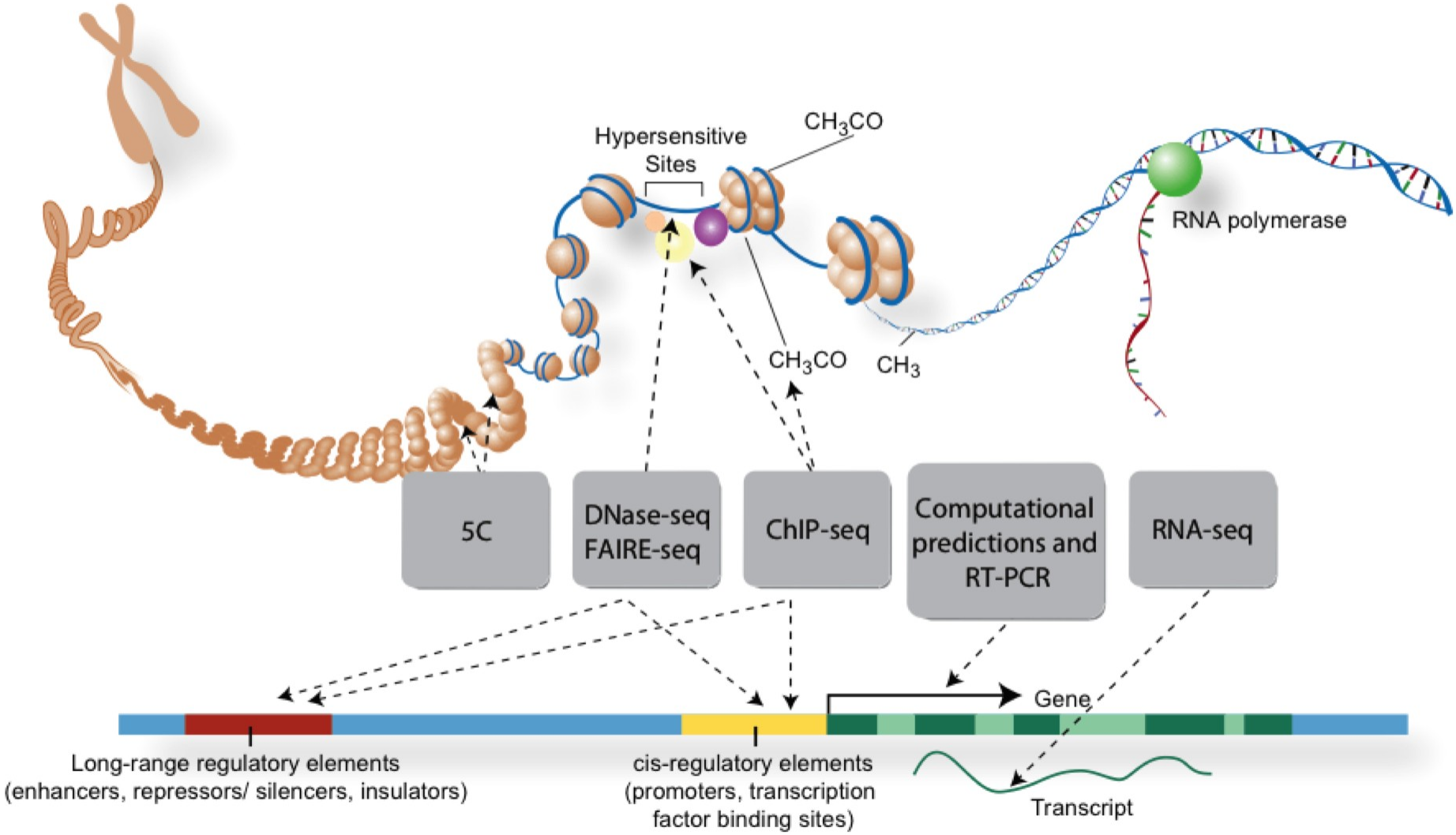
- http://www.ebi.ac.uk/~juettema/api_course/04.12.2013-Cambridge
- RegulatoryFeatures
- ENCODE Segmentation
- AnnotatedFeature: DNase1 sites, ChIP peaks etc.
- Sets
- http://www.ensembl.org/info/docs/api/funcgen/index.html
- Not covered in the course:
- MicroArrays & Probe/Set Transcript Annotations
- Raw data access i.e. Probe mappings, Read alignments
- External data

Regulatory Elements



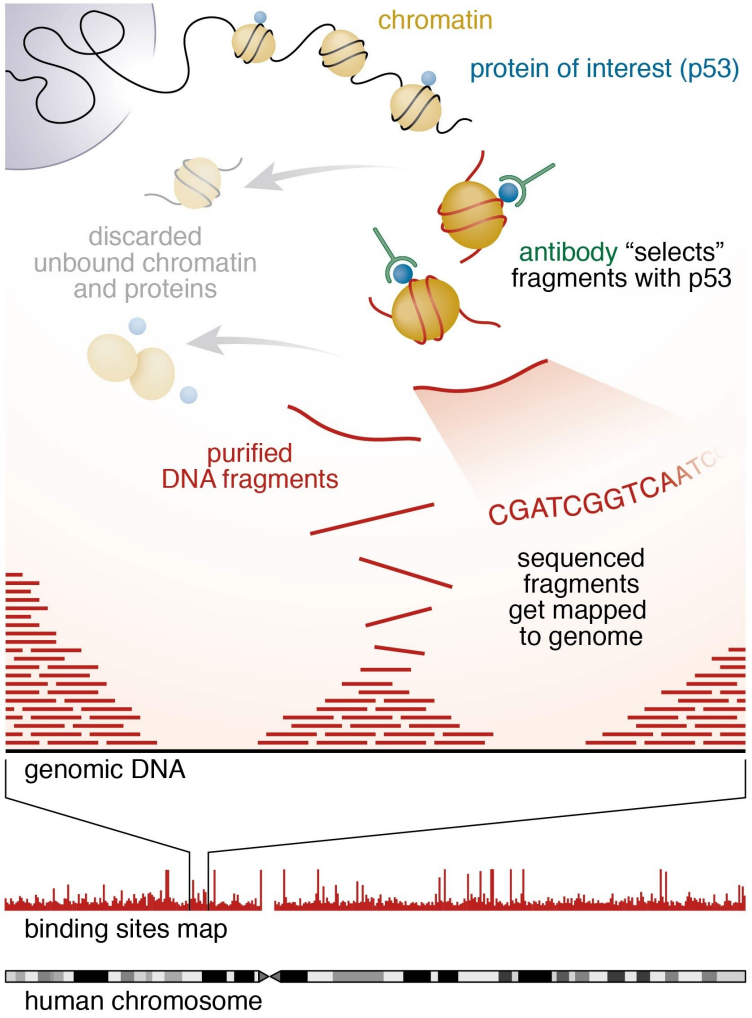
© http://scienceblogs.com/pharyngula/2007/01/16/basics-what-is-a-gene/

Aspects of Regulation



Credits: Darryl Leja (NHGRI), Ian Dunham (EBI)

ChIP-Seq: DNA-Protein binding

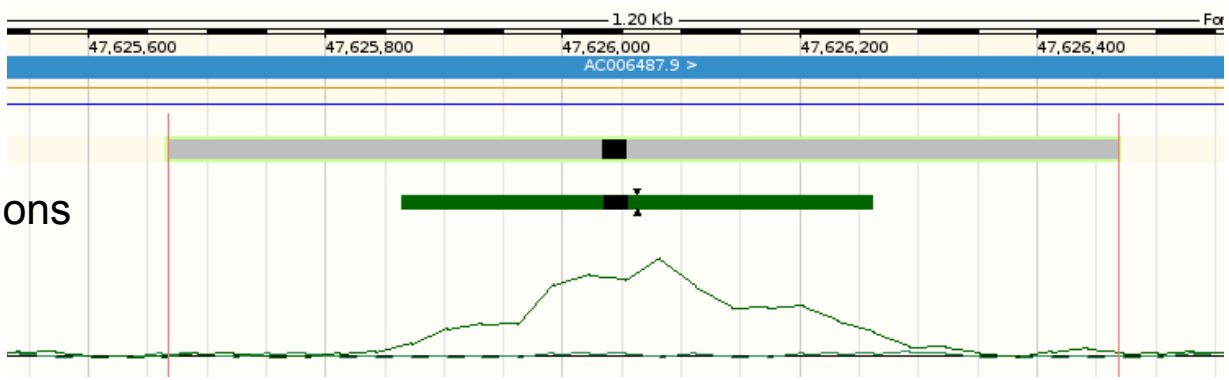


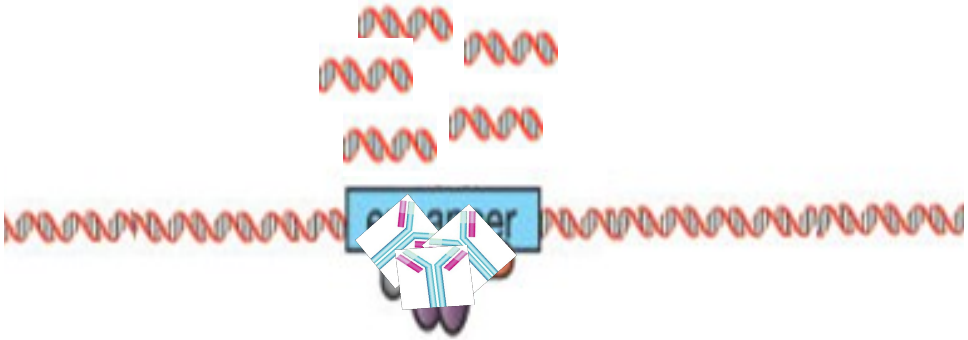
http://www.bnl.gov/bnlweb/pubaf/pr/photos/2011/11/chip_seq_illustration_final-hr.jpg

ChIP-Seq: DNA-Protein binding

enriched regions

reads

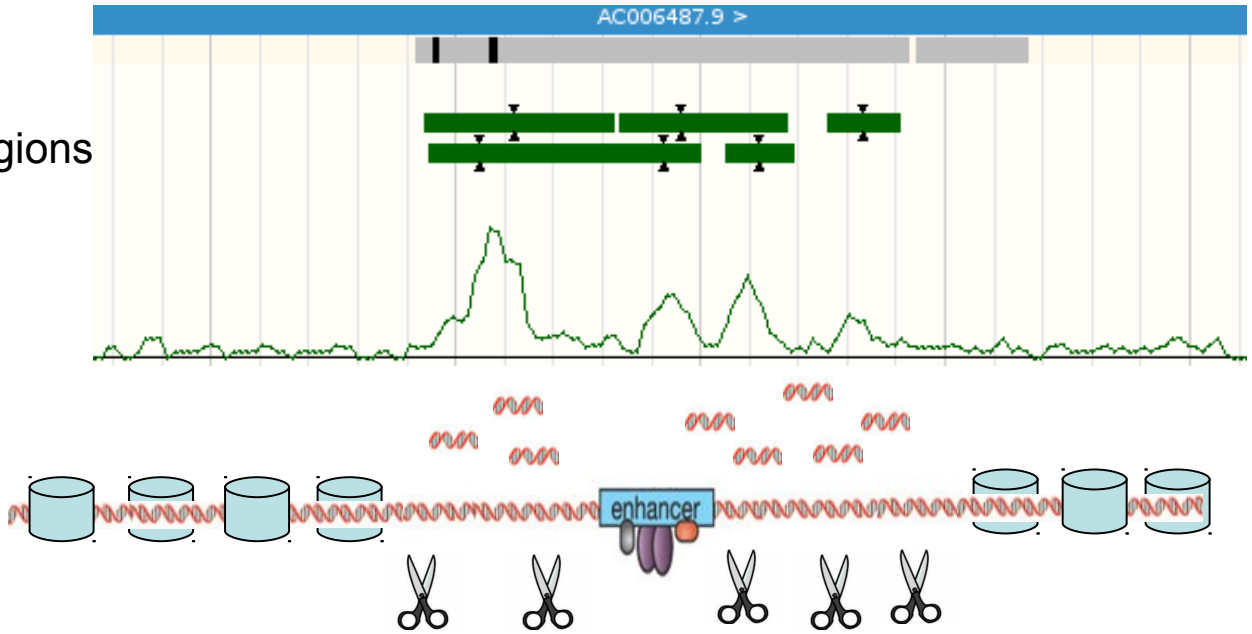




DNase-Seq: Open Chromatin

enriched regions

reads



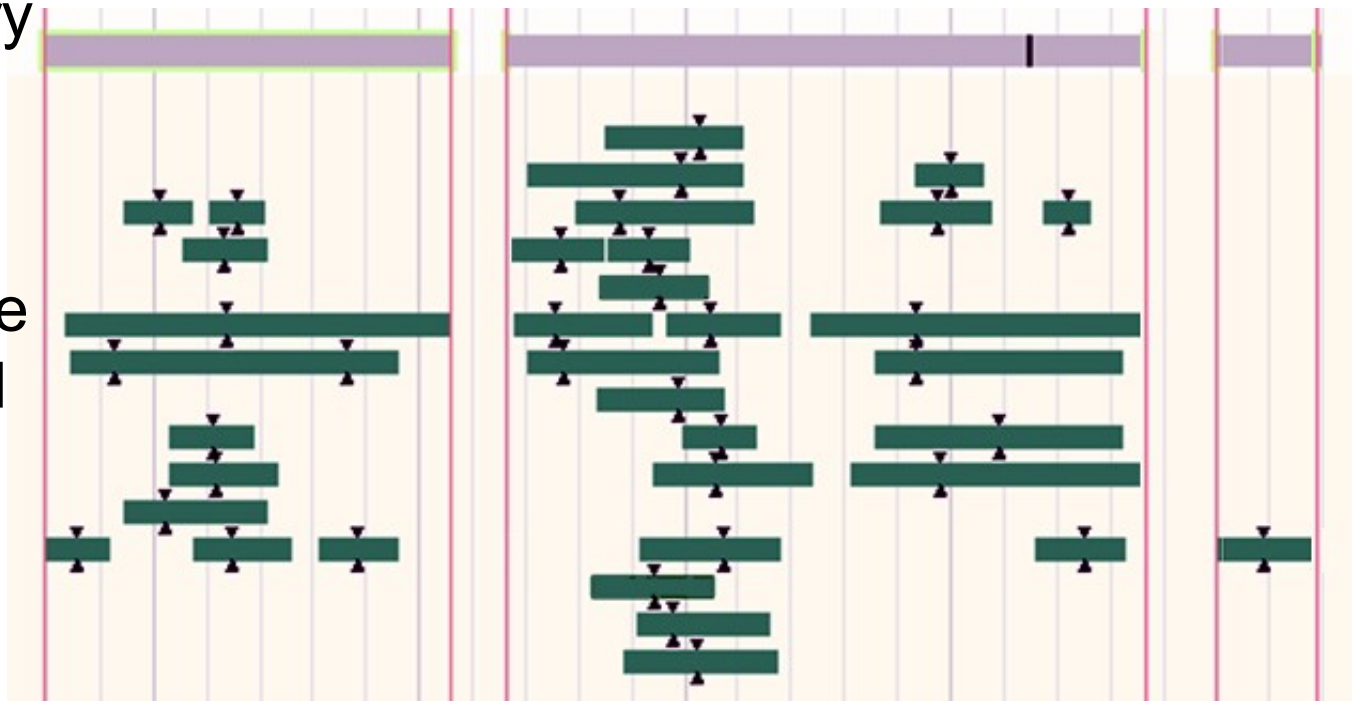
Ensembl Regulatory Build

- Identification of candidate regulatory elements (MultiCell)
- Cell specific feature construction
- Cell specific classification

MultiCell Regulatory Features

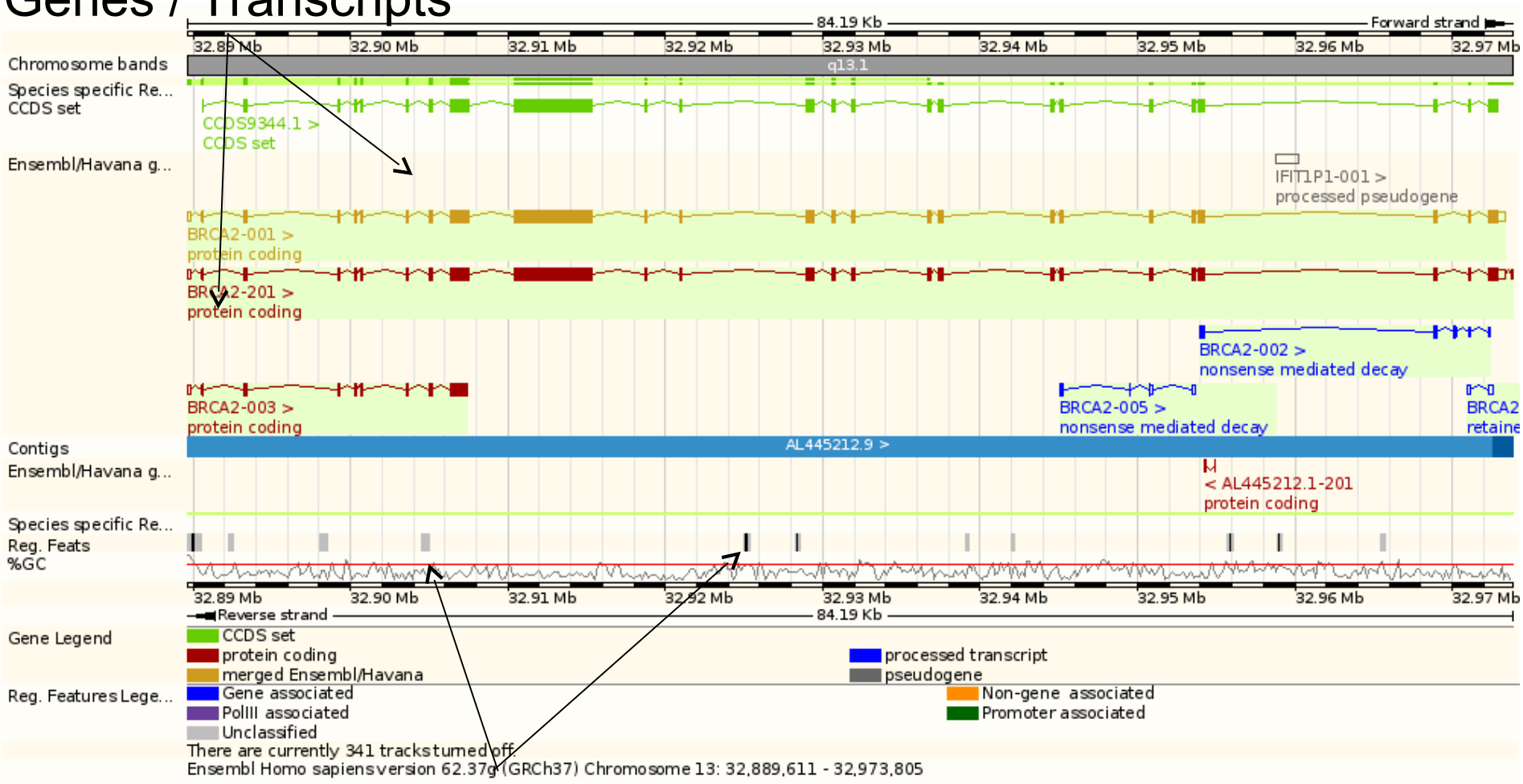
Regulatory Features

TFs/Dnase Annotated Features



Regulatory Features in Ensembl

Genes / Transcripts

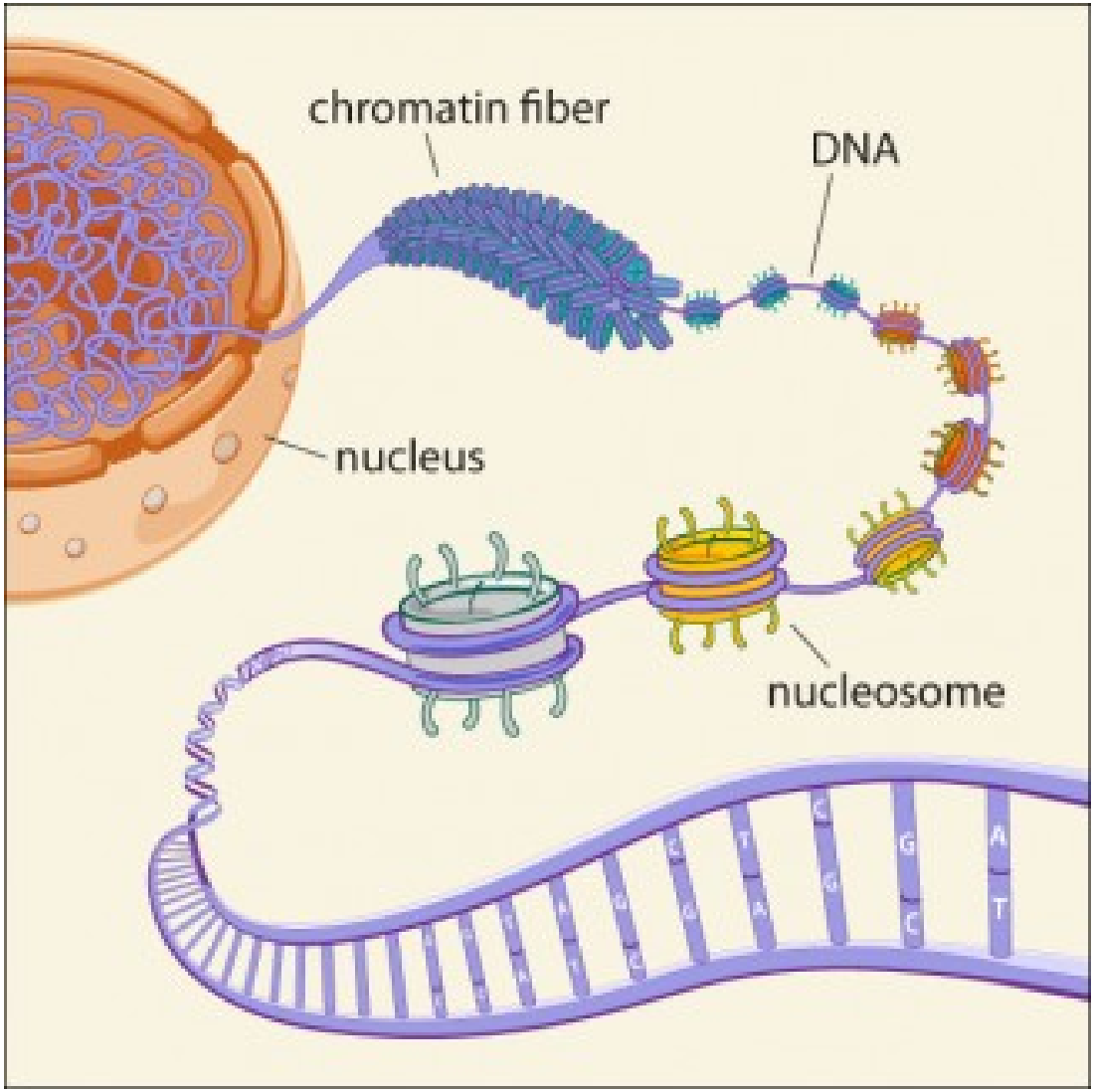


Regulatory Features

Ensembl Regulatory Build

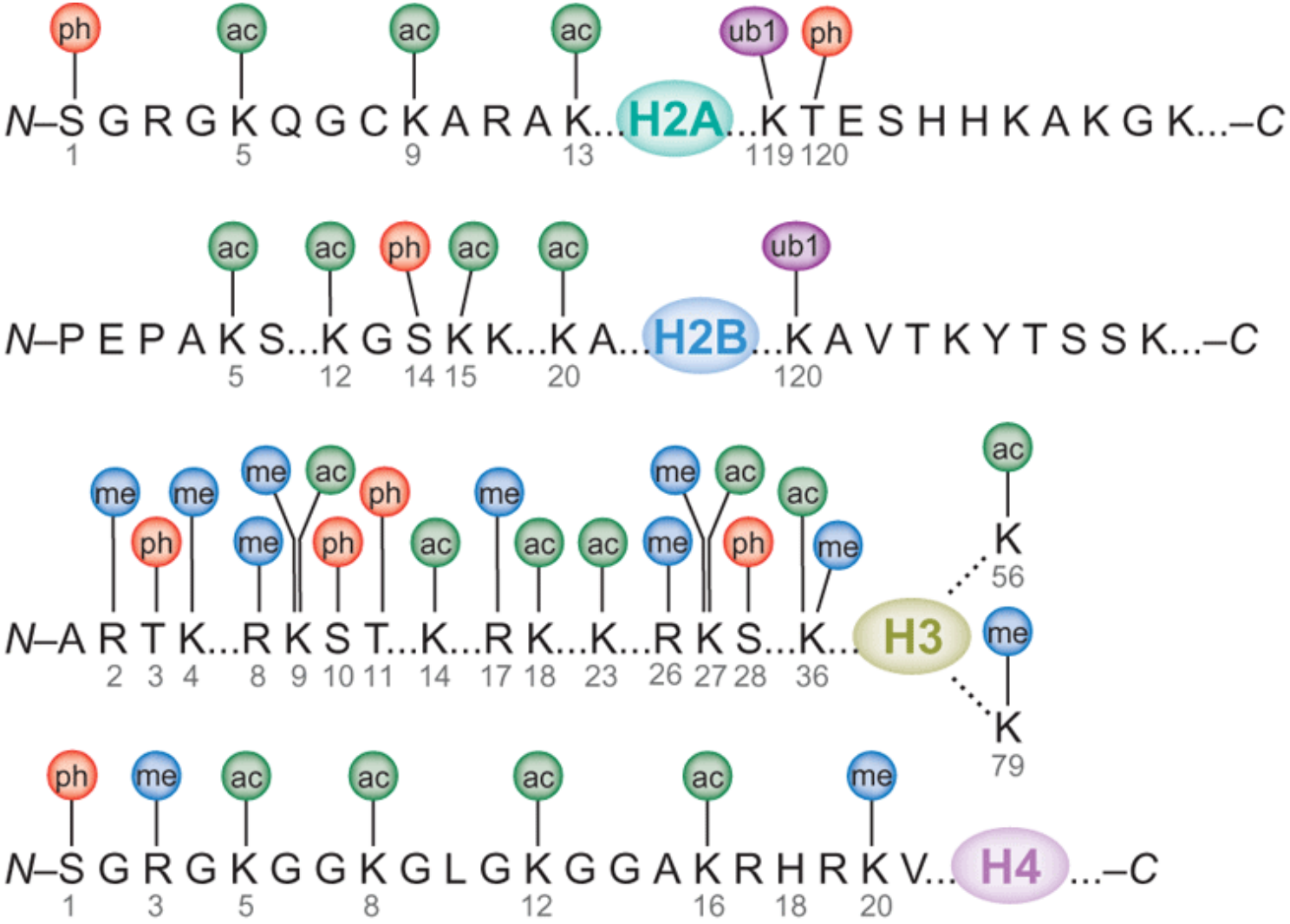
- Identification of candidate regulatory elements (MultiCell)
- Cell specific feature construction
- Cell specific classification

Beads on a string



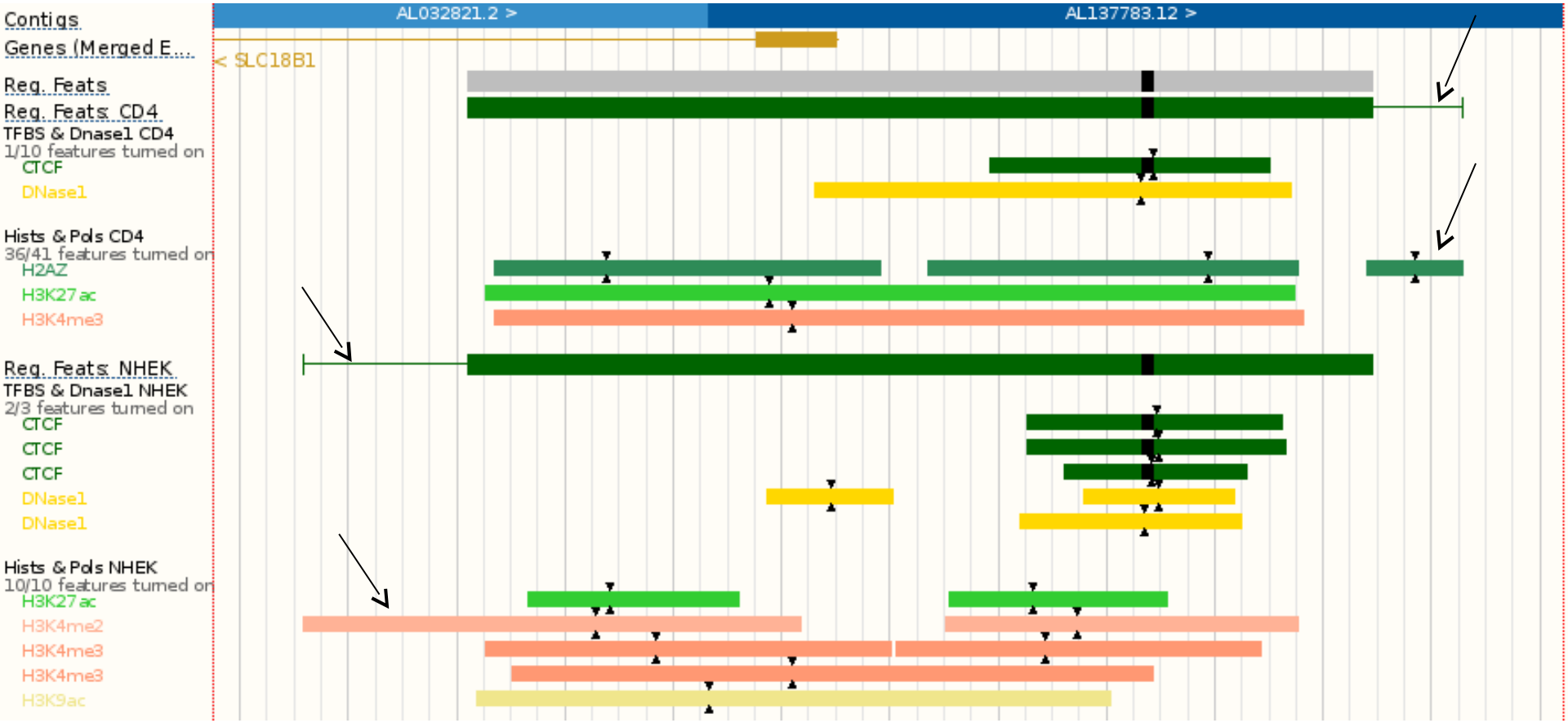
Credits: Haley Bridger, Broad Communications

Histone Modifications



Sukesh R Bhaumik, Edwin Smith & Ali Shilatifard. doi:10.1038/nsmb1337

Histone Modification and Polymerase Binding



Ensembl Regulatory Build

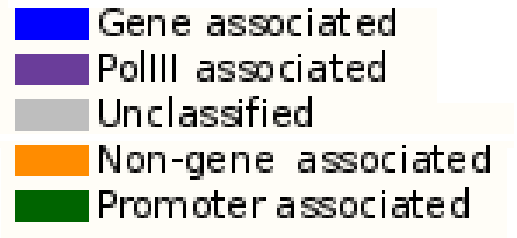
- Identification of candidate regulatory elements (MultiCell)
- Cell specific feature construction
- **Cell specific classification**

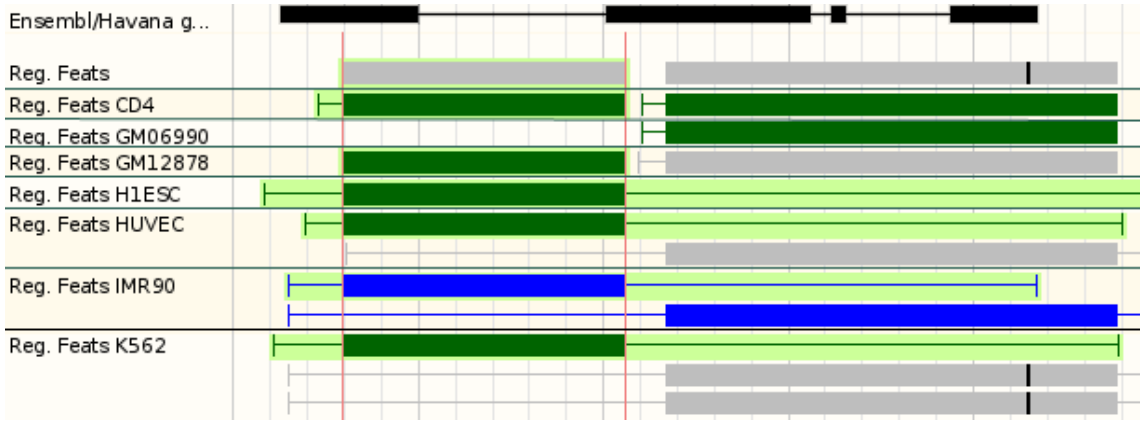
Regulatory Feature Classification

- Cell Specific
- Genomic features:

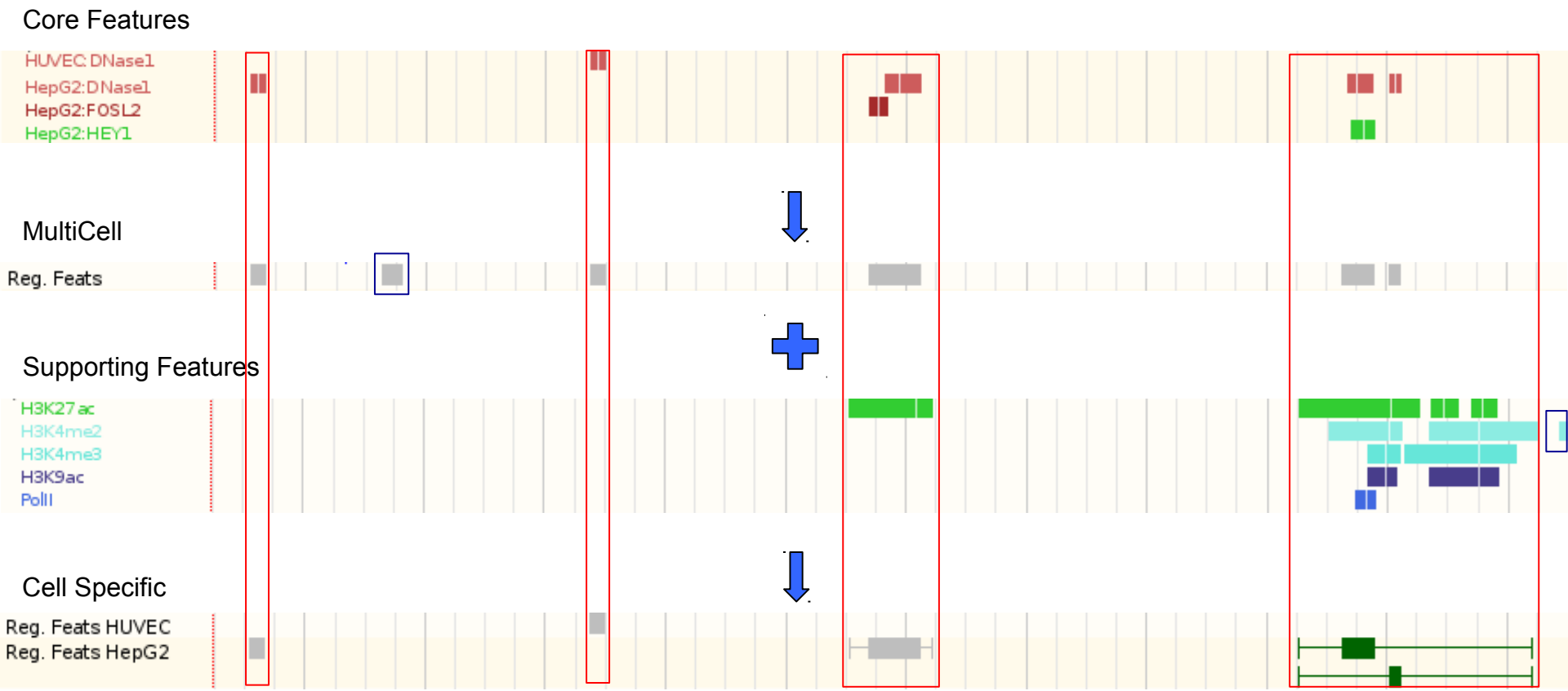
protein_coding_transcript_downstream_2500
protein_coding_single_exon_gene_plus_enhancer
protein_coding_intron1
protein_coding_gene_body
intergenic_2500
RNA_gene_single_exon_gene_plus_enhancer
tss_centred_5000

- Over-represented(χ^2) combinations of marks:
  - e.*g.* H3K4me3 + PolII + H3K36me3 <~> Gene areas





Regulatory Build Summary



Binding Matrices & Motifs

- TF binding affinity represented as a BindingMatrix (PWM)
- Jaspar

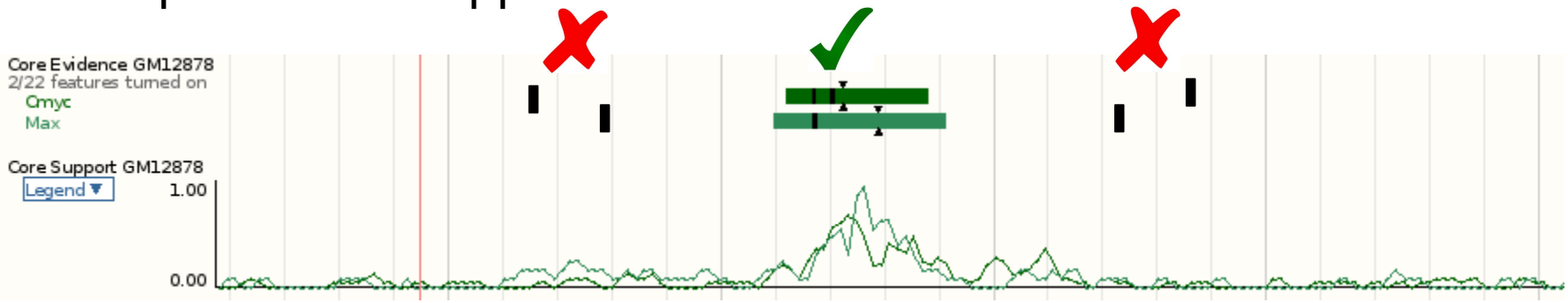
AGATAA
TGATAC
TGATAA
AGATAT
AGATTA
CGATAA
• • •

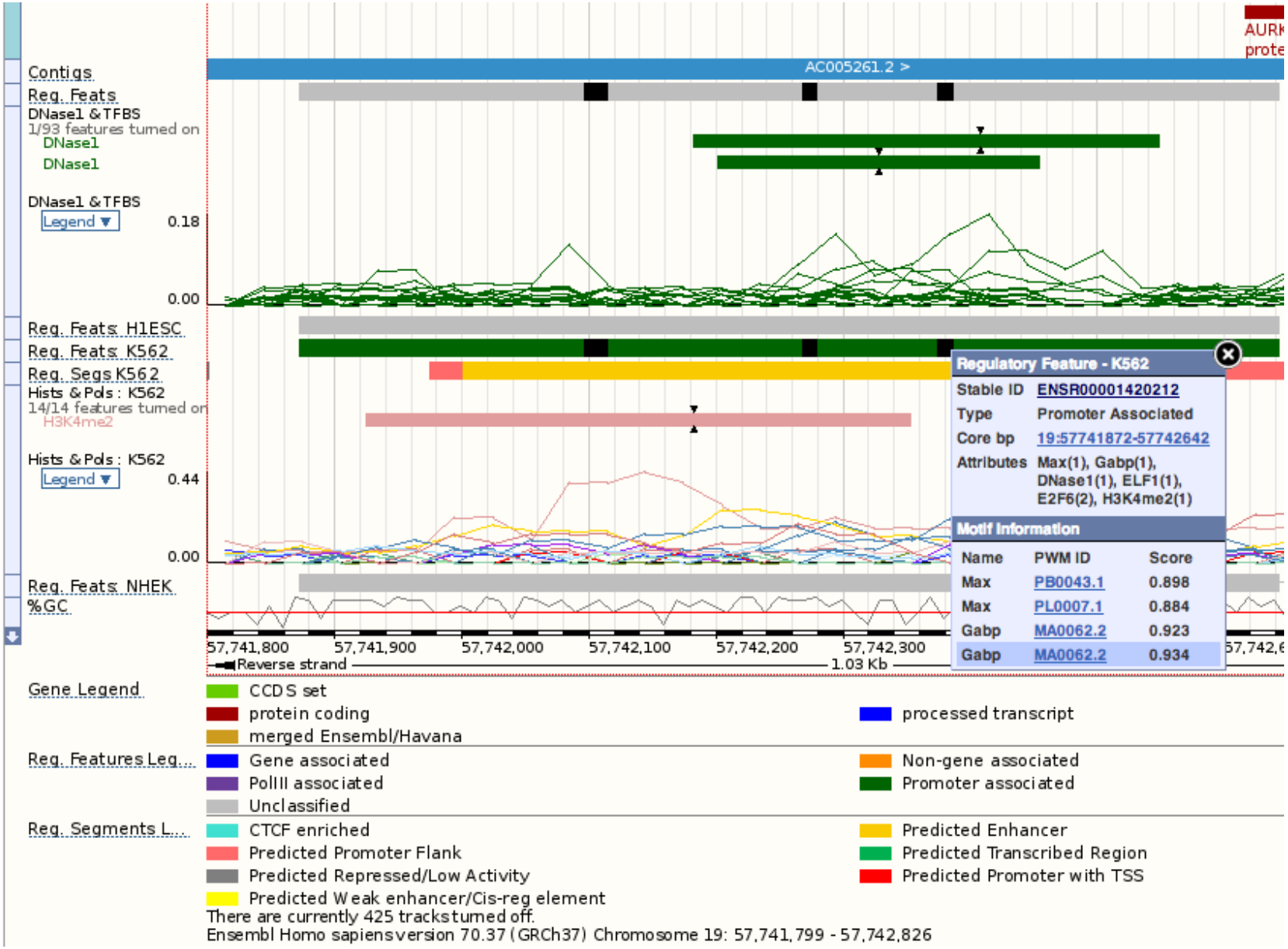
A	10	0	15	0	14	12
C	2	0	0	0	0	2
G	0	15	0	0	0	0
T	3	0	0	15	1	1

Counts Matrix

Logo

- MOODS
- 5% background
- Experimental support

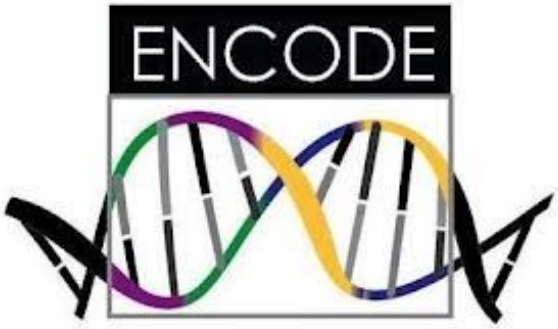




http://www.ensembl.org/Homo_sapiens/Share/322b1ac94b963215225ea6cedbe6fd0f89797976

ENCODE Segmentation

- Functional architecture of the human genome
- HMM derived segmentation:
  - **6 Cell types**: GM12878, K562, H1-hESC, HepG2, HeLa-S3 and HUVEC
  - **14 Assays**: DNAse1; PolII and CTCF; H3k4me1, H3k4me2, H3k4me3, H3k9ac, H3k27ac, H3k27me3, H3k36me3, H4k20me1
- ~ Genomewide coverage basepair resolution.
- Combination two HMM approaches: ChromHMM & Segway
- 7 States (manual labeling):
  - **CTCF** enriched
  - Predicted **Weak Enhancer/Cis-reg element**
  - Predicted **Transcribed Region**
  - Predicted **Enhancer**
  - Predicted **Promoter Flank**
  - Predicted **Repressed/Low Activity**
  - Predicted **Promoter with TSS**
- http://www.ensembl.org/info/docs/funcgen/regulatory_segmentation.html

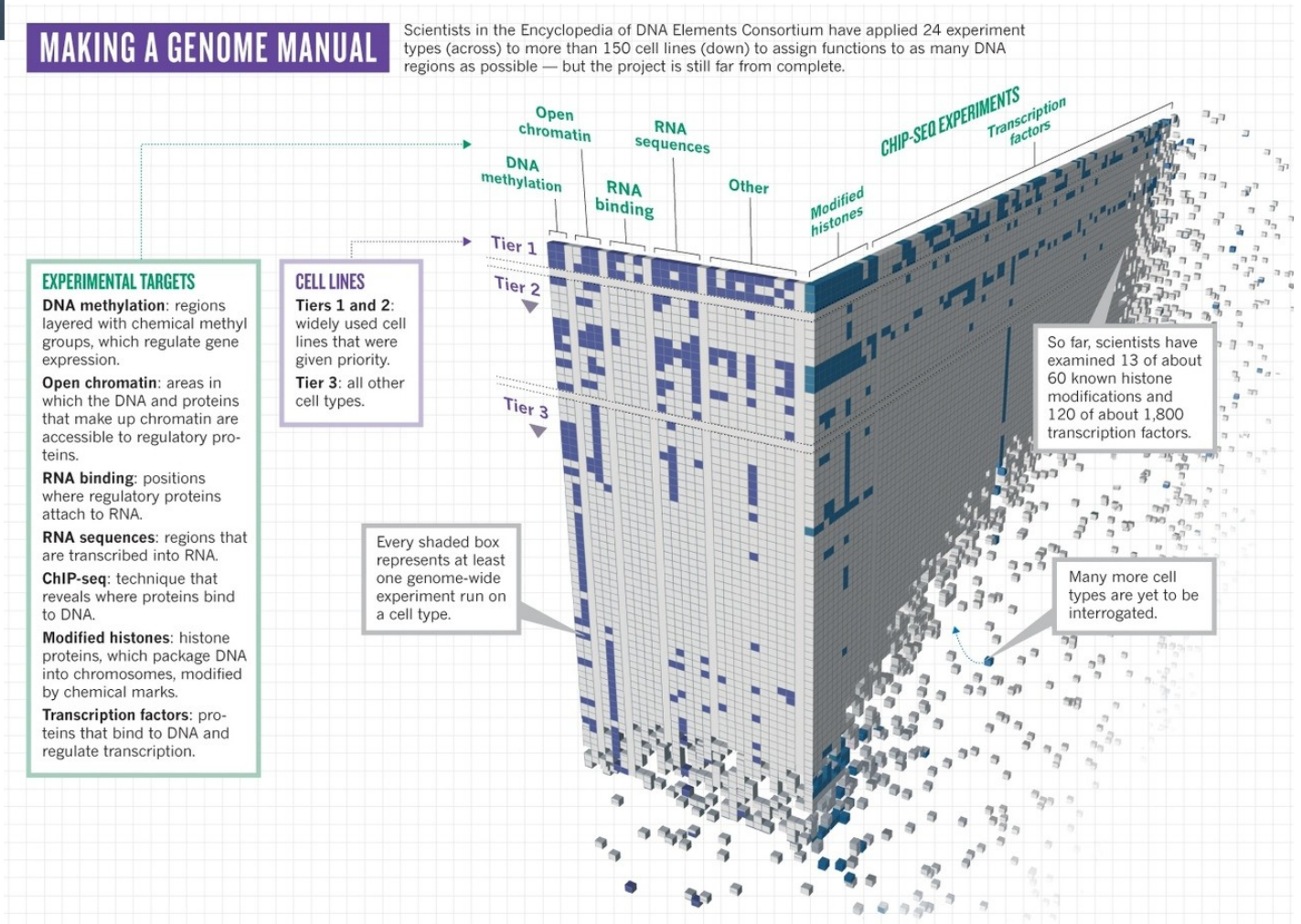


Available data

ROADMAP epigenomics PROJECT

ENCODE

BLUEPRINT epigenome



Data in Ensembl (release 73)

Human

- ~500k Regulatory Features
- ~500 Data sets
- 13 Cell types:
  - H1ESC, NHEK, HUVEC, GM06990, GM12878, K562, IMR90, HepG2, HeLa-S3, CD4, HMEC, HSMM, NH-A.
- 137 Feature types:
  - 91 Transcription Factors
  - 42 Histone Modifications
  - Pol II & III
  - DNase1 & FAIRE

Mouse

- ~200k Regulatory Features
- ~50 Data sets
- 5 Cell types:
  - ES, ESHyb, NPC, MEF, MEL.
- 33 Feature types:
  - 23 Transcription Factors
  - 8 Histone Modifications
  - Pol II
  - DNase1

ROADMAP epigenomics PROJECT

Gene Expression Omnibus

ENCODE

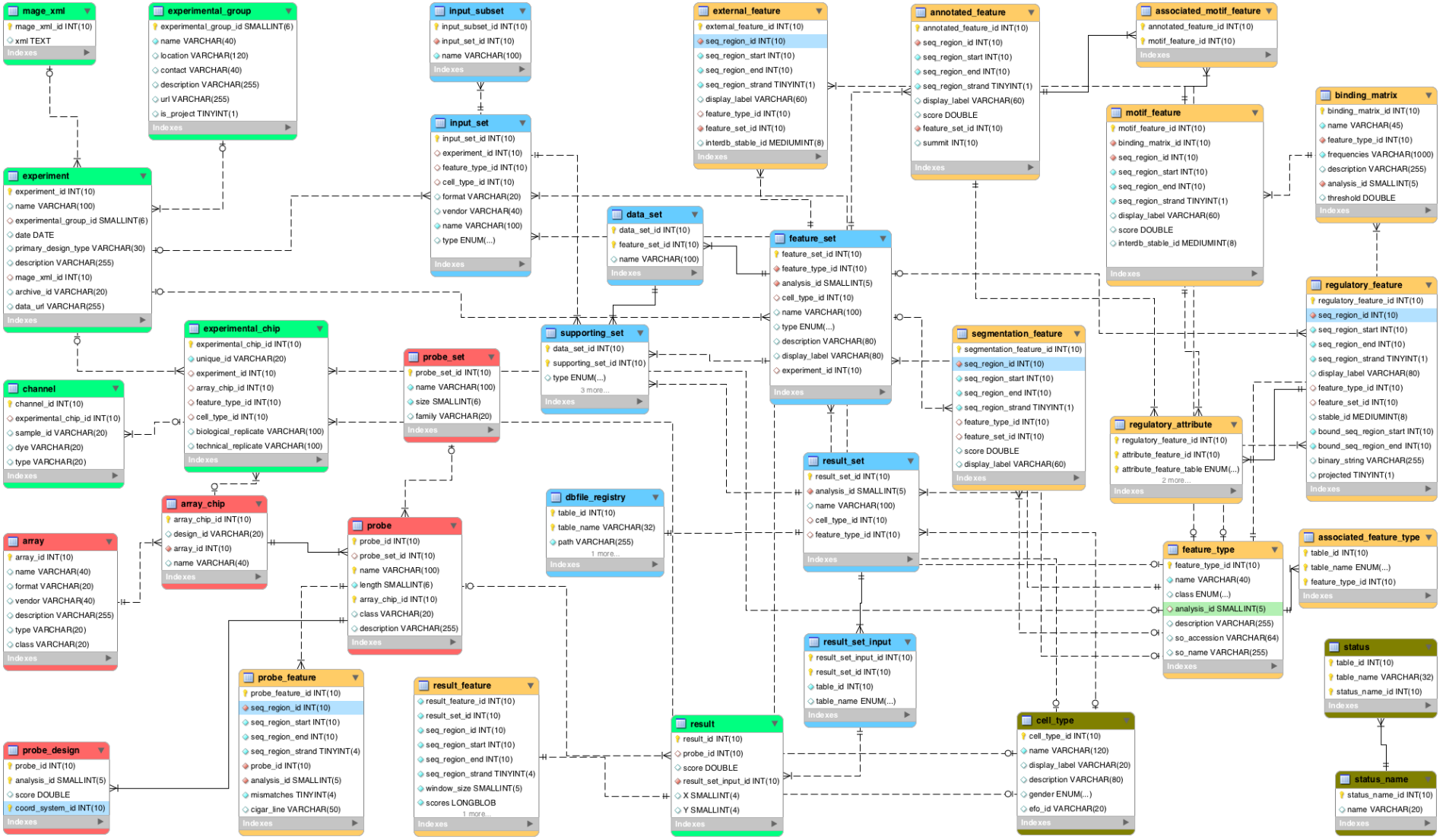
Sequence Read Archive

Questions?

Data Classes

- Metadata
  - **CellType**, **FeatureType**
- ResultSet
  - Raw data: Aligned reads from *-seq,…
- **FeatureSet**: Processed data
  - ChIP peaks (Annotated features)
  - miRanda, cisRED, VISTA enhancers (External features)
  - Regulatory Build (Regulatory features)
- **DataSet**: Links raw and processed data
- **Features:** Annotated, External, Motif, Regulatory, Segmentation

Funcgen DB Schema



http://www.ensembl.org/info/docs/api/funcgen/trimmed_funcgen_schema.png

API - Registry

```
use strict;_
use warnings;_
use Bio::EnsEMBL::Registry;

my $reg = 'Bio::EnsEMBL::Registry';

$reg->load_registry_from_db
   (
    -host => 'ensembldb.ensembl.org',
    -user => 'anonymous'
   );

#Object adaptors create objects, hiding how they are stored
#An object adaptor (RegulatoryFeature Adaptor)
my $rfa = $reg->get_adaptor('human','funcgen','regulatoryfeature');
```

API - Object Adaptors

```
#Registry access

#Meta data
my $cta = $reg->get_adaptor('human', 'funcgen', 'celltype');
my $fta = $reg->get_adaptor('human', 'funcgen', 'featuretype');

#Sets
my $dsa = $reg->get_adaptor('human', 'funcgen', 'dataset');
my $fsa = $reg->get_adaptor('human', 'funcgen', 'featureset');

#Features
my $rfa = $reg->get_adaptor('human', 'funcgen', 'regulatoryfeature');
my $afa = $reg->get_adaptor('human', 'funcgen', 'annotatedfeature');
my $efa = $reg->get_adaptor('human', 'funcgen', 'externalfeature');


#DBAdaptor access
my $db  = $reg->get_DBAdaptor('human', 'funcgen'); #Or via DBAdaptor->new
my $fsa = $db->get_FeatureSetAdaptor;
```

CellType

- Metadata associated to experiments
- Access via CellTypeAdaptor
- Key attributes:

Attribute	Example value(s)	Method(s)
name	HeLa CD4	`$ct->name`
description	Human CD4 T-Cells	`$ct->description`
gender	male	`$ct->gender`

FeatureType

- Metadata associated to experiments
- Describes type of genomic feature
- Access via FeatureTypeAdaptor
- Key attributes/methods:

Attribute	Example value(s)	Method(s)
name	H3K4me3, CTCF DNase1	`$ft->name`
description	Histone 3 Lysine 4 Tri-Methylation	`$ft->description`
class	Histone Regulatory Feature	`$ft->class`

RegulatoryFeatureAdaptor

- Enables access to Regulatory Features:
  - By default it accesses MultiCell Regulatory Features
- Key methods:

Methods	Example arguments	Result
fetch_all_by_Slice	Bio::EnsEMBL::Slice object	MultiCell Regulatory Features
fetch_by_stable_id	String ID: 'ENSR00000000430'	A MultiCell Regulatory Feature
fetch_all_by_stable_ID	String ID: 'ENSR00000000430'	MultiCell & Cell-specific Regulatory Features
fetch_all_by_Slice_FeatureSets	Bio::EnsEMBL::Slice Bio::EnsEMBL::Funcgen::FeatureSet	Regulatory Features specific to Slice and FeatureSets

RegulatoryFeature

- Result of the Regulatory Build Process
- Generic Bio::EnsEMBL::Feature methods:
  - (seq_region_)start/end, strand, slice *etc…*
- Key attributes/methods:

Attribute	Example value(s)	Method(s)
FeatureType	Promoter Associated Gene Associated	`$rf->feature_type->name`
CellType	HeLa-S3 MultiCell	`$rf->cell_type->name`
Bound start/end	5000 - 5800	`$rf->bound_start/end`
Stable ID	ENSR00000000430	`$rf->stable_id`
Regulatory Attributes	List of underlying supporting features	`$rf->regulatory_attributes('annotated')`

Example: MultiCell Re

```
my $regfeat_adaptor =_
  $reg->get_adaptor('Human', 'f

my $slice =_
  $slice_adaptor->fetch_by_regi

my @reg_feats =_
  @{$regfeat_adaptor->fetch_all

#These Features are global 'Mul
foreach my $rf (@reg_feats){_
  print "Stable ID: ".$rf->stab
  print "\tLocation: ".$rf->seq
    $rf->seq_region_start."-".$

  print "\tCell: ".$rf->cell_ty
  print "\tFeature Type: ".$rf-
}
```

```
Stable ID: ENSR00000536878
 Location: 1:54964217-54964448
 Cell: MultiCell
 Feature Type: Unclassified

Stable ID: ENSR00001037991
 Location: 1:54964669-54964987
 Cell: MultiCell
 Feature Type: Unclassified

Stable ID: ENSR00000165384
 Location: 1:54965058-54965738
 Cell: MultiCell
 Feature Type: Unclassified

Stable ID: ENSR00000282669
 Location: 1:54968623-54969107
 Cell: MultiCell
 Feature Type: Unclassified

Stable ID: ENSR00000536879
 Location: 1:54972264-54972681
 Cell: MultiCell
 Feature Type: Unclassified

Stable ID: ENSR00001520371
 Location: 1:54976416-54976742
 Cell: MultiCell
```

Example: MultiCell Regulatory Features

```
my $regfeat_adaptor =_
  $reg->get_adaptor('Human', 'funcgen', 'regulatoryfeature');

my $slice =_
  $slice_adaptor->fetch_by_region( 'chromosome',
                                   1,54960000, 54980000 );_
my @reg_feats =_
  @{$regfeat_adaptor->fetch_all_by_Slice($slice)};_

#These Features are global 'MultiCell' Regulatory Features._
foreach my $rf (@reg_feats){_
  print "Stable ID: ".$rf->stable_id."\n";_
  print "\tLocation: ".$rf->seq_region_name.":".
    $rf->seq_region_start."-".$rf->seq_region_end."\n";

  print "\tCell: ".$rf->cell_type->name."\n";_
  print "\tFeature Type: ".$rf->feature_type->name."\n";_
}
```

Example: Cell-specific RegulatoryFeature details

```
my $rfs =
  $regfeat_adaptor->fetch_all_by_stable_ID('ENSR00000165384');

foreach my $cell_rf (@{$rfs}){
  #The stable id will always be 'ENSR00000165384'
  print $cell_rf->stable_id.": \n";

  #But now it will be for a specific cell type
  print "\tCell: ".$cell_rf->cell_type->name."\n";

  #It will also contain cell-specific annotation
  print "\tType: ".$cell_rf->feature_type->name."\n";

  #And cell-specific extra boundaries
  print "\t".$cell_rf->seq_region_name.":".
    $cell_rf->bound_start."..".$cell_rf->start."-".
      $cell_rf->end."..". $cell_rf->bound_end."\n\n";
}
```

Example: Cell-specific RegulatoryFeature details

```
my $rfs =_
   $regfeat_adaptor->fetch_all_by.

foreach my $cell_rf (@{$rfs}){_
   #The stable id will always be
   print $cell_rf->stable_id.": \

   #But now it will be for a spec.
   print "\tCell: ".$cell_rf->cel.

   #It will also contain cell-spe
   print "\tType: ".$cell_rf->fea

   #And cell-specific extra bound.
   print "\t".$cell_rf->seq_regio
     $cell_rf->bound_start."..".$
       $cell_rf->end."..". $cell_
}
```

```
ENSR00000165384:
        Cell: K562
        Type: Promoter Associated
        1:54962350..54965058-54965738..54968000

ENSR00000165384:
        Cell: H1ESC
        Type: Unclassified
        1:54963500..54965058-54965738..54967100

ENSR00000165384:
        Cell: HUVEC
        Type: Unclassified
        1:54959700..54965058-54965738..54969400

ENSR00000165384:
        Cell: HMEC
        Type: Promoter Associated
        1:54964780..54965058-54965738..54965833

ENSR00000165384:
        Cell: CD4
        Type: Promoter Associated
        1:54964840..54965058-54965738..54965738

ENSR00000165384:
        Cell: HepG2
```

SegmentationFeatureAdaptor

- Enables access to Segmentation Features
- Key methods:

Methods	Example arguments	Result
fetch_all_by_Slice	Bio::EnsEMBL::Slice object	Mixed CellType SegmentationFeatures
fetch_all_by_Slice_FeatureSets	Bio::EnsEMBL::Slice Bio::EnsEMBL::Funcgen::FeatureSet	SegmentationFeatures specific to Slice and FeatureSets

SegmentationFeature

- Result of the merged ENCODE segmentation.
- Generic Bio::EnsEMBL::Feature methods:
  - (seq_region_)start/end, strand, slice *etc…*
- Key attributes/methods:

Attribute	Example value(s)	Method(s)
FeatureType	CTCF enriched Predicted Enhancer	`$seg_feat->feature_type->name`
CellType	HeLa-S3	`$seg_feat->cell_type->name`

Exercises

Now try the ‘Regulatory Features’ questions here!

Tips

Q: Seen some code in the presentation you want to use? Can’t copy it as it’s an image?

A: I’m not *that* cruel, look in the **notes** section for that slide and you’ll find the text!

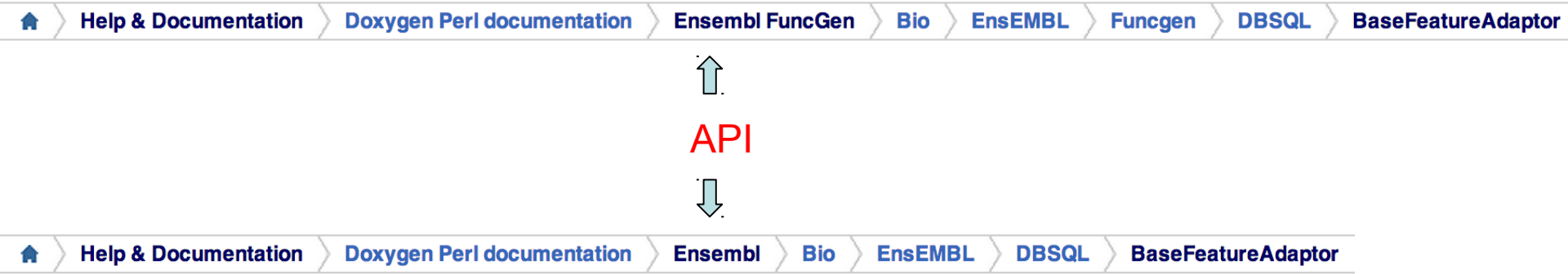
Q: Want to see what’s in the database but not sure which method to use?

A: **fetch_all** works with most adaptors. Try it and print out some details.

Just be mindful how much data you might be fetching i.e. FeatureSets ~ 500…okay, Genes ~24k…very slow, Variations…don’t bother!

Q: Getting lost in the Doxygen docs?

A: Look at the ‘**breadcrumbs**’ menu at the top of the page to see if you are in the right API section.



Exercise: Regulatory Features

Regulatory Features

Regulatory Features are regions in the genome for which there is some experimental evidence indicating a potential role in the regulation of gene expression.

1. Regulatory Features: cell type specific data

Using the human DB, fetch the all the cell type specific regulatory features with stable ID 'ENSR00000623613'.

Print out the stable ID, bound_start/end and start/end values, name of the cell and feature type for each.

HINT: To get all the cell type specific RegulatoryFeatures use the *fetch_all_by_stable_ID* method from the *RegulatoryFeatureAdaptor*.

General information about Ensembl stable IDs

2. Regulatory Features: What RegulatoryFeatures are near the oncogene BRCA2?

Create a script which fetches all the RegulatoryFeatures within 1KB of the BRCA2 gene.

Print out their stable IDs, bound_start/end and start/end values, name of the cell and feature types.

HINT: Use fetch_all_by_external_name with 'BRCA2' to get the gene object from the core API. This will return a copy of the gene on a chromosome and on an LRG. Use the following code to grab the right one:

```
my $gene = (grep {$_->slice->coord_system_name eq 'chromosome'} @genes )[0];
```

HINT: Look at the arguments for fetch_by_gene_stable_id, or use the Gene->feature_Slice method and Slice->expand methods.

Exercise: Regulatory Features

3. Regulatory Features: associated experimental evidence

Now fetch just the ENSR00000623613 MultiCell feature.

Print out the display_label, start/end values of all the underlying supporting (evidence) features.

Compare with the start/end values of the regulatory feature itself.

HINT: By default the fetch_by_stable_id method returns just the MultiCell features.

4. ENCODE Segmentation

Now using the human RegulatoryFeature ENSR00001348194, fetch all the cell type specific classifications (similar to Q1).

Also fetch all the ENCODE segmentation features for that region by using the feature_Slice of one of the RegulatoryFeatures.

For each cell type, list the RegulatoryFeature classifications and compare to the SegmentatioFeature classifications.

HINT: You will need to cache the features based on cell type to print them out in a sensible order.

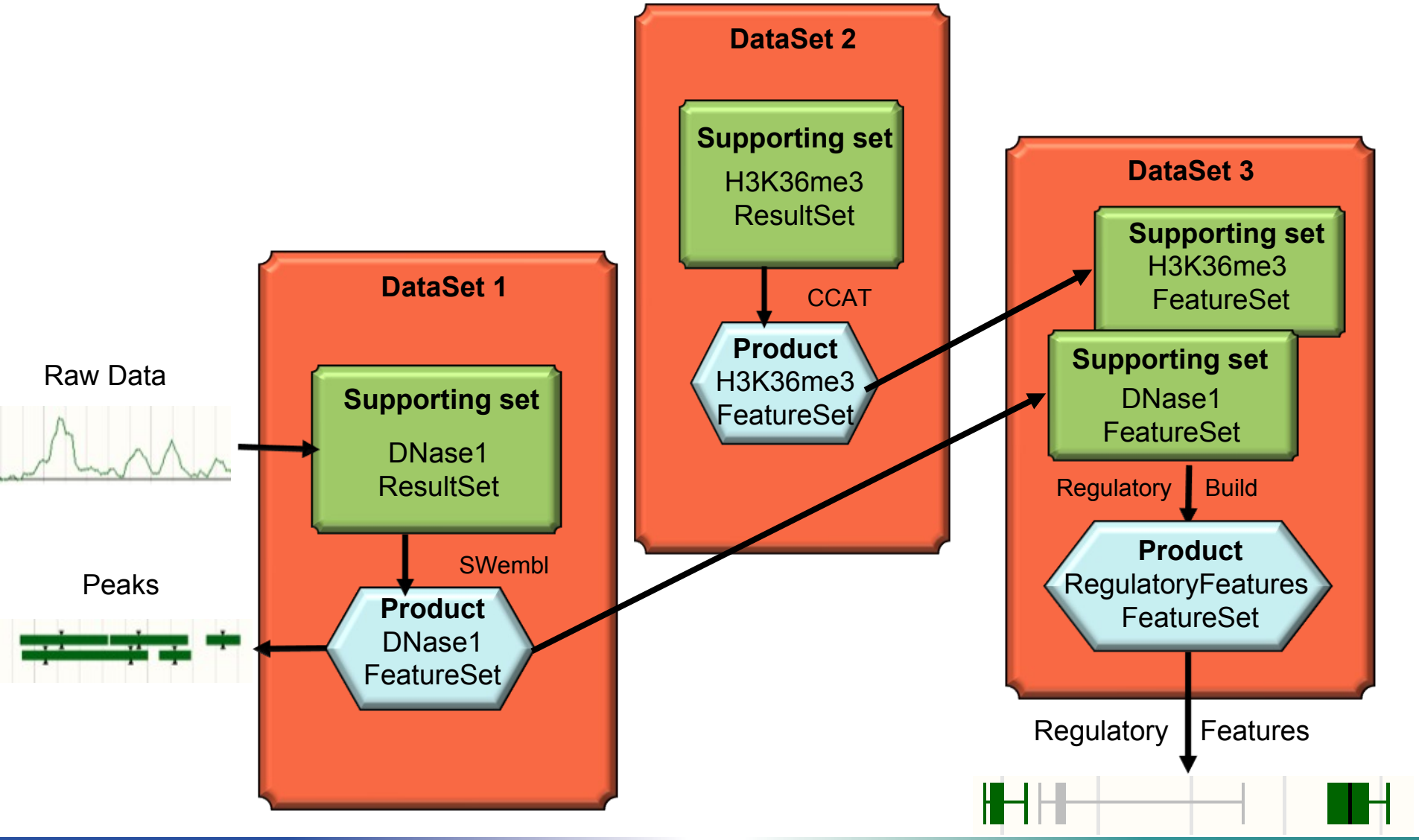
HINT: Not all cell types have segmentation features.

DataSet

- High level ‘container’ object
  - Fetched with DataSetAdaptor
- Links raw data to processed data
- Key attributes/methods:

Attribute	Example value(s)	Method(s)
Name	RegulatoryFeatures:MultiCell	`$ds->name`
Product feature set	FeatureSet Object	`$ds->product_FeatureSet`
Supporting Sets	List of supporting Set objects e.g. ResultSet or FeatureSet	`$ds->supporting_sets`

Bio::EnsEMBL::Funcgen Sets



FeatureSetAdaptor

- Provides fetch methods for FeatureSet objects:
- Key methods:

Methods	Example arguments
fetch_by_name	'cisRED motifs'
fetch_all_by_CellType	CellType object
fetch_all_by_FeatureType	FeatureType object
fetch_all_by_feature_class	external, annotated or regulatory

FeatureSet

- Processed Feature container:
  - Regulatory, Annotated, External, Segmentation
- Key attributes/methods:

Attributes	Example value(s)	Method(s)
Name	HepG2_USF1_ENCODE_Hudsonalpha_SWEMBL_R015	`$fs->name`
Display label	USF1 - HepG2 Enriched Sites	`$fs->display_label`
Cell Type (*)	K562	`$fs->cell_type->name (*)`
Feature Type	CTCF	`$fs->feature_type->name`
Features	List of Features e.g. Annotated or RegulatoryFeatures	`$fs->get_Features_by_Slice` `$fs->get_all_Features`

* : Some Feature Sets may not have metadata like cell type

Example: Get all TFBS FeatureSets

```
my $fset_adaptor =
  $reg->get_adaptor('Human', 'funcgen', 'featureset');

my @tfs =
  @{$ftype_adaptor->fetch_all_by_class('Transcription Factor')};

foreach my $ft (@tfs){
  my @fsets = @{$fset_adaptor->fetch_all_by_FeatureType($ft)};
  print "Found ".scalar(@fsets).' '.$ft->name." FeatureSets:\n";

  foreach my $fset (@fsets){
    print "\t".$fset->name." - ";
    print "\t".$fset->display_label."\n";
    print "\t\t".$fset->cell_type->name;
    print "\t".$fset->feature_type->name;
    print "\t".scalar(@{$fset->get_all_Features})." AnnotatedFeatures\n";
  }
}
```

Example: Get all TFBS FeatureSets

```
my $fset_adaptor =
  $reg->get_adaptor

my @tfs =
  @{$ftype_adaptor-

foreach my $ft (@tf
  my @fsets = @{$fs
  print "Found ".sc

  foreach my $fset
    print "\t".$fse
    print "\t".$fse
    print "\t\t".$f
    print "\t".$fse
    print "\t".scal
  }
}
```

```
Found 29 CTCF FeatureSets
    Nessie_NG_STD_2_ctcf_ren_BR1 -  CTCF - IMR90 Enriched Sites
        IMR90   CTCF    43427 AnnotatedFeatures
    K562_CTCF_ENCODE_Broad_SWEmbl_R015_D150 -       CTCF - K562 Enriched Sites
        K562    CTCF    32604 AnnotatedFeatures
    CD4_CTCF_BarskiZhao_PMID17512414_SWEmbl_R015_D150 -     CTCF - CD4 Enriched Site
        CD4     CTCF    25804 AnnotatedFeatures
    HepG2_CTCF_ENCODE_Uta_SWEmbl_R015_D150 -        CTCF - HepG2 Enriched Sites
        HepG2   CTCF    40910 AnnotatedFeatures
    GM12878_CTCF_ENCODE_Broad_SWEmbl_R015_D150 -    CTCF - GM12878 Enriched Sites
        GM12878 CTCF    24837 AnnotatedFeatures
    GM12878_CTCF_ENCODE_Uta_SWEmbl_R015_D150 -      CTCF - GM12878 Enriched Sites
        GM12878 CTCF    33841 AnnotatedFeatures
    NHEK_CTCF_ENCODE_Broad_SWEmbl_R015_D150 -       CTCF - NHEK Enriched Sites
        NHEK    CTCF    36209 AnnotatedFeatures
    HeLa-S3_CTCF_ENCODE_Uta_SWEmbl_R015_D150 -      CTCF - HeLa-S3 Enriched Sites
        HeLa-S3 CTCF    42904 AnnotatedFeatures
    H1ESC_CTCF_ENCODE_Broad_SWEmbl_R015_D150 -      CTCF - H1ESC Enriched Sites
        H1ESC   CTCF    38179 AnnotatedFeatures
    HUVEC_CTCF_ENCODE_Broad_SWEmbl_R015_D150 -      CTCF - HUVEC Enriched Sites
        HUVEC   CTCF    31243 AnnotatedFeatures
    HepG2_CTCF_ENCODE_Uw_SWEmbl_R015_D150 -         CTCF - HepG2 Enriched Sites
        HepG2   CTCF    34242 AnnotatedFeatures
    GM06990_CTCF_ENCODE_Uw_SWEmbl_R015_D150 -       CTCF - GM06990 Enriched Sites
        GM06990 CTCF    31001 AnnotatedFeatures
    K562_CTCF_ENCODE_Uta_SWEmbl_R015_D150 -         CTCF - K562 Enriched Sites
        K562    CTCF    42393 AnnotatedFeatures
    HeLa-S3_CTCF_ENCODE_Uw_SWEMBL_R015 -    CTCF - HeLa-S3 Enriched Sites
        HeLa-S3 CTCF    33119 AnnotatedFeatures
    HUVEC_CTCF_ENCODE_Uw_SWEMBL_R015 -      CTCF - HUVEC Enriched Sites
        HUVEC   CTCF    36709 AnnotatedFeatures
    K562_CTCF_ENCODE_Uw_SWEMBL_R015 -       CTCF - K562 Enriched Sites
        K562    CTCF    28255 AnnotatedFeatures
    NHEK_CTCF_ENCODE_Uw_SWEMBL_R015 -       CTCF - NHEK Enriched Sites
```

More Exercises

Now try the ‘Sets’ questions here!

Tips

Q: Seen some code in the presentation you want to use? Can’t copy it as it’s an image?

A: I’m not *that* cruel, look in the **notes** section for that slide and you’ll find the text!

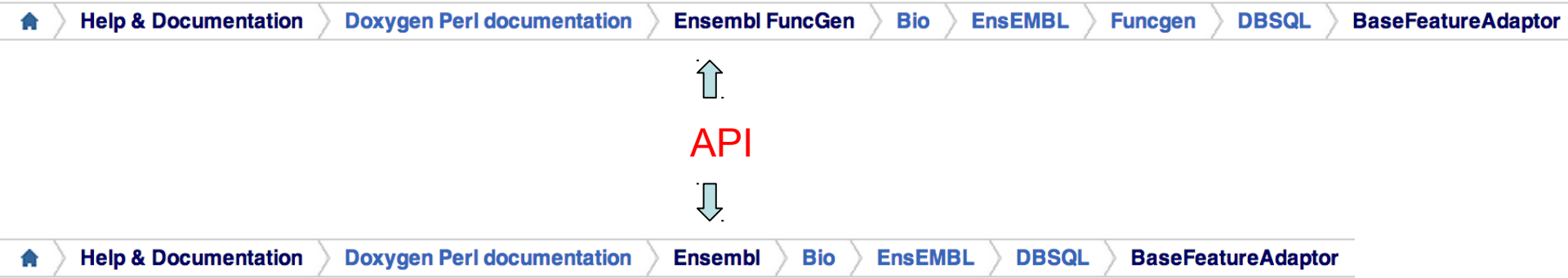
Q: Want to see what’s in the database but not sure which method to use?

A: **fetch_all** works with most adaptors. Try it and print out some details.

Just be mindful how much data you might be fetching i.e. FeatureSets ~ 500…okay, Genes ~24k…very slow, Variations…don’t bother!

Q: Getting lost in the Doxygen docs?

A: Look at the ‘**breadcrumbs**’ menu at the top of the page to see if you are in the right API section.



Exercise: Set questions

1. DataSets

DataSets are containers associating the data obtained by an analysis (FeatureSets) to the underlying raw data (ResultSets).

Create a script which fetches all available DataSets for Human.

How many are there?

Now get the 'RegulatoryFeatures:MultiCell' data set and print the display label of the product feature set and all the supporting sets.

HINT: Use the DataSetAdaptor methods.

2. FeatureSets

FeatureSets hold processed data or features *i.e.* peak calls or the output of a high level analysis *e.g.* the Regulatory Build.

Print the name of the feature sets for the Human 'GM12878' cell type.

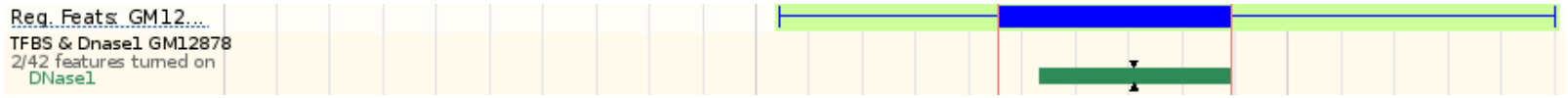
Print the name of the feature sets for the Human 'CTCF' feature type.

Is the Human FeatureSet 'VISTA enhancer set' associated to any cell type or feature type?

Trick question: Get the supporting data for the VISTA FeatureSet.

HINT: Most adaptors have a fetch_by_name method
HINT: DataSetAdaptor->fetch_by_product_FeatureSet will fetch the DataSet containing the supporting/raw data for a FeatureSet.

AnnotatedFeature



- Simple ‘Processed’ Feature
  - *i.e.* a peak call from ChIP-Seq
  - Also inherits methods from Feature
- Used as evidence for Regulatory Features. Fetched via:
  - $regulatory_feature->regulatory_attributes(‘annotated’)
- Some key attributes/methods:

Method	Example value(s)	Code
display_label	CTCF – IMR90 Enriched Site	`$af->display_label`
score	Value	`$af->score`
summit	Coordinate	`$af->summit`
get_associated_MotifFeatures	List of MotifFeature objects	`$af->get_associated_MotifFeatures`

Example: What AnnotatedFeatures support this RegulatoryFeature?

```
#This gets the 'MultiCell' Regulatory Feature
my $rf =
  $regfeat_adaptor->fetch_by_stable_id('ENSR00000165384');

my @annotated_features =
  @{$rf->regulatory_attributes('annotated')};

#Get annotated features supporting a regulatory feature
foreach my $annotated_feature (@annotated_features) {
  print $annotated_feature->display_label."\t";
  print $annotated_feature->score."\t";
  print $annotated_feature->summit."\n";
}
```

Example: What AnnotatedFeatures support this RegulatoryFeature?

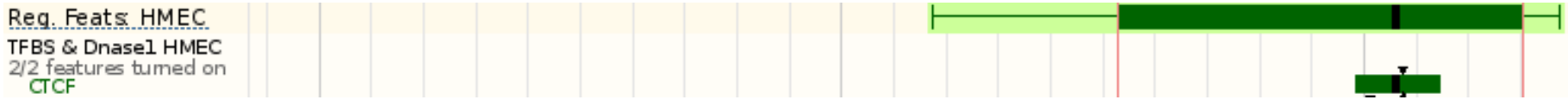
```
#This gets the 'Mul
my $rf =_
   $regfeat_adaptor-

my @annotated_featu
   @{$rf->regulatory

#Get annotated feat
foreach my $annotat
   print $annotated_
   print $annotated_
   print $annotated_
}
```

```
DNase1 - HUVEC Enriched Site    40.489637    54965202
DNase1 - K562 Enriched Site     27.862417    54965209
DNase1 - H1ESC Enriched Site    26.207104    54965236
CTCF - HUVEC Enriched Site      125.020915   54965284
DNase1 - HSMM Enriched Site     17.812017    54965234
DNase1 - H1ESC Enriched Site    107.692877   54965254
CTCF - HMEC Enriched Site       84.500112    54965296
CTCF - CD4 Enriched Site        12.242441    54965282
CTCF - HepG2 Enriched Site      36.243751    54965285
CTCF - NHEK Enriched Site       156.68494    54965300
DNase1 - K562 Enriched Site     19.889997    54965289
CTCF - HUVEC Enriched Site      47.901836    54965291
Rad21 - K562 Enriched Site      33.054809    54965299
CTCF - K562 Enriched Site       29.488397    54965300
CTCF - NHEK Enriched Site       21.50665     54965304
CTCF - H1ESC Enriched Site      22.526458    54965314
CTCF - NHEK Enriched Site       84.547054    54965322
CTCF - K562 Enriched Site       85.822038    54965326
DNase1 - NHEK Enriched Site     26.878816    54965373
DNase1 - HepG2 Enriched Site    17.995201    54965390
```

MotifFeature



- BindingMatrix (PWM) match
  - Also inherits generic Feature methods (eg. seq)
- Fetched via:
  - $regulatory_feature->regulatory_attributes('motif')
  - $annotated_feature->get_associated_MotifFeatures()
- Some Key attributes / methods:

Method	Example value(s)	Code
display_label	Egr1:PB0010.1	`$mf->display_label()`
binding_matrix	MA0060.1	`$mf->binding_matrix->name()`
score	0.81	`$mf->score()`
associated_annotated_features	List of AnnotatedFeature objects	`$mf->associated_annotated_features()`

Example: What MotifFeatures support this RegulatoryFeature?

```
#This gets the 'MultiCell' Regulatory Feature
my $rf =
     $regfeat_adaptor->fetch_by_stable_id('ENSR00000165384');

my @motif_features = @{$rf->regulatory_attributes('motif')};

#Get motif features supporting a regulatory feature
foreach my $motif_feature (@motif_features) {
   print $motif_feature->display_label."\t";
   print $motif_feature->binding_matrix->name."\t";
   print $motif_feature->score."\n";

   foreach my $ann_feat(@{$motif_feature->associated_annotated_features()})
     {
       print $ann_feat->display_label."\t";
       print $ann_feat->score."\t";
       print $ann_feat->summit."\n";
     }
}
```

Example: What MotifFeatures support this RegulatoryFeature?

```
#This gets the 'MultiCell' Regulatory Feature
my $rf =
    $regfeat_adapt

my @motif_features

#Get motif feature
foreach my $motif_
  print $motif_fea
  print $motif_fea
  print $motif_fea

  foreach my $ann_
    {
      print $ann_f
      print $ann_f
      print $ann_f
    }
}
```

```
CTCF:MA0139.1   MA0139.1         0.885
CTCF - HUVEC Enriched Site      125.020915      54965284
CTCF - HMEC Enriched Site       84.500112       54965296
CTCF - CD4 Enriched Site        12.242441       54965282
CTCF - HepG2 Enriched Site      36.243751       54965285
CTCF - NHEK Enriched Site       156.68494       54965300
CTCF - HUVEC Enriched Site      47.901836       54965291
CTCF - K562 Enriched Site       29.488397       54965300
CTCF - NHEK Enriched Site       21.50665        54965304
CTCF - H1ESC Enriched Site      22.526458       54965314
CTCF - NHEK Enriched Site       84.547054       54965322
CTCF - K562 Enriched Site       85.822038       54965326
```

Yet More Exercises

Now try the ‘Other Features’ questions here!

Tips

Q: Seen some code in the presentation you want to use? Can’t copy it as it’s an image?

A: I’m not *that* cruel, look in the **notes** section for that slide and you’ll find the text!

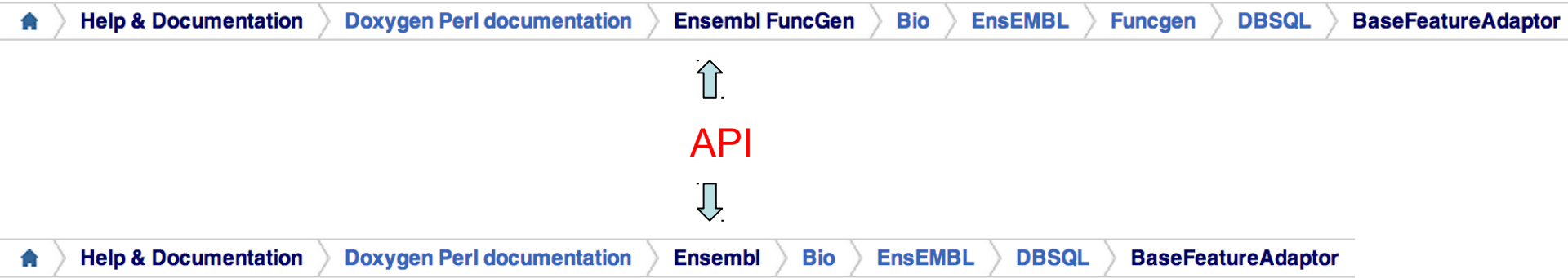
Q: Want to see what’s in the database but not sure which method to use?

A: **fetch_all** works with most adaptors. Try it and print out some details.

Just be mindful how much data you might be fetching i.e. FeatureSets ~ 500…okay, Genes ~24k…very slow, Variations…don’t bother!

Q: Getting lost in the Doxygen docs?

A: Look at the ‘**breadcrumbs**’ menu at the top of the page to see if you are in the right API section.



Feature Exercises

1. Annotated Features

Annotated Features represents the results of an analysis of raw or processing signal data. These correspond to regions in the genome enriched for specific events (like TF binding or Histone Marks) *i.e.* they are 'peak calls'.

Fetch the AnnotatedFeatures in the region Y:5000000-40000000 for the Human FeatureSets with name:
K562_DNase1_ENCODE_Duke_SWEmbl_R0025_D150
HepG2_DNase1_ENCODE_Duke_SWEmbl_R0025_D150

Print the number of features returned by each, the details of the CellType (e.g. gender) associated with the FeatureSet and the details of the few features including the 'summit'.

What are the differences and why?

Optional: Print the properties(logic_name, display_label, parameters) of the Analysis used in the previous feature sets.

2. Motif Features

Motif features represent putative binding sites based on alignments of PWMs from JASPAR. MotifFeatures are always associated to AnnotatedFeatures representing Transcription Factor (TF) Binding. More information about how we integrate these into the regulatory build process can be found here.

Get the 'motif' regulatory attributes associated to the Human Regulatory Feature 'ENSR00001227187'. Print their properties.
Hint: use 'motif' as a parameter for regulatory_attributes.

Print the properties of the annotated features associated to the motif feature.

3. Binding Matrices and motif strength

Each MotifFeature is associated with a PWM, which are represented by the 'BindingMatrix' class. The MotifFeature score represents the relative binding affinity with respect to the PWM defined in the BindingMatrix.

Using the Motif feature obtained in exercise 2, get the associated Binding Matrix and print some details.

Check potential effect of changes in the sequence of the motif feature on the relative strength of that motif feature.

Optional: Check the CEBP conservation score along the motif. Compare with the JASPAR matrix.

Getting More Information

- Funcgen API tutorial – including microarray annotationshttp://www.ensembl.org/info/docs/api/funcgen/regulation_tutorial.html
- Funcgen overview:
  http://www.ensembl.org/info/docs/api/funcgen/index.html
- Regulatory Build info:http://www.ensembl.org/info/docs/funcgen/index.html
- Experimental Data Sourceshttp://www.ensembl.org/Homo_sapiens/Experiment?db=core;ex=all
- helpdesk@ensembl.org
- dev@ensembl.org

Ensembl Regulation Team

Daniel Zerbino

Nathan Johnson

Thomas Juettemann

Steve Wilder

Ensembl Acknowledgements

Ensembl Regulation Team

- Nathan Johnson
- Daniel Zerbino
- (Steven Wilder)
- Thomas Juettemann

- Paul Flicek, Steve Searle and the Entire Ensembl Team

Funding

wellcome trust

EMBL

National Human Genome Research Institute

BBSRC bioscience for the future

European Commission Framework Programme 7

SEVENTH FRAMEWORK PROGRAMME

Quantomics
From Sequence to Consequence : Tools for the Exploitation of Livestock Genomes

GEN2PHEN

BLUEPRINT epigenome