



# Introduction to the Sequence Ontology

SOFG

October 2004

Karen Eilbeck

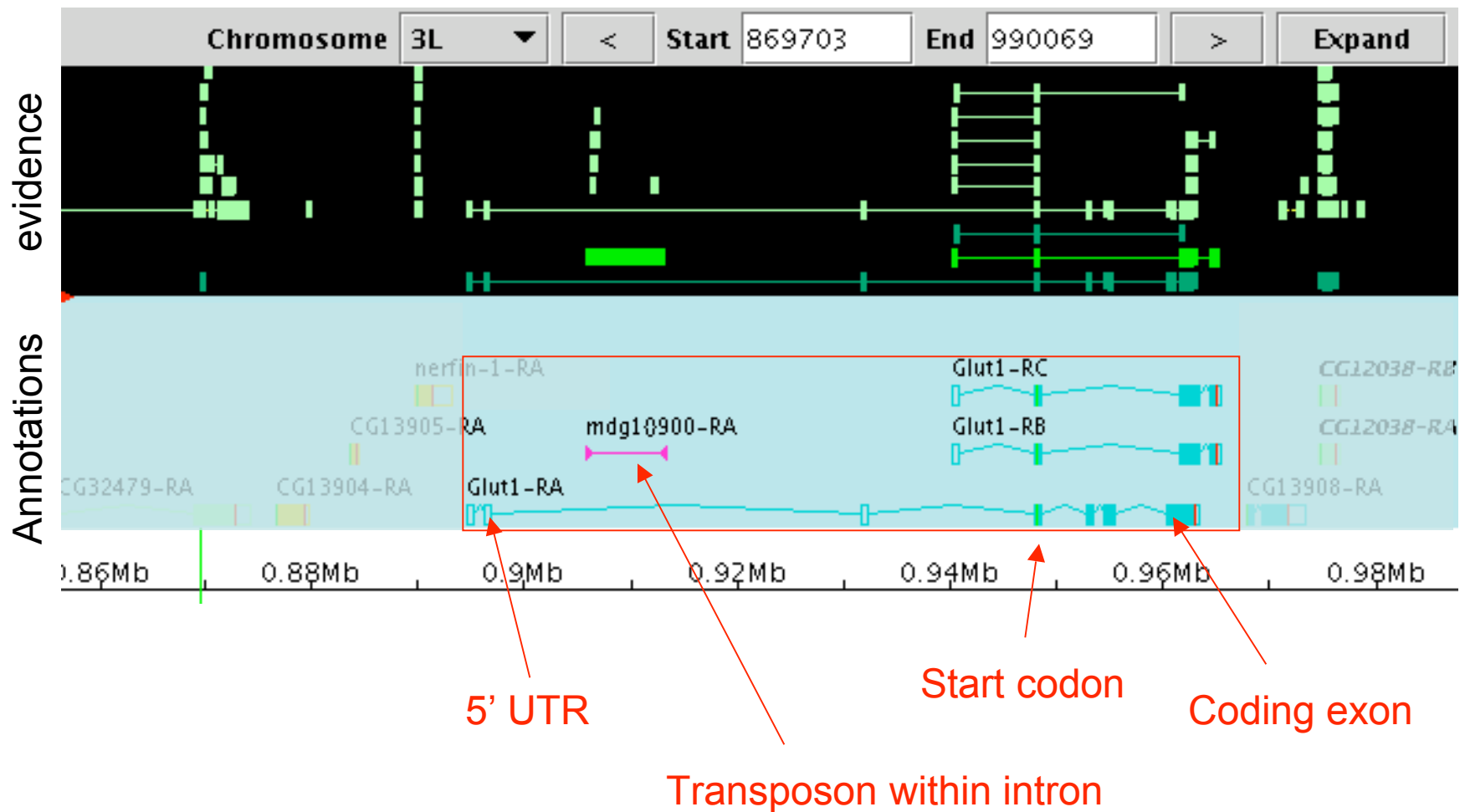
Berkeley Drosophila Genome Project

# Questions about SO

- What is sequence annotation and why does it need structure? Why aren't we happy with what exists now?
- What exactly is SO?
- What can I do with SO?
- Who is using SO, and what formats support it?
- Where can I get it?

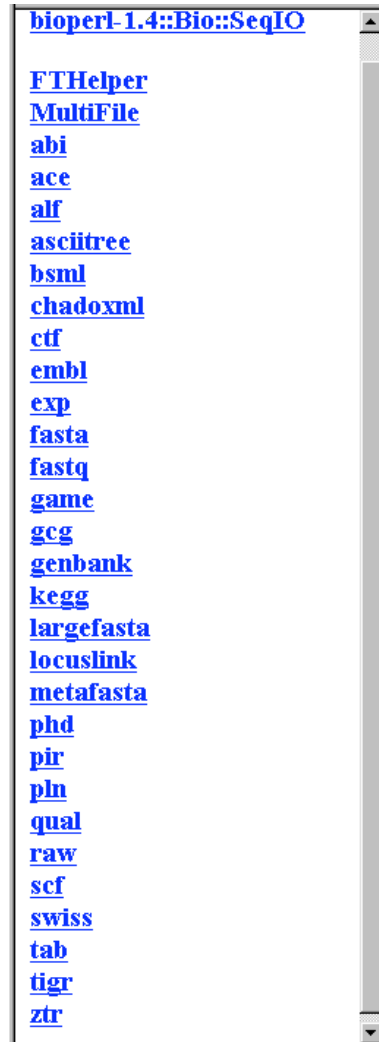
# Annotation = knowledge

## 3 Alternate transcripts of Glut1 gene



# Sequence annotations come in many formats from many sources

- Formats



- Sources

- Model Organism DB
  - FlyBase
  - WormBase
- Sequencing Centers
  - TIGR
  - JGI
- Genome Collections
  - GenBank
  - EMBL
  - DDJB
- Mirror sites

# What does this mean?

- You can not get all sequences from the same place.
- You can not get all sequences in the same format, using the same terminology.
- Even if you are using the same data exchange format as another group it may not adhere to the same data model.
- Data exchange can be hard work.

# Where did SO come from?

- The model organism community wanted a way to unify how they annotate genomes.
- Group of scientists from BDGP, FlyBase, WormBase, MGI, Ensembl got together and drafted a first version of SO.



# The aims of SO

- To unify the description of sequence annotations
  - Standardize the vocabulary we use to describe biological sequence
  - Facilitate queries over biological sequence
- To organize the terms in such a way that allows computational reasoning over the parts of sequence.

# What is the scope?

- Features that can be located on a sequence with coordinates. *exon*, *promoter*, *binding\_site*
- Properties of these features:
  - Sequence attributes
    - *Maternally\_imprinted\_gene*
  - Consequences of mutation
    - *mutation\_affecting\_editing*
  - Chromosome variation
    - *aneuploid*

# SOFA is a subset of SO

- **Sequence Ontology Feature Annotation**
- A subset of the SO terms that can be located on a sequence in coordinates.
- Used for automated/semi-automated annotation pipelines
- **SO** has around 900 terms
- **SOFA** has 170 terms

# What SO is

- Controlled vocabulary – terms for the concepts involved with sequence
- Descriptive biological definitions of those terms
- Synonyms of those terms
- Terms are structured into a graph by relationships.

# Controlled vocabulary

- Terms describe the concepts associated with sequence.
- Terms are computationally friendly:
  - No hyphens
  - No strange characters
  - Do not begin with a number
- The terms chosen are in common use by the community, and if they are short we like them even better.

(prefer UTR over untranslated\_region)

# Each term has a definition

- There are many types of definition
  - Logical definition – uses the proximate genus of the term and the differentiae
  - Definition by property – define carbon by the atomic number
  - Definition by cause
  - Definition by example
  - Definition by description

# Rules for making a definition

- Positive rather than negative
- Free from words sharing the same root
- Clear
- Conveys the essence of the concept to the biologist and software engineer.

# Structure

- SO is structured into a directed acyclic graph.

The screenshot displays the DAG-Edit software interface, version 1.419-beta3. The left sidebar shows a hierarchical tree of ontology terms, with 'TF\_binding\_site' selected. The main window is divided into several panels:

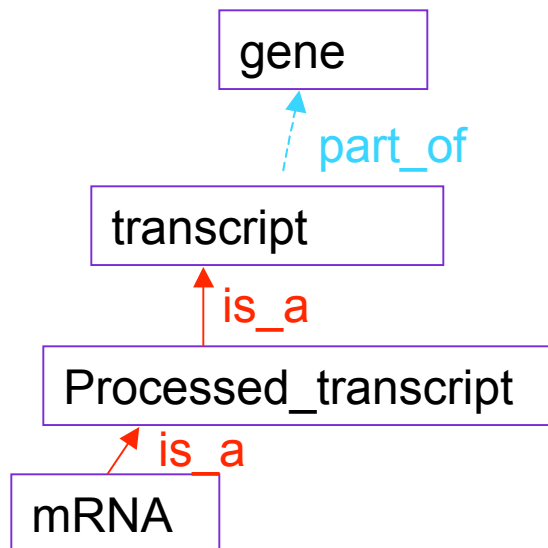
- Find terms:** A search filter is set to 'AND [0] Term has self'. The search criteria are 'term name' that 'contains' 'self'. A 'Search' button is present.
- Term details:** The selected term is 'TF\_binding\_site' with ID 'SO:0000235' and namespace 'so.ontology'. The definition is 'A region of a molecule that binds to a transcription factor.' The text field contains 'SO:ke'.
- Categories:** The category 'SO feature annotation (S...' is checked.
- Synonyms:** A synonym 'transcription\_factor' is listed with the instruction 'Select a synonym from the list to edit it, or press add to create a new synonym'.
- General DbXrefs:** A section for adding or editing database cross-references.
- DAG Viewer:** A tree view showing the parent-child relationships in the ontology, highlighting the path from 'located\_sequence\_feature' to 'region' to 'gene' to 'regulatory\_region' to 'TF\_binding\_site'.

# The relationships structure the DAG

- There are two main relationships that structure the ontology.
- `is_a` produces a hierarchy
- `part_of` produces a meronymy
- They are asymmetrical, transitive and hierarchical
- There are other minor relationships in the ontology

# The is\_a relationship

- The is\_a relation is like inheritance.
- Children terms inherit the properties and relationships of the parent term.



*mRNA* inherits the attributes of *transcript*

Therefore *mRNA* sequence is part of the *gene* sequence

# The part\_of relationship

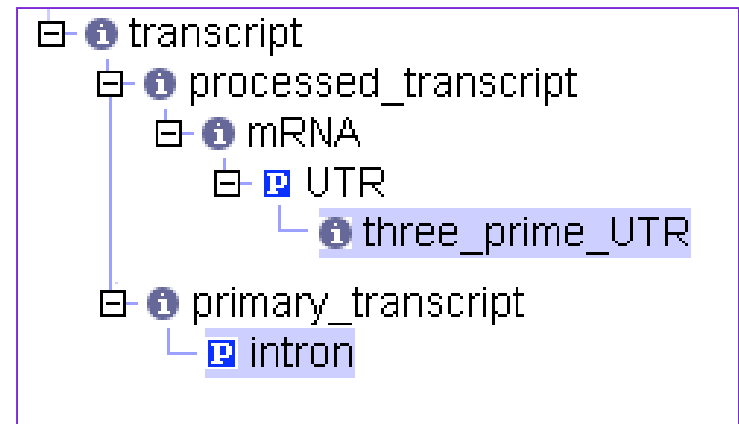
- The rules of being a part:
  - Nothing is a part of itself
  - If **A** is a part of **B** then the **B** is not a part of **A**
  - If **A** is a part of **B** and **B** is a part of **C** then **A** is a part of **C**
  - The relationship is **asymmetrical** and **transitive**
- There are subtypes of part\_of that are relevant to SO:
  - component\_part\_of
  - member\_part\_of

# Topological relationships and restrictions

- **Meets** – used when a region of sequence abuts another – ie polyA\_tail meets mRNA
- The **component\_part\_of** relation allows us to restrict the location of a term on a sequence.
  - *Exon* is **component\_part\_of** *transcript*
  - So the coordinates of the *exon* must be within those of the *transcript*.

# Relationships allow reasoning.

- **VALIDATION** - We can check the internal consistency of an annotation against the ontology. We can also check that any topological assertions are true.
- **< 3' UTR part\_of mRNA**
- **= intron part\_of mRNA**



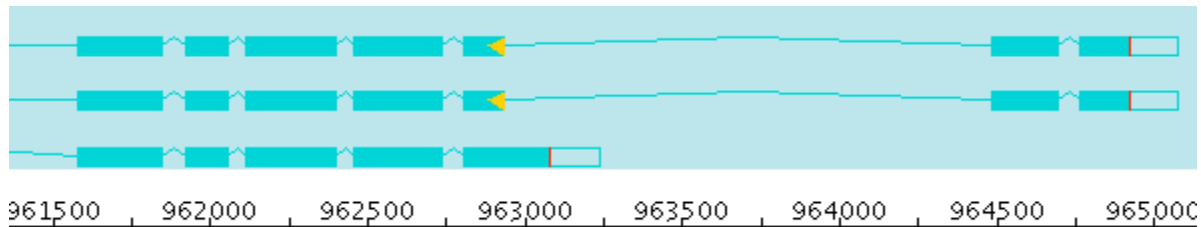
# SO is not a molecule ontology

The relationships described by it do not have to physically exist

- SO labels the parts of sequence that have biological potential.
- This means that we can infer the implied location of these parts regardless of the kind of molecule the sequence encodes
  - Which region of genomic sequence is this peptide derived from?
  - Where is the splice junction on the translation?

## For example

- We can label the genomic sequence with concepts that are normally associated with RNA or protein.
- *exon*, *non\_canonical\_splice\_site*, *stop\_codon*, *signal\_peptide*



- we can infer the positions of these parts in different substrates.

# SO is not a database or file format

- SO is not a database schema. It is an ontology.
- It therefore transcends any particular database schema or file-format
- It can be used equally well to type the concepts in a data exchange format or as integral components of a database

# SO is not a database or file format...

it needs a data model

- A data model is a framework for capturing knowledge in a way that is computable
  - We need a data model for recording SO instances
- Data model formalisms
  - Relational i.e. database schema
  - Hierarchical i.e. XML
  - Object Oriented i.e. perl objects
  - Ontology formalisms i.e. OWL
  - flat file formats i.e. GenBank flat file format
- Multiple formalisms are suitable for modeling SO instances

# Existing data models reliant upon SO or SOFA to type features

- **GFF3**
  - tab delimited text
  - wide spread genome data exchange format
- **Chado** – relational schema from GMOD
  - ChadoXML
  - ChaosXML
- FlyBase, SGD, WormBase, TIGR + others use SO to type features
- (EMBL mapping to SOFA in pipeline eta june 2005)

## 2 ways to get SO compliant annotations

- De novo annotation – many of the model organism groups now annotate their sequences using SO or SOFA (E.g. SGD, FlyBase).
- Convert existing annotations to SO compliant format.
  - bioperl tool called **unflattener** converts GenBank annotations to Bioperl objects, to SO compliant form.

# How are SO terms made?

- Someone proposes a new term.
- SO community debates new terms and their position in the ontology, their properties, synonyms and definitions via mailing list.
- [song-devel@lists.sourceforge.net](mailto:song-devel@lists.sourceforge.net)



# Where can I get it?

- <http://song.sourceforge.net>
- SO and SOFA are in obo format

# How do I view it?

- DAG-Edit
- <http://sourceforge.net/projects/geneontology>

# Conclusions

- SO unifies the way we describe biological sequence
- This simplifies querying and analysis, especially between organisms
- The relationships allow reasoning over SO instances which facilitates complex analysis -e.g. validation
- Many data models can adopt SO to type their sequence instances
- SO is open source

# Acknowledgements

- Suzi Lewis
- Michael Ashburner
- Chris Mungall
- John Day-Richter
- Lincoln Stein
- Judy Blake
- Richard Durbin
- Contributors to developers mailing list
  
- Funded by NIH via Gene Ontology Consortium

