Biological databases and resources

Bioinformatics · Universitat Politècnica de Catalunya

Alexandre Perera-Lluna

Database Organization and Abstraction Levels

- Nucleotide
- Protein Sequence
- Protein Structure
- Genetic Variability
- Expression
- Disease
- Metabolic/Proteic Pathways
- In-level (nucleotide $\leftrightarrows$ nucleotide)
- Cross-level (nucleotide $\leftrightarrows$ protein)

Database access

Databases can be accessed locally or online.

- Most DBs offer a dual access (web based, API services).
- There are two main types of DBs access.

Limited Access

Human access. Exploratory queries. Few queries.

Massive Access

- Mostly used in cross-reference.
- Massive programmatic access.
- Some DBs limit the access/IP/hour.
- Most can be downloaded.

Primary Data

- Raw experimental results.
- Initial interpretation (Nucleotide sequences).
- Protein sequence records. Reviewed annotations add a derived layer of interpretation.

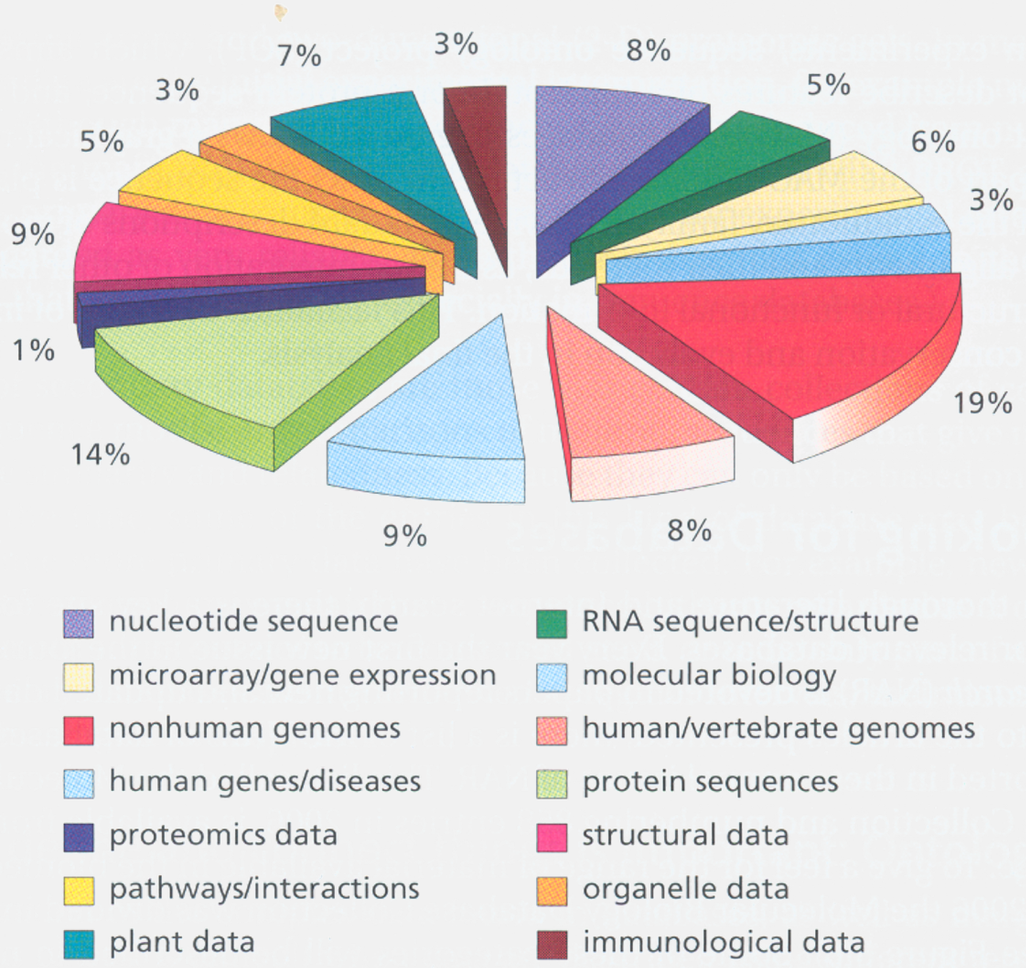
Derived or Secondary Data

- Collection of conserved protein motifs.
- Aggregation of Primary databases.
- Results are very specific of query time. Is the database updated regularly?

Ontology Databases

Formal and explicit specifications of the terms used, and the relationship between them.

Distribution of the Type of DBs



Annotation quality

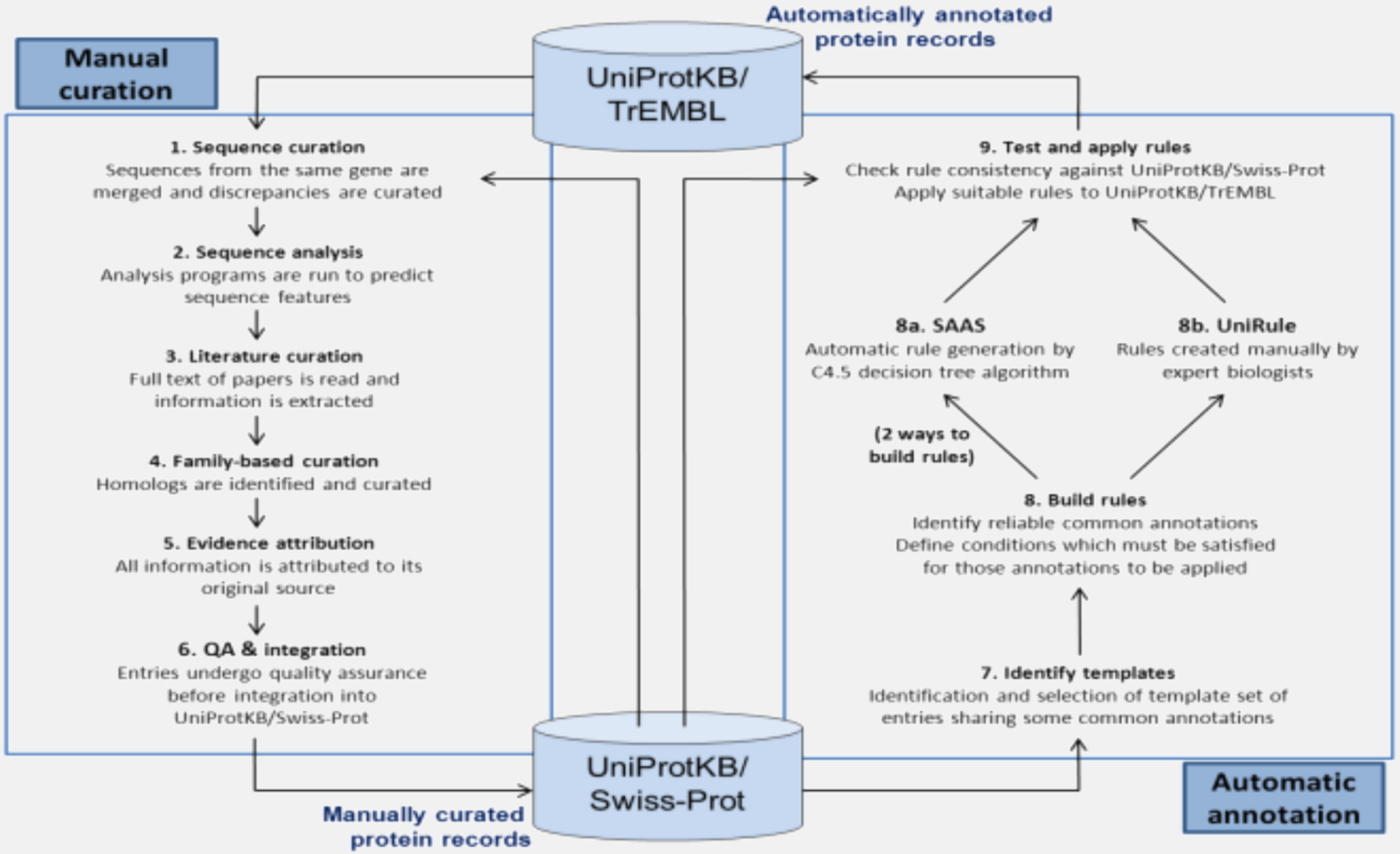
- Automated annotation transfers or predicts information using defined computational evidence.
- Manual curation evaluates evidence and resolves inconsistencies.
- Assess the evidence type, provenance, and limitations of each annotation.
- Neither the annotation method nor the database name alone guarantees correctness.

Evidence Codes (The Gene Ontology Example)



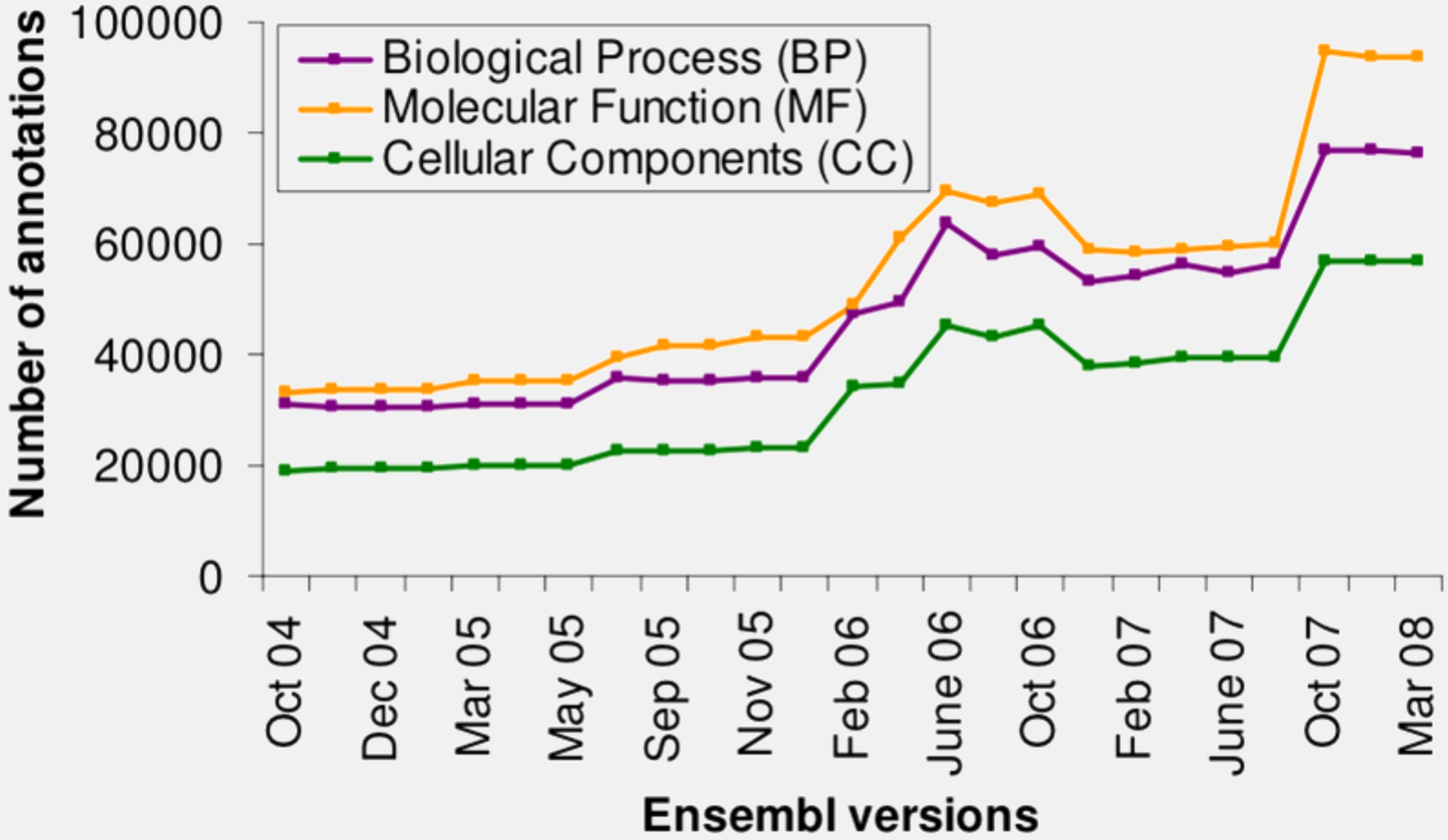
- **Introduction**
  - Experimental Evidence Codes
    - EXP: Inferred from Experiment
    - IDA: Inferred from Direct Assay
    - IPI: Inferred from Physical Interaction
    - IMP: Inferred from Mutant Phenotype
    - IGI: Inferred from Genetic Interaction
    - IEP: Inferred from Expression Pattern
  - Computational Analysis Evidence Codes
    - ISS: Inferred from Sequence or Structural Similarity
    - ISO: Inferred from Sequence Orthology
    - ISA: Inferred from Sequence Alignment
    - ISM: Inferred from Sequence Model
    - IGC: Inferred from Genomic Context
    - RCA: inferred from Reviewed Computational Analysis
  - Author Statement Evidence Codes
    - TAS: Traceable Author Statement
    - NAS: Non-traceable Author Statement
  - Curator Statement Evidence Codes
    - IC: Inferred by Curator
    - ND: No biological Data available
- **Automatically-assigned Evidence Codes**
  - IEA: Inferred from Electronic Annotation
- **Obsolete Evidence Codes**
  - NR: Not Recorded
- **Note on Usage of the With/From Column**

Uniprot Curation Scheme



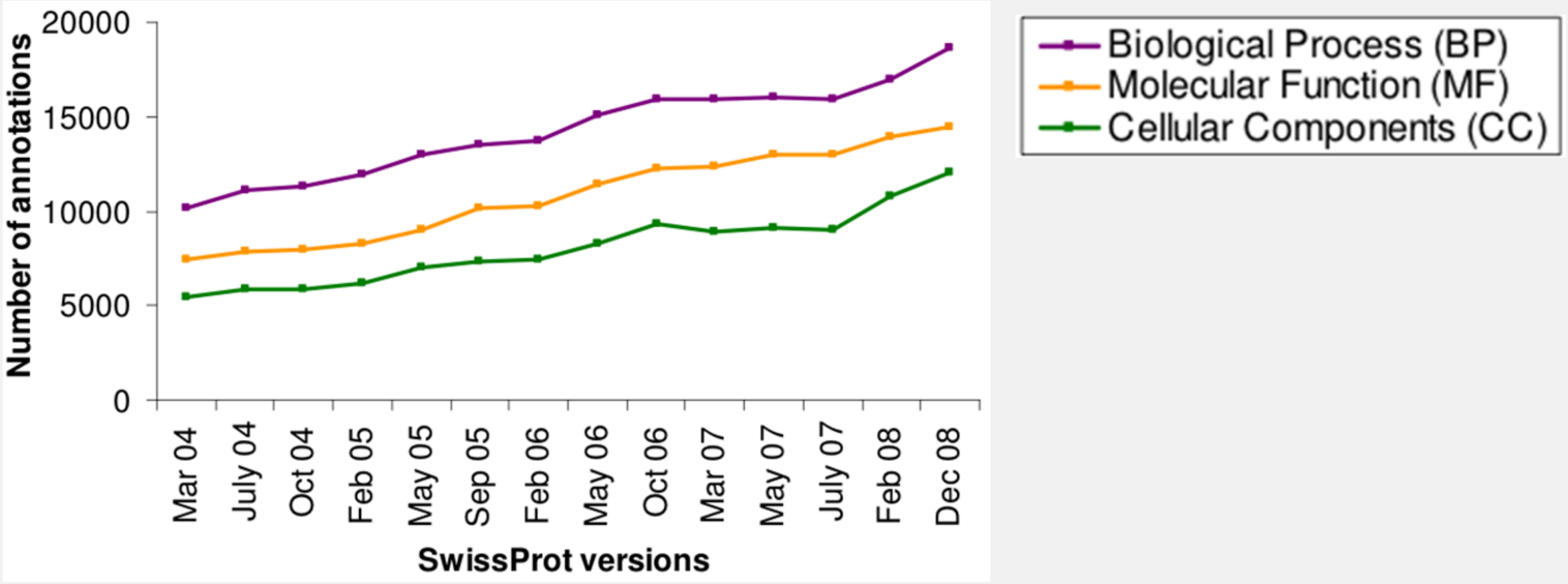
from http://www.uniprot.org/help/biocuration

Evolution of Annotations in Ensembl



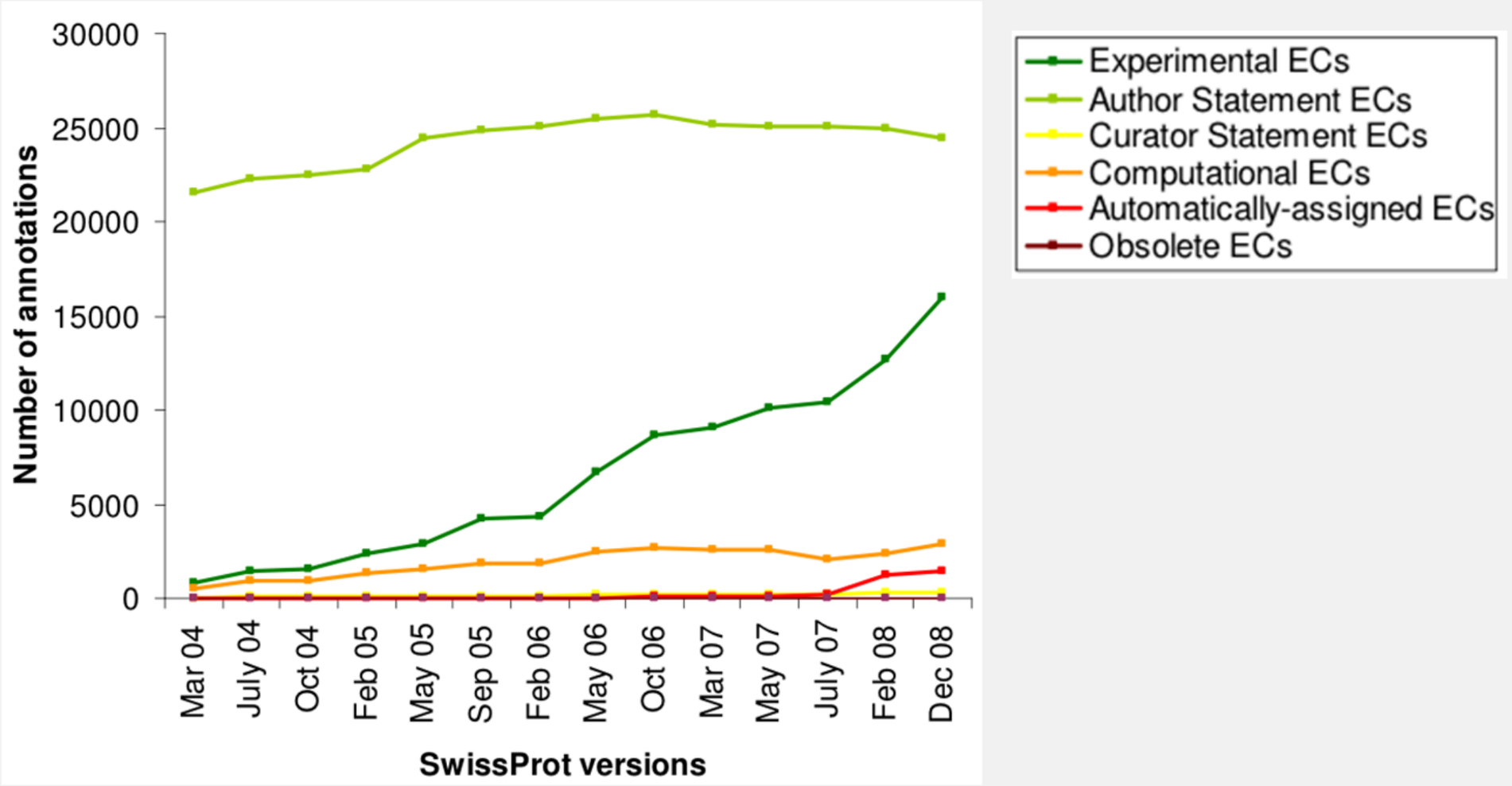
from Groß et al. 2009

Evolution of Annotation in Swiss-Prot



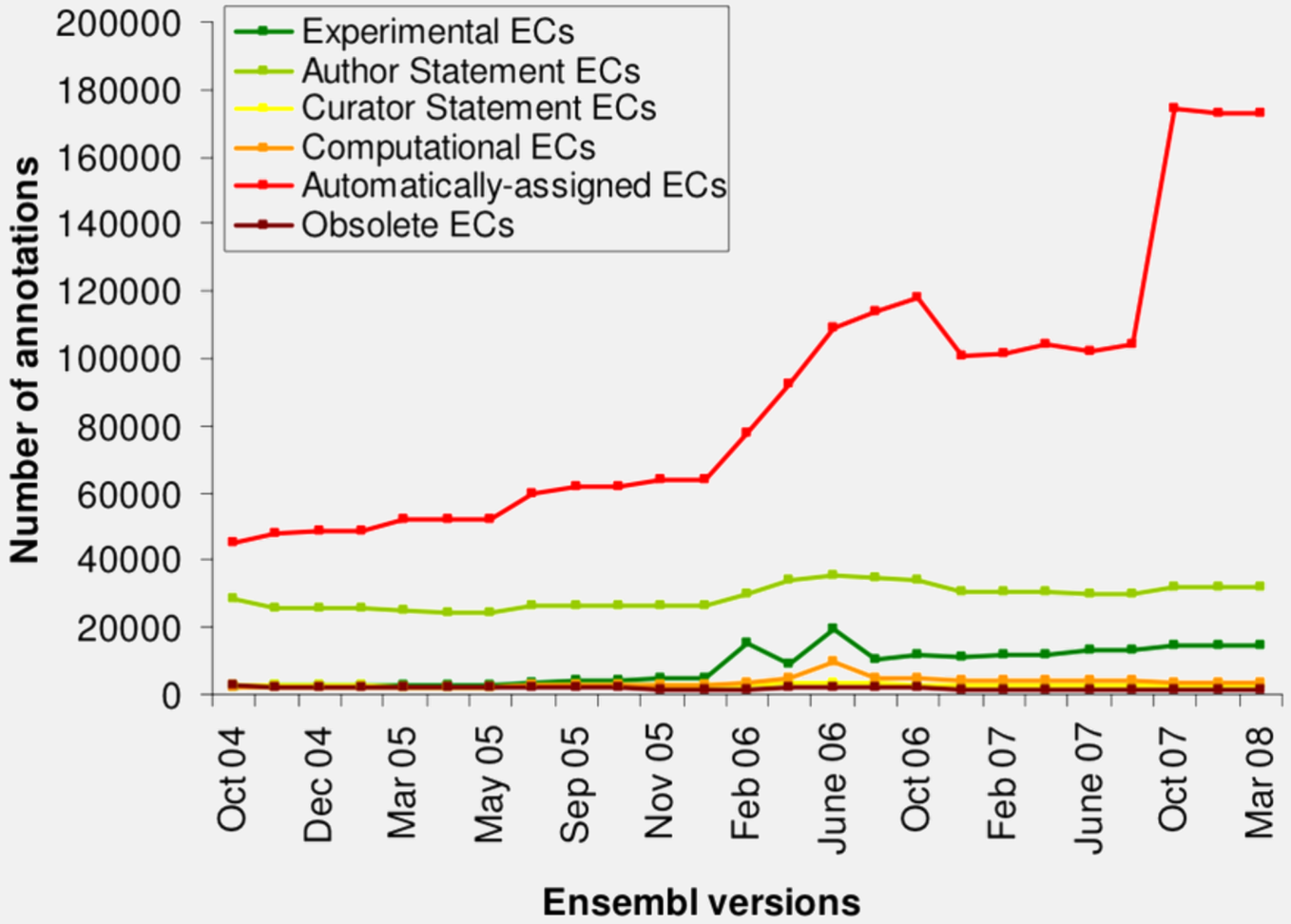
from Groß et al. 2009

Evolution of ECs in Swiss-Prot



from Groß et al. 2009

Evolution of ECs in EMBL



from Groß et al. 2009

Evidence and annotation scope

- Compare resources using the same organisms, sequence types, and annotation categories.
- UniProtKB/Swiss-Prot contains reviewed protein records.
- Ensembl provides genome annotation, including computationally generated gene models.
- For a functional claim, inspect the supporting evidence rather than comparing raw annotation counts.

RAW Genomic Sequences

Sequence of chromosomal DNA.
Derived from genome sequencing projects (GenBank).

cDNAs

Sequences of DNA molecules that have been synthesized by reverse transcription of mRNA molecules).lack any intronic/control sequences related to the gene.

Expressed Sequence Tags (ESTs)

An EST is a partial cDNA sequence.
A library of EST, indicates the range of genes being expressed in a sample.

Nucleotide Conventions (1/3)

Symbol	Meaning	Origin of designation
G	G	Guanine
A	A	Adenine
T	T	Thymine
C	C	Cytosine
R	G or A	puRine

Nucleotide Conventions (2/3)

Symbol	Meaning	Origin of designation
Y	T or C	pYrimidine
M	A or C	aMino
K	G or T	Keto
S	G or C	Strong interaction (3 H bonds)
W	A or T	Weak interaction (2 H bonds)

Nucleotide Conventions (3/3)

Symbol	Meaning	Origin of designation
H	A or C or T	not-G, H follows G in the alphabet
B	G or T or C	not-A, B follows A
V	G or C or A	not-T (not-U), V follows U
D	G or A or T	not-C, D follows C
N	G or A or T or C	aNy

RefSeq accession prefixes

RefSeq accessions have an underscore after the two-letter prefix.

Prefix	Common meaning
`NC_`	Complete genomic molecule
`NM_` / `NR_`	RefSeq mRNA / non-coding RNA
`NP_`	RefSeq protein
`XM_` / `XR_` / `XP_`	Computational model transcript / protein

The accession **version** identifies a specific sequence version.

Assignment (1/2)

Please elaborate a description of the following databases:

1. The Gene Ontology Project (http://www.geneontology.org/)
2. Human Gene Mutation Database (HGMD) (http://www.hgmd.cf.ac.uk)
3. Gene Cards (http://www.genecards.org)
4. The Mouse Genome Informatics (http://www.informatics.jax.org)
5. Protein Structure Classification CATH (http://www.cathdb.info/)
6. NCBI Short Genetic Variations Database (dbSNP) (https://www.ncbi.nlm.nih.gov/snp/)
7. Human Phenotype Ontology (https://hpo.jax.org)
8. PRINTS Database (Protein Fingerprints) (http://bioinf.man.ac.uk/dbbrowser/PRINTS/)
9. Kyoto encyclopedia of Genes and Genomes (just PATHWAY Section) (http://www.genome.jp/kegg/)

Assignment (2/2)

10. Database of Genomic Variants (http://projects.tcag.ca/variation/)
11. Reactome (https://reactome.org)
12. STRING - Known and Predicted Protein-Protein Interactions (http://string-db.org/)
13. BioGRID - Biological General Repository for Interaction Datasets (http://thebiogrid.org/)
14. OMIM - Online Mendelian Inheritance in Man (https://www.omim.org)
15. Pfam protein families in InterPro (https://www.ebi.ac.uk/interpro/entry/pfam/)

Slides (8’ each + questions)

Describe the following items for your database (if applicable).

1. Historic remark, research group/company, url, purpose, primary/secondary.
2. Database architecture and organization.
3. Annotation Statistics.
4. Annotation format, main IDs description.
5. Entry example/s description.
6. *Curation* (manual, automatic, evidence codes or levels).
7. Programmatic access (webservice, location, available software packages for remote access).
8. Access conditions/license.

EMBOSS

The European Molecular Biology Open Software Suite. EMBOSS is a free Open Source software analysis package specially developed for the needs of the molecular biology (e.g. EMBnet) user community.

http://emboss.sourceforge.net

- The software automatically copes with data in a variety of formats and even allows transparent retrieval of sequence data from the web.
- Extensive libraries are provided with the package, it is a platform to allow other scientists to develop and release software in true open source spirit.
- EMBOSS also integrates a range of currently available packages and tools for sequence analysis into a seamless whole. EMBOSS breaks the historical trend towards commercial software packages

EMBOSS applications

- Sequence alignment .
- Rapid database searching with sequence patterns.
- Protein motif identification, including domain analysis
- EST analysis
- Nucleotide sequence pattern analysis, for example to identify CpG islands.
- Simple and species-specific repeat identification
- Codon usage analysis for small genomes
- Rapid identification of sequence patterns in large scale sequence sets.
- Presentation tools for publication.
- …

Databases searches

- Searching by keyword, identifiant ...
- Search similar sequences in databases
- HMMER: Sequences search or HMM profiles search
- Blast post-processing

Sequence format conversions and sequence edition

- Formats conversions
- Features
- Sequence edition
- Sequences manipulation
- Display

Sequences alignments and comparisons

- Sequences comparaison and sequence consensus
- Pairwise comparisons
- Multiple alignments
- Alignments display
- Structural alignments
- HMM (Hidden Markov Models)

Phylogeny

- Parsimony method programs
- Distance matrix method programs
- Maximum likelihood method programs
- Computation of distance
- Manipulation and visualization of phylogenetic tree
- Other programs

Pattern search and discovery, HMM

- Pattern discovery
- HMM (Hidden Markov Models)
- Other programs

Protein sequence analysis

- Pattern searching
- Protein structure analysis
- Protein properties
- Enzyme Kinetics

Nucleic sequence analysis

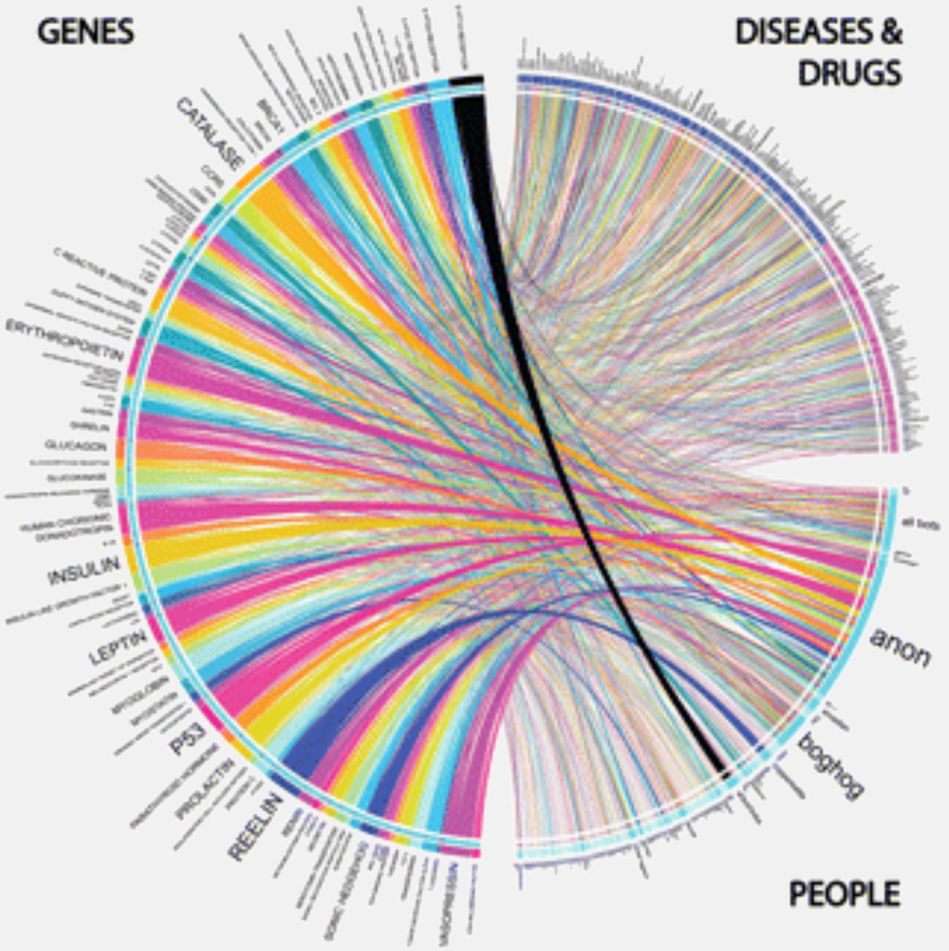
- Pattern searching
- Genes searching, coding regions
- Restriction enzyme sites
- Transcription factors
- Repeats
- Codon usage
- CpG islands
- Primers
- Nucleic sequences properties and composition
- Features analysis
- RNA analysis
- Nucleic sequence translation
- Display
- Other programs

Structure analysis

- Secondary structure
- Tertiary structure

http://bioweb2.pasteur.fr/

Annual Database Issue of Nucleic Acids Research



Bio-computing Languages

- Bio-Perl.
- Bio-Java.
- Bio-Python.
- R, through Bioconductor.

PERL

Perl is a general-purpose language with a long history in text processing.
http://perl.org

- Interpreted.
- Some features similar to *UNIX* shell.
- Some other features extracted from *AWK*, *regexp* and *sed*.
- Automatic memory-management.
- Extremely fast on text processing.
- Classic language for CGI-scripting on web.
- Also used in finance and bioinformatics.

BioPerl

BioPerl is a toolkit of perl modules useful in building bioinformatics solutions in Perl.
http://bioperl.org

- Object oriented.
- bioperl-live repository consists of the core of the functionality of bioperl.
- bioperl-db persistent storage in RDMBS.
- bioperl-gui auxiliary modules for creating graphical interfaces.
- bioperl-pipeline software to automate bioinformatic analyses.
- bioperl-pedigree objects for Pedigrees, Individuals, Markers, & Genotypes.

GC content across FASTA records

```python
from Bio import SeqIO

records = list(SeqIO.parse("data/example.fa", "fasta"))
total = sum(len(r) for r in records)
if total == 0:
    raise ValueError("No sequence bases found")
gc = sum(str(r.seq).upper().count(b)
         for r in records for b in "GC")
print(f"Total bases: {total}")
print(f"GC content: {100 * gc / total:.1f}%")
```

This is a Python example. It includes ambiguous symbols in the denominator.

BioJava is

A open source project dedicated to providing a Java framework for processing biological data.
http://biojava.org

- Manipulating biological sequences, file parsers.
- DAS client and server support.
- Access to BioSQL and Ensembl databases.
- Tools for making sequence analysis GUIs.
- Powerful analysis and statistical routines including a dynamic programming toolkit.
- Web programming, multiplatform.

GC content with BioJava

```java
import org.biojava.nbio.core.sequence.DNASequence;

DNASequence dna = new DNASequence("ACGTACGT");
String s = dna.getSequenceAsString().toUpperCase();
long gc = s.chars().filter(c -> c == 'G' || c == 'C').count();
double percent = 100.0 * gc / s.length();
System.out.println(percent);
```

This example uses the modern `org.biojava.nbio` namespace. For FASTA files, use BioJava's `FastaReaderHelper`.

Python

Powerful dynamic programming language (“batteries included”)
http://python.org

- Very clear, readable syntax.
- Intuitive object orientation
- Natural expression of procedural code
- Full modularity, supporting hierarchical packages
- Very high level dynamic data types
- Extensive standard libraries and third party modules for virtually every task.
- Extensions and modules easily written in C, C++ (or Java for Jython, or .NET languages for IronPython)
- Embeddable within applications as a scripting interface

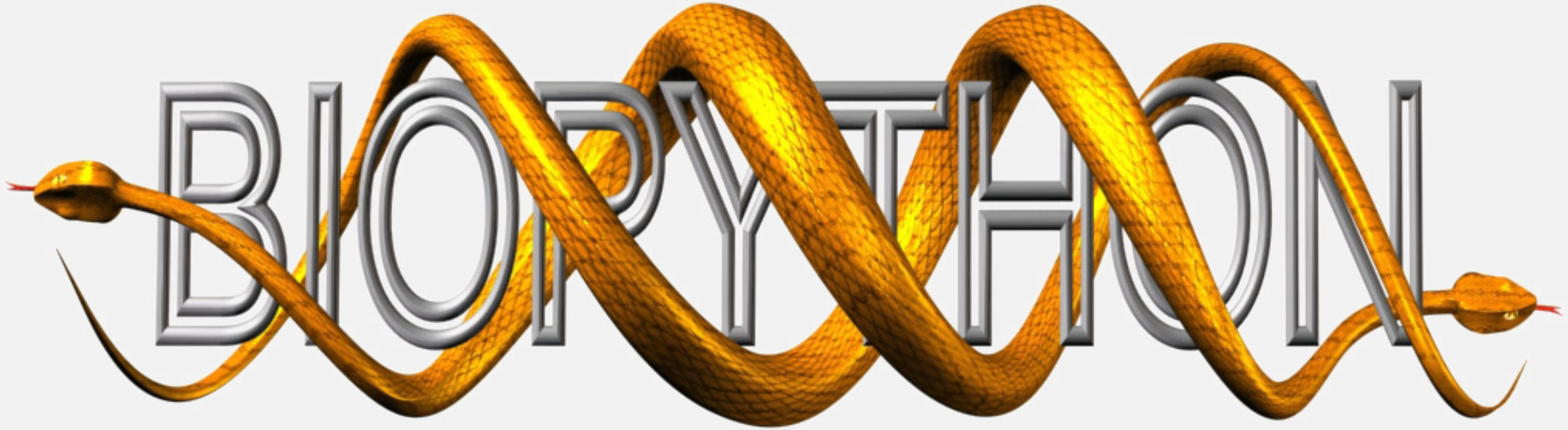
Biopython

Biopython provides Python tools for working with biological data.

- Read and write sequence files with `Bio.SeqIO`.
- Represent sequences and annotated records.
- Read, compare, and construct alignments.
- Work with substitution matrices and structural data.
- Access NCBI services through the relevant interfaces.
- Store annotated sequences through BioSQL integration.

Documentation and tutorials

Biopython Tutorial and Cookbook, By Jeff Chang, Brad Chapman, Iddo Friedberg, Thomas Hamelryck, Michiel de Hoon, Peter Cock, Tiago Antao, Eric Talevich, Bartek Wilczyński.



http://biopython.org/DIST/docs/tutorial/

GC content in Python

```python
def gc_percent(dna):
    sequence = str(dna).upper()
    if not sequence:
        raise ValueError("The sequence is empty")
    return 100 * (sequence.count("G") +
                  sequence.count("C")) / len(sequence)
```

The denominator includes all symbols. Ambiguous bases contribute to length but not to the G + C count.

For `"aCgTN"`, the GC content under this convention is **40%**.

Reading genome annotations with Biopython

```
1  from Bio import SeqIO
2  from Bio.Graphics import GenomeDiagram
3  from reportlab.lib import colors
4
5  record = SeqIO.read("data/NC_005816.gb", "genbank")
6  diagram = GenomeDiagram.Diagram(record.id)
7  track = diagram.new_track(1, name="Coding sequences")
8  features = track.new_set()
9
10 for feature in record.features:
11     if feature.type == "CDS":
12         features.add_feature(feature, color=colors.teal)
```

`SeqIO.read()` expects one record. Use `SeqIO.parse()` for a file containing several records.

Drawing a genome diagram

```
diagram.draw(format="linear", pagesize="A4",
             fragments=1, start=0, end=len(record))
diagram.write("genome-diagram.pdf", "PDF")
```

The diagram uses the annotated coding-sequence coordinates in the GenBank record.

R and Bioconductor





The End

