Microbial Genomics

Outline

- Genomics Terminology
- Assemblies vs. Variants
- Assembly-based analyses
- Orthology
- Variant-based analyses
- How to choose?

Basic Genomics Terminology

- Assembly: Reconstruction of a longer sequence from smaller sequencing reads
- Annotation: Assigning a function to a string of nucleotides
- Variant calling: Identifying differences between a set of sequencing reads and a reference assembly

Whole genome shotgun sequencing

- Rapid
- Generation of small insert genomic library
- Library is not initially ordered
- DNA sequence ends of inserts
- Depends on powerful computing to assemble sequence read

Challenges

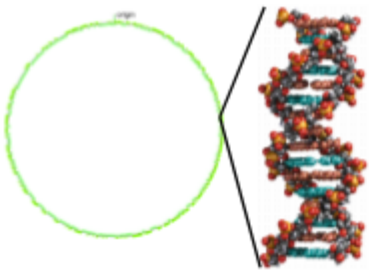
Removal of artifacts in short reads ??

Genome assembly of short reads ??

Several assemblers available, which is best ??

Annotation and validation of assembled genome ??

Sequencing a genome



fragments of sequence

luatedgeneticsrel ourcesforteach chershealthprofession cisahubofevaluatedgen esforteachershealt atedgene ourcesforteach luatedgeneticsrel atedgene cisahubofevaluatedgenc chershealthprofession esforteachershealt luatedgeneticsrel hprofessionalsandgeneralpub hprofessionalsandgeneralpub tatedgene cisahubofevaluatedgenc chershealthprofession ourcesforteach

vgecisahubof
bofevaluatedgenetics
icsrelatedresourcesforteachershealth
lthprofessionalsandgeneralp
generalpublic

overlaps

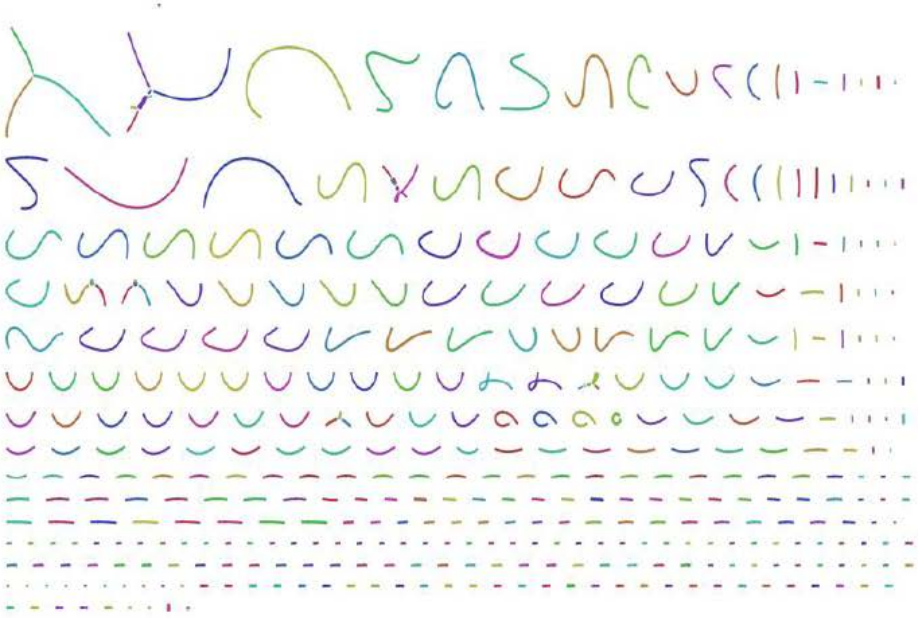
contiguous sequence

vgecisahubofevaluatedgeneticsrelatedresourcesforteachershealthprofessionalsandgeneralpublic

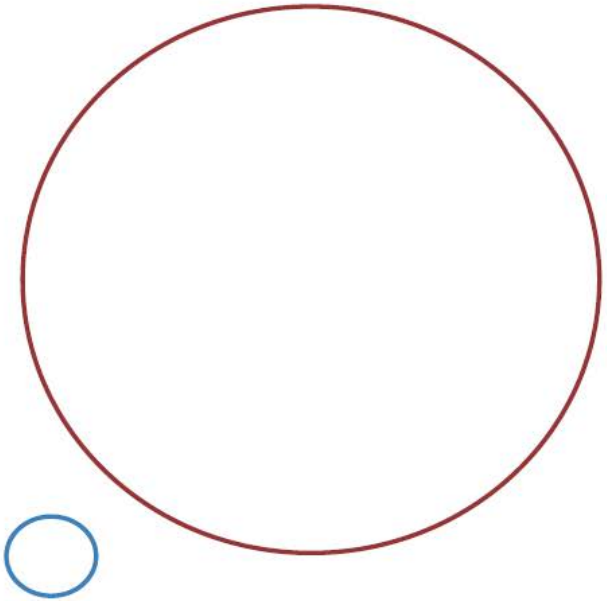
annotation

VGEC is a hub of evaluated genetics related resources for teachers, health professionals and general public.

Draft vs. finished genomes

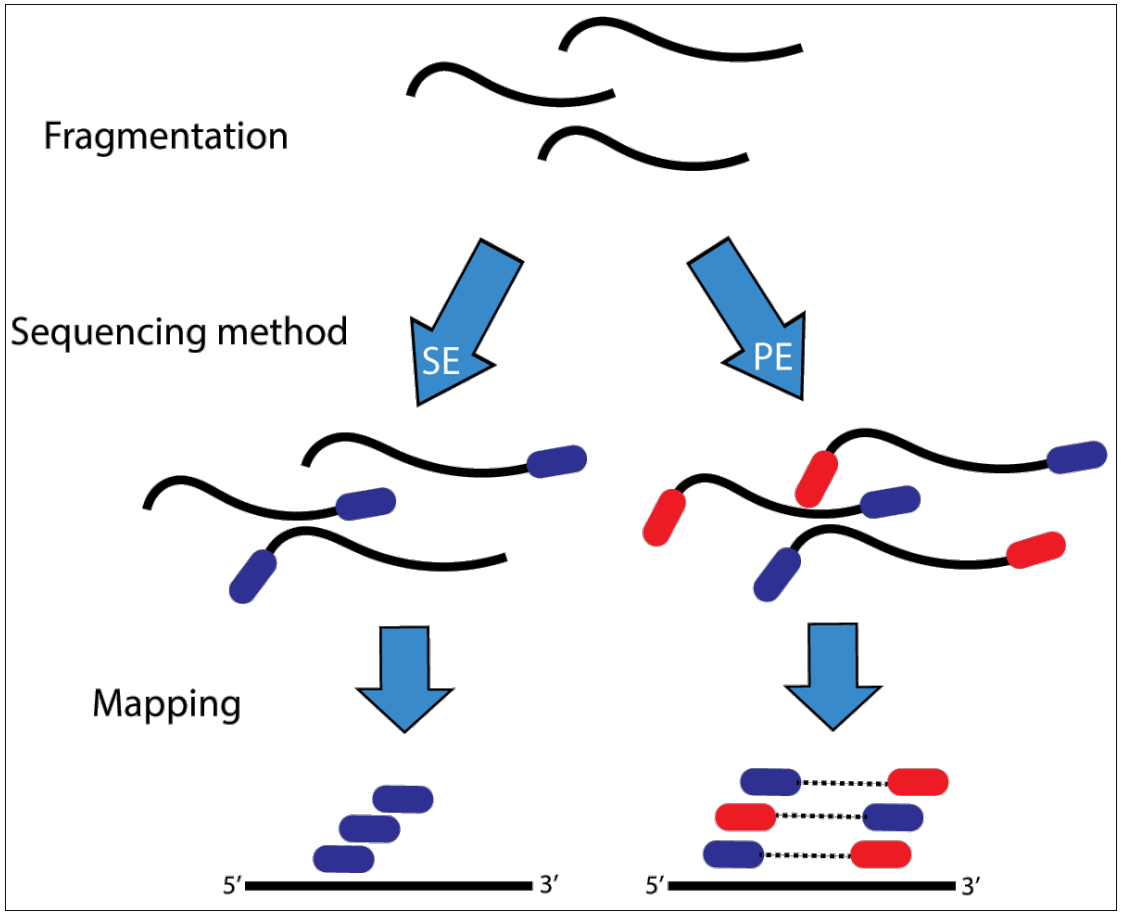


Lots of contigs



One contig per replicon

Illustration of single-end (SE) versus paired-end (PE) sequencing



The raw sequence file

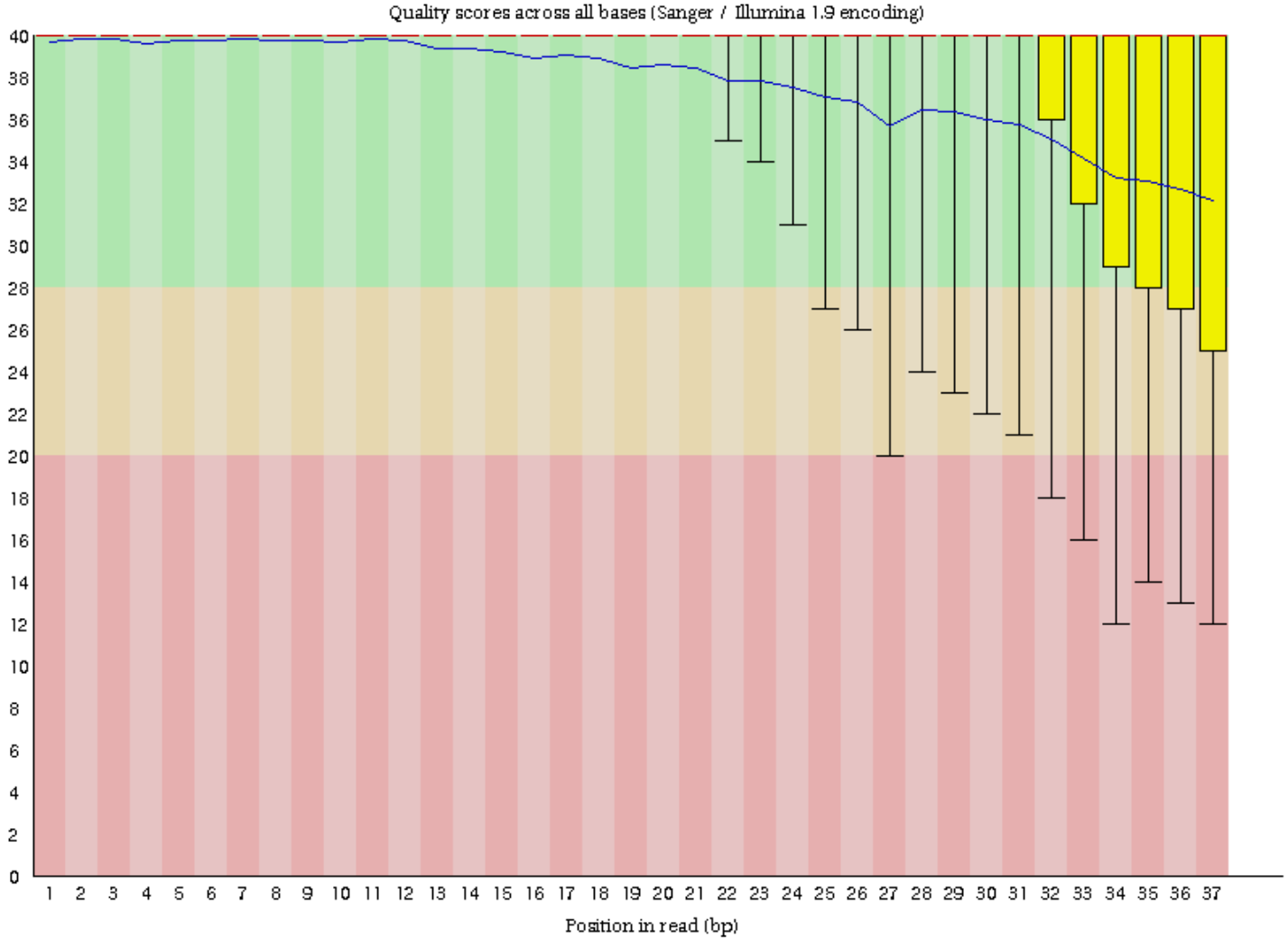
- FastQ file: Text-based format for storing both a biological sequence (nucleotide sequence) and its corresponding quality scores. Both, the sequence letter and quality score are each encoded with a single ASCII character.
- Each nucleotide is assigned an ASCII character, representing its Phred quality score, the probability of an incorrect base call

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Phred quality score

Phred Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%
60	1 in 1,000,000	99.9999%

Per base sequence quality

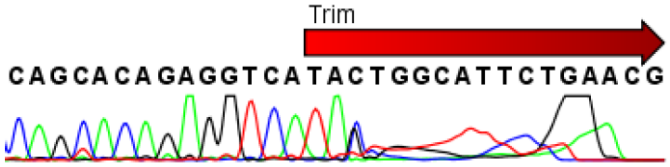


Tools: FastQC

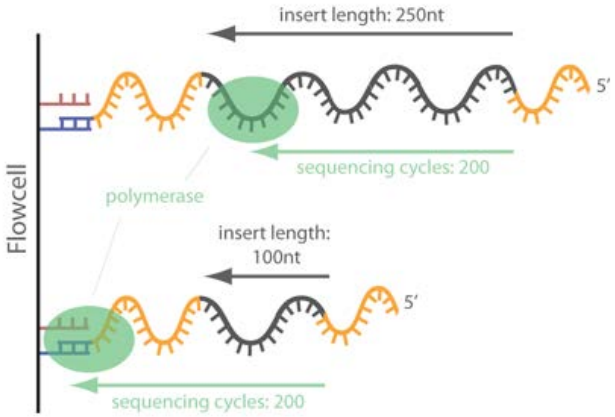
Sequences must be treated to reduce bias in downstream analysis

In general, quality treatments include:

- Filtering of sequences
  - with low mean quality score
  - too short
  - with too many ambiguous (N) bases
  - based on their GC content

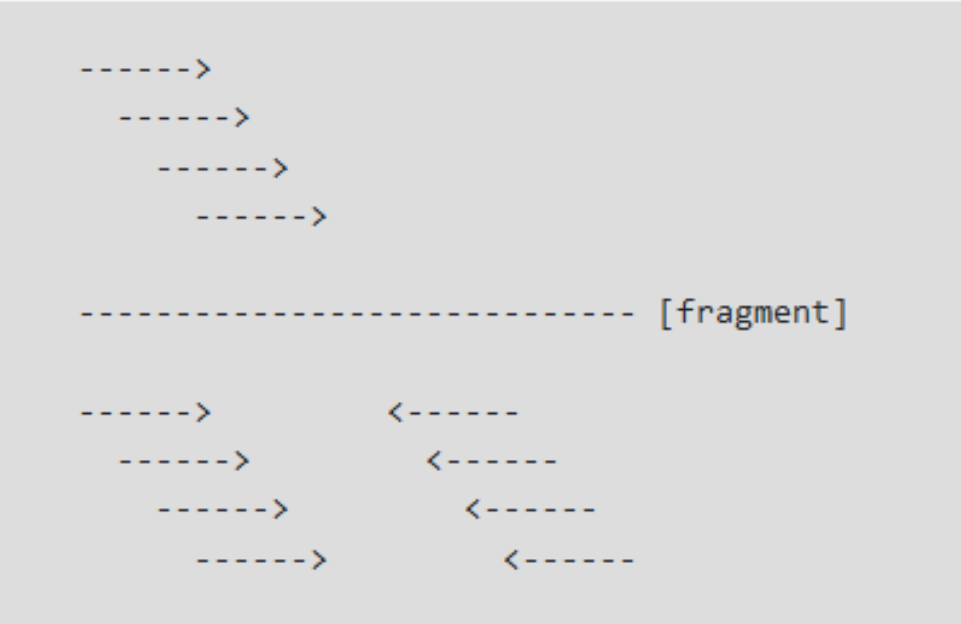


- Cutting/Trimming/masking sequences
  - from low quality score regions
  - beginning/end of sequence
  - removing adapters



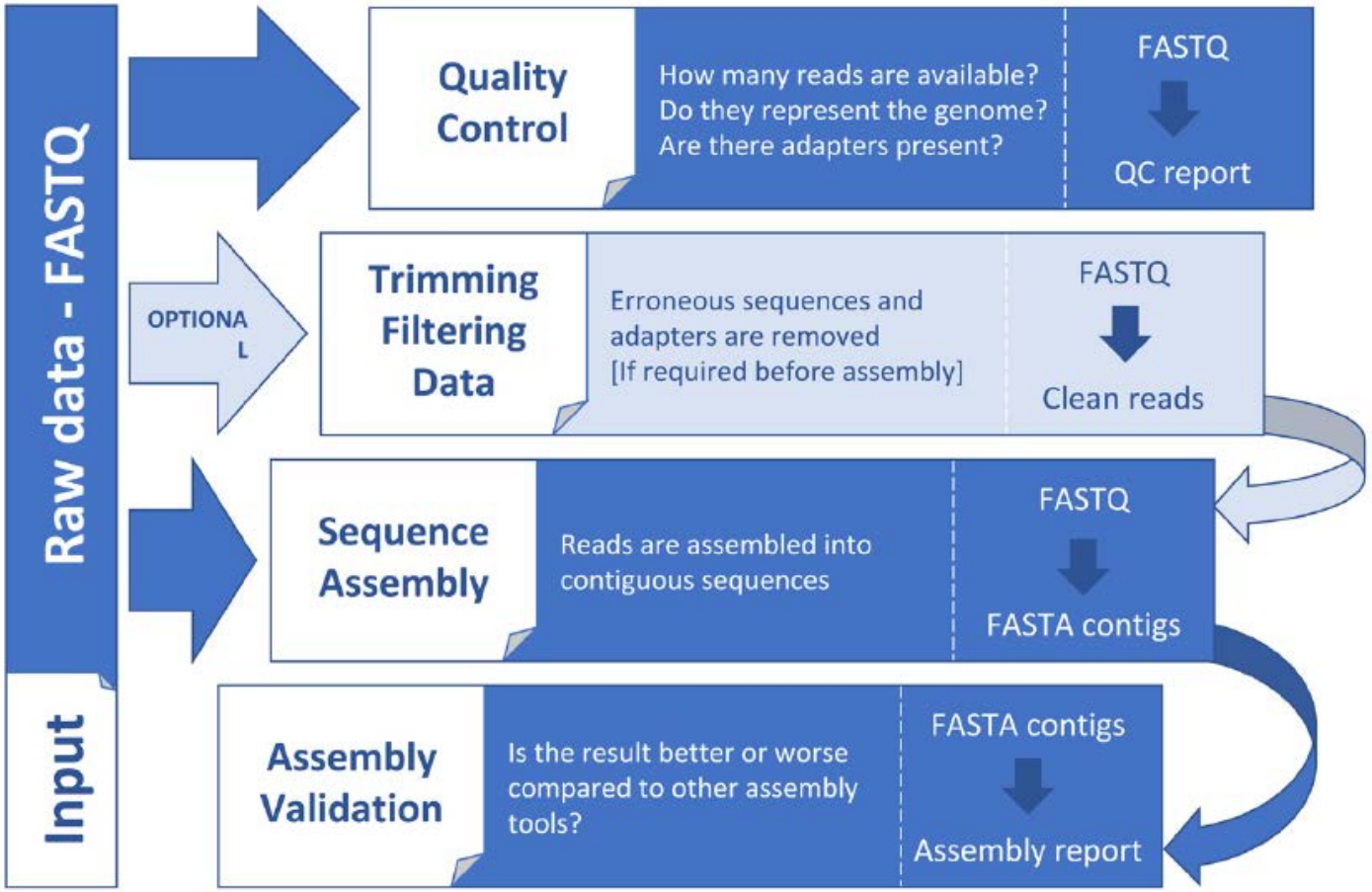
Tools: Sickle, Cutadapt

Single vs. paired-end reads

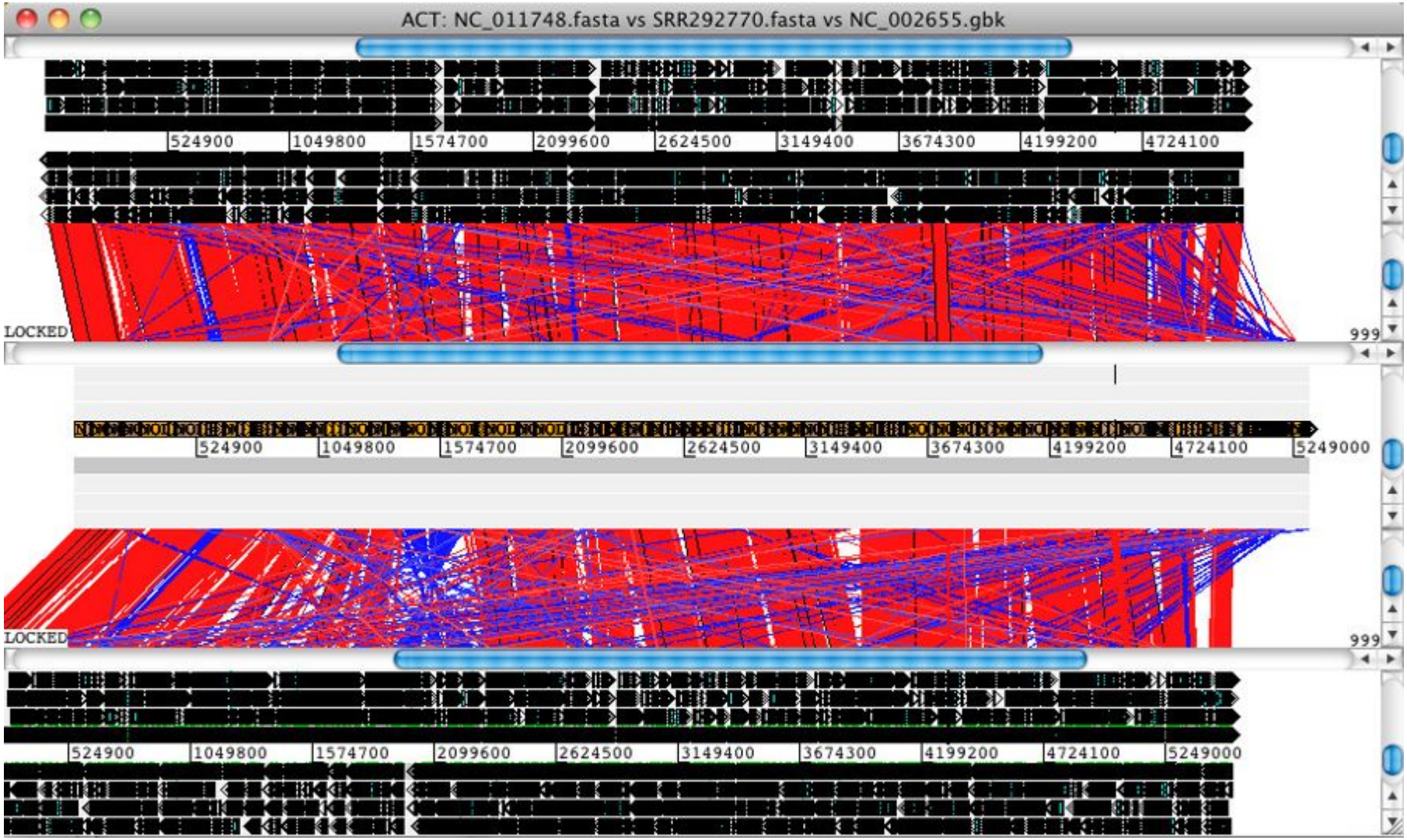


Paired-end sequencing generates 2 FASTQ files:
One file with the sequences corresponding to foward orientation of all the fragments.
One file with the sequences corresponding to reverse orientation of all the fragments.

Steps in a genome assembly workflow

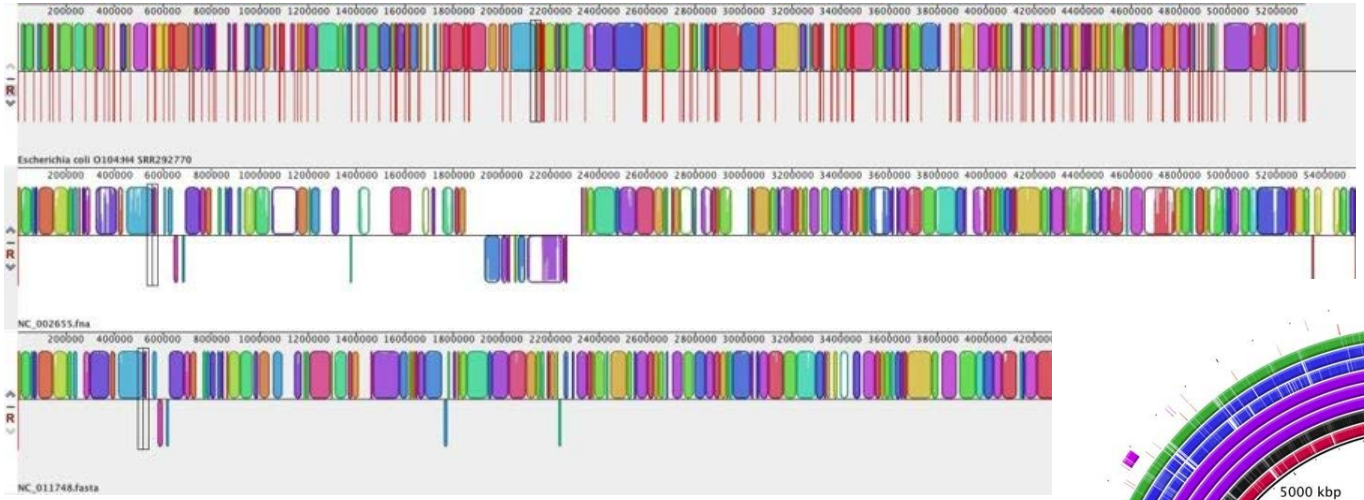


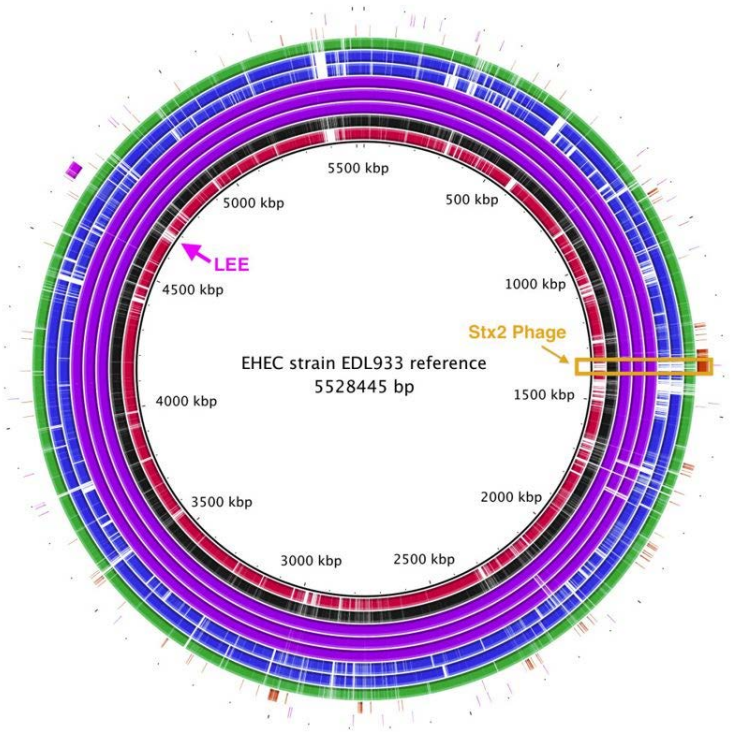
Pairwise genome comparison



Tools: ACT

Multiple genome comparison

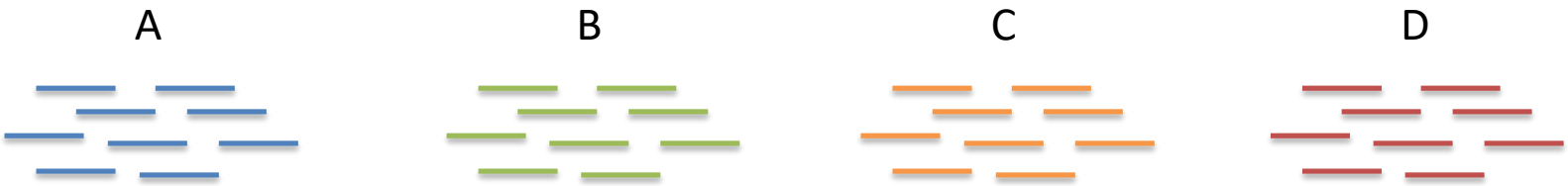




Tools: Mauve, BRIG

Two Approaches to Microbial Genomics

Starting with sets of reads representing your study isolates…



<u>Assembly-based</u>

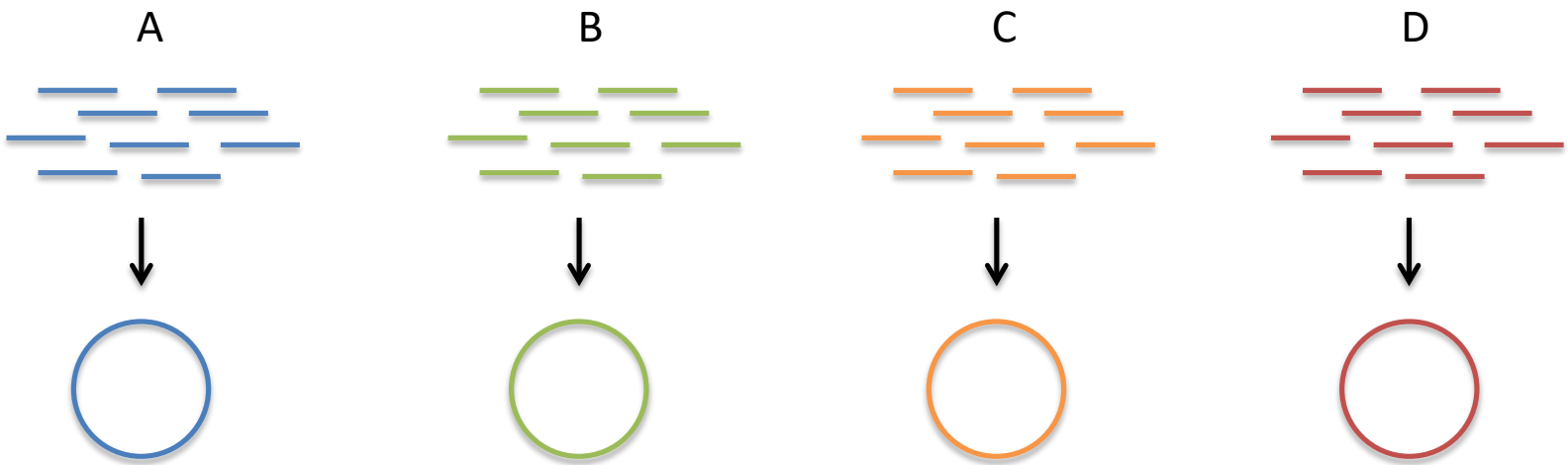
1. Assemble each set of reads into a genome sequence
2. Annotate each genome
3. Cluster genes and compare between each genome

<u>Variant-based</u>

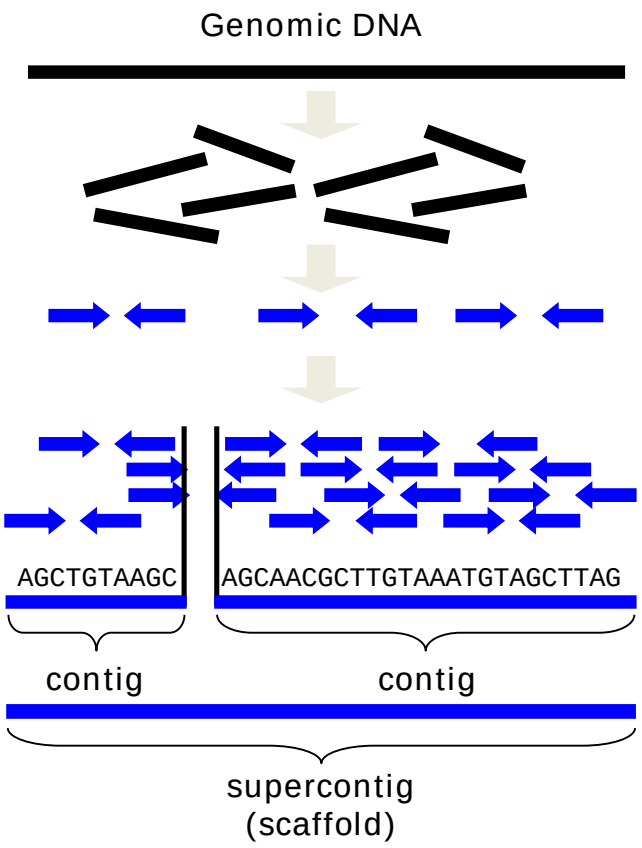
1. Compare each read set to a reference genome assembly
2. Directly compare variants between each genome

Assembly-Based Approach

1. Assemble each genome (*de novo* or reference-based)



Assembly Basics

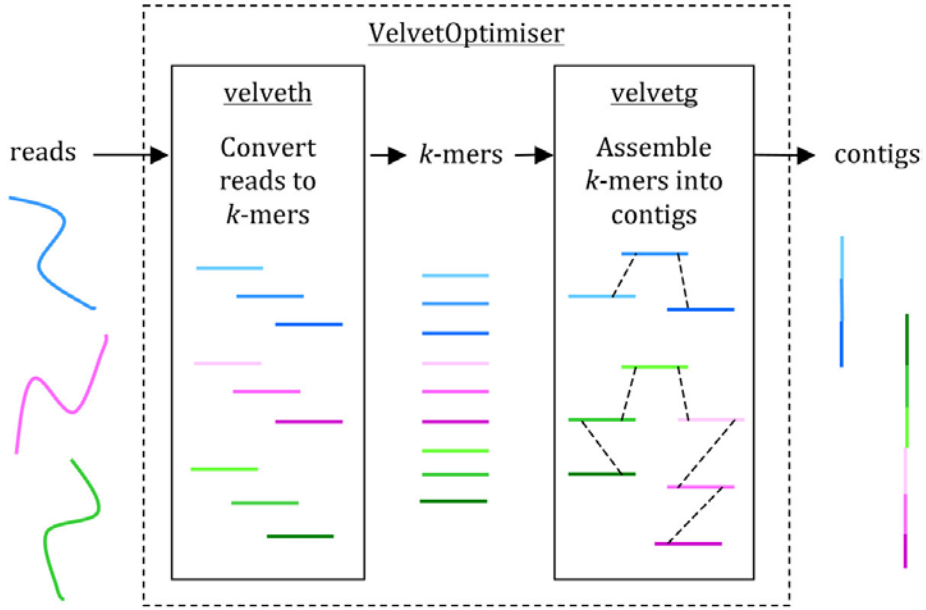


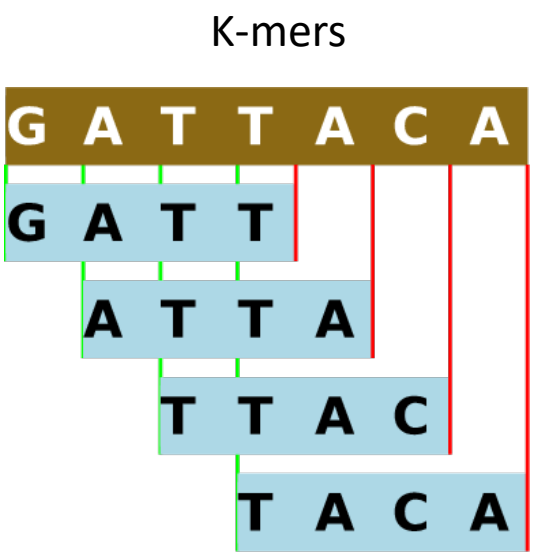
Assembly Methods

de Bruijn graph assemblers:

- SPAdes (http://cab.spbu.ru/software/spades/)
- Velvet (https://www.ebi.ac.uk/~zerbino/velvet/)

CANU (hybrid assemblies with long and short reads), HGap





Edwards and Holt 2013 *MIE*

de Bruijn Graphs

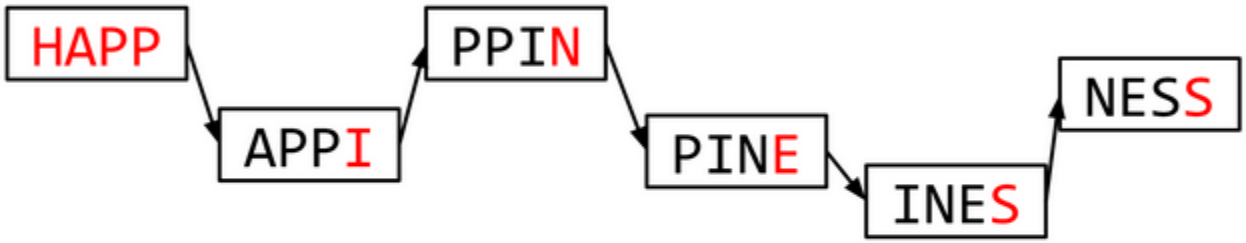
HAPPI PINE INESS APPIN

k = 4 k-mers:

HAPP APPI

PINE PPIN

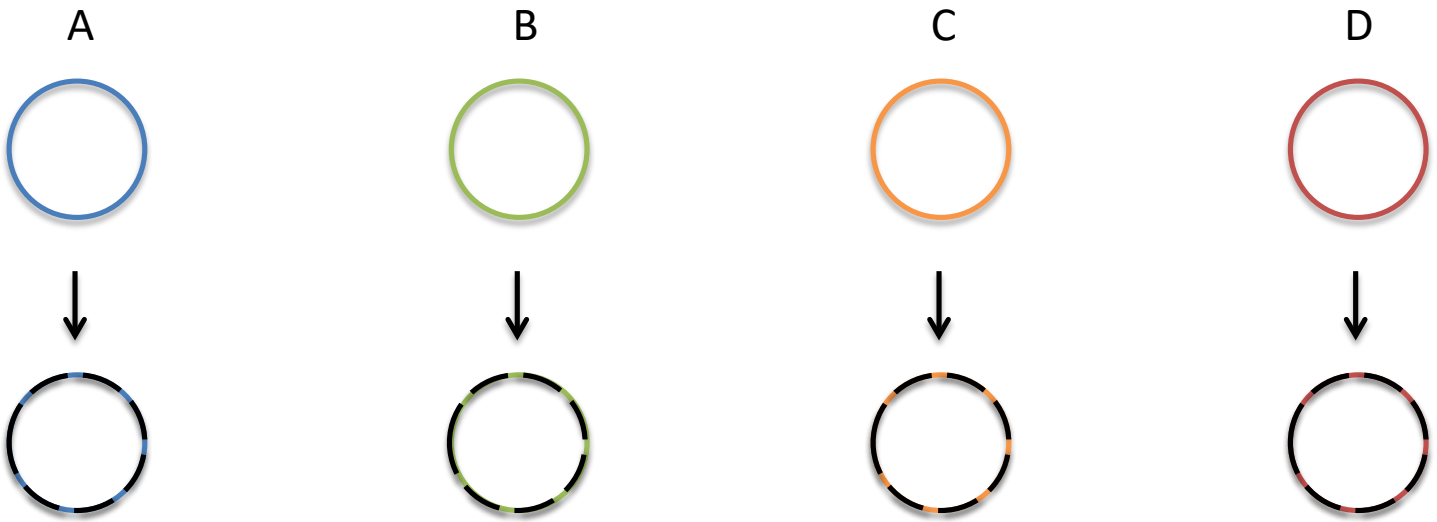
INES NESS



HAPPINESS

Assembly-Based Approach

2. Annotate each genome



Genome annotation

- A process of attaching biological information to sequences (contigs or chromosomes).
- Consists of two main steps:

  A. Identifying elements on genome a process called gene prediction (*Structural annotation*).

  B. Attaching biological information to these elements (*Functional annotation*).

Genome annotation

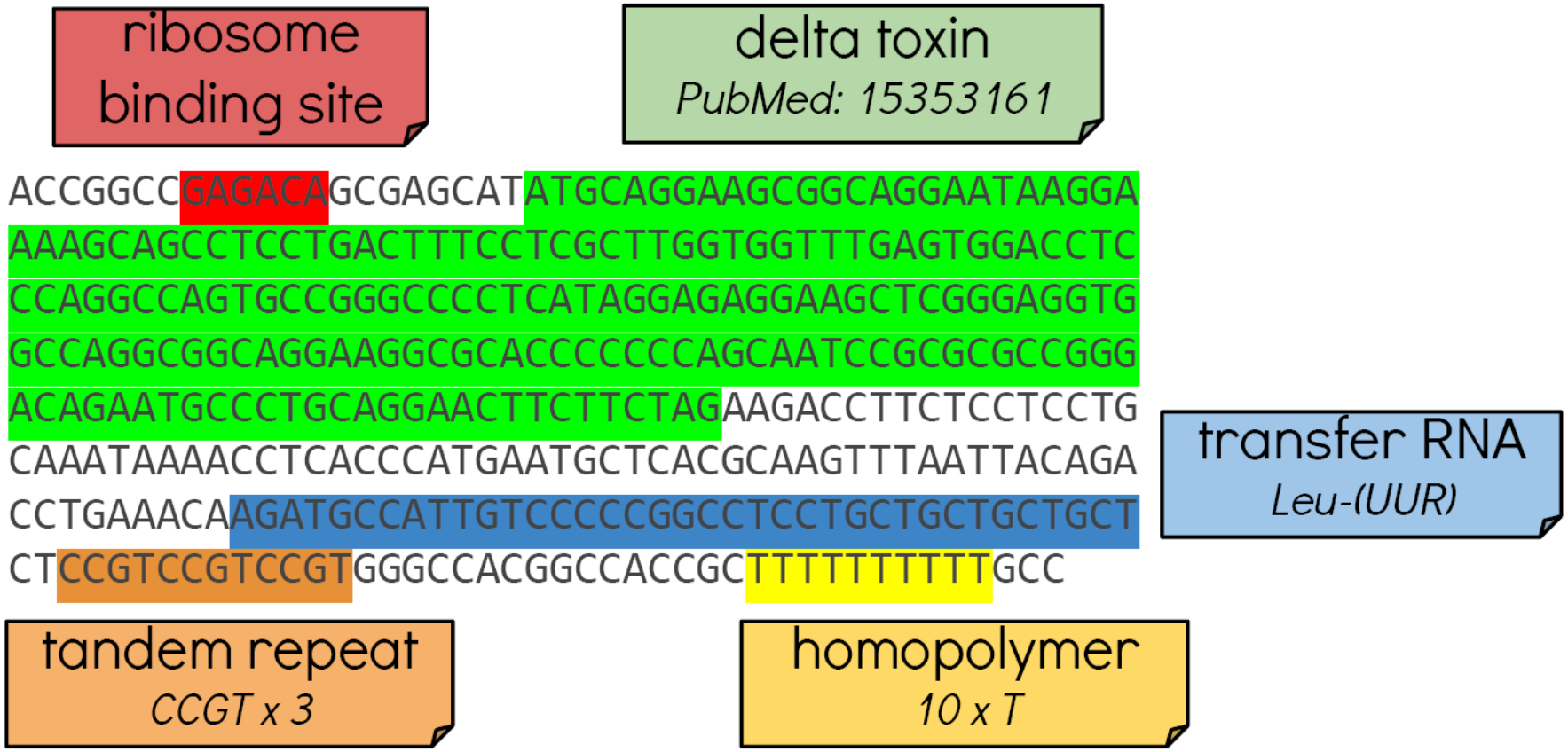
- *Structural annotation*
  - ORFs and their localisation
  - Gene structure
  - Coding regions
  - Location of regulatory motifs

```
1. ATG CAA TGG GGA AAT GTT ACC AGG TCC GAA CTT ATT GAG GTA AGA CAG ATT TAA
2. A TGC AAT GGG GAA ATG TTA CCA GGT CCG AAC TTA TTG AGG TAA GAC AGA TTT AA
3. AT GCA ATG GGG AAA TGT TAC CAG GTC CGA ACT TAT TGA GGT AAG ACA GAT TTA A
```

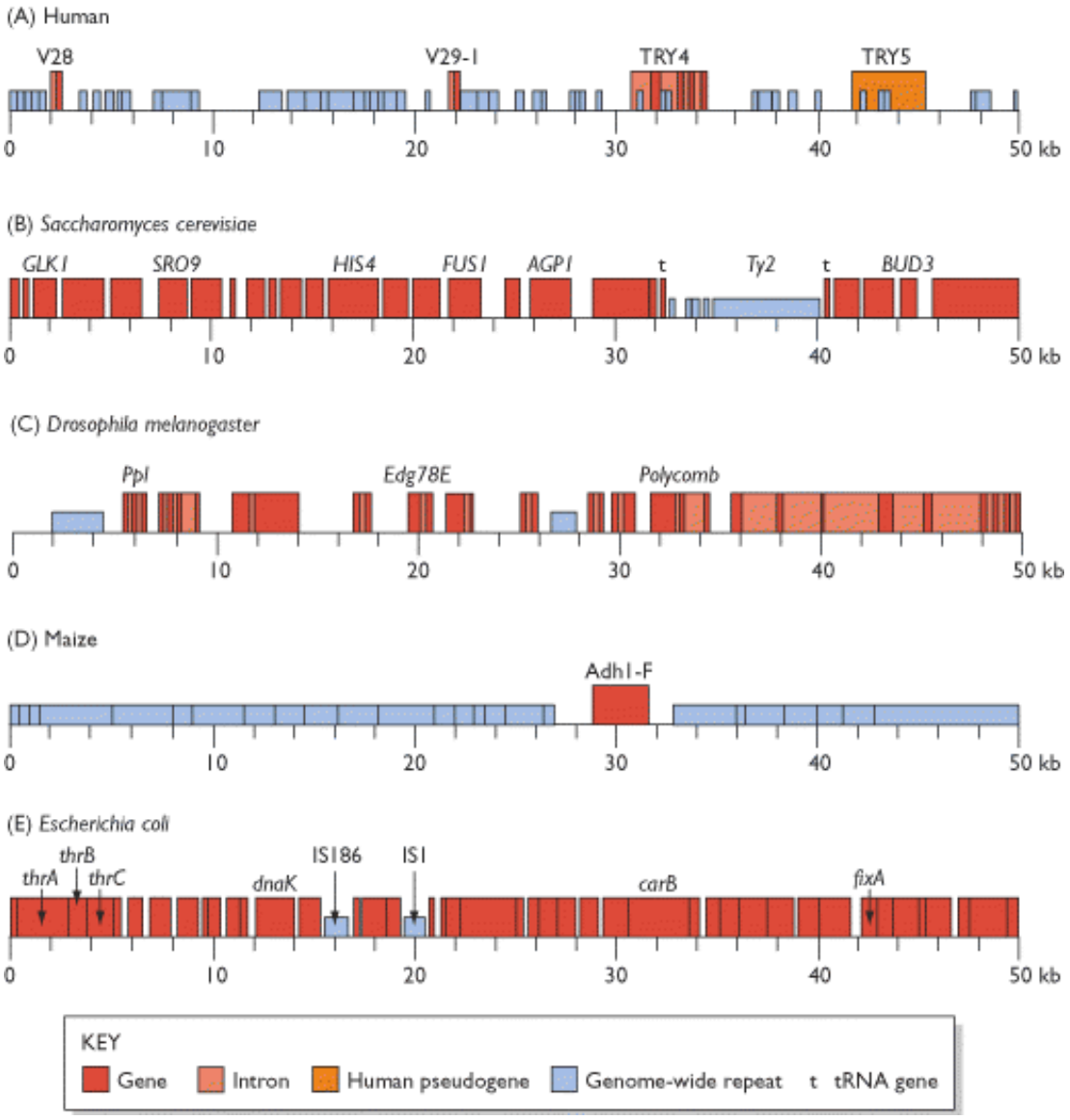
- *Functional annotation*
  - Biochemical function
  - Biological function
  - Involved regulation and interactions
  - Expression

Tools: Prodigal, ORFfinder, Prokka, RAST

Annotation: Adding biological info to sequences (using Prokka as an example)



Genomics Terminology



Brown Fig 2.2

BIOS Scientific Publishers Ltd, 1999

What is in an annotation?

- Location
  - which sequence? *chromosome 2*
  - where on the sequence? `100..659`
  - what strand? `-ve`
- Feature type
  - what is it? *protein coding gene*
- Attributes
  - protein product? *alcohol dehydrogenase*
  - enzyme code? *EC:1.1.1.1*
  - subcellular location? *cytoplasm*
  - note? *beer processing*

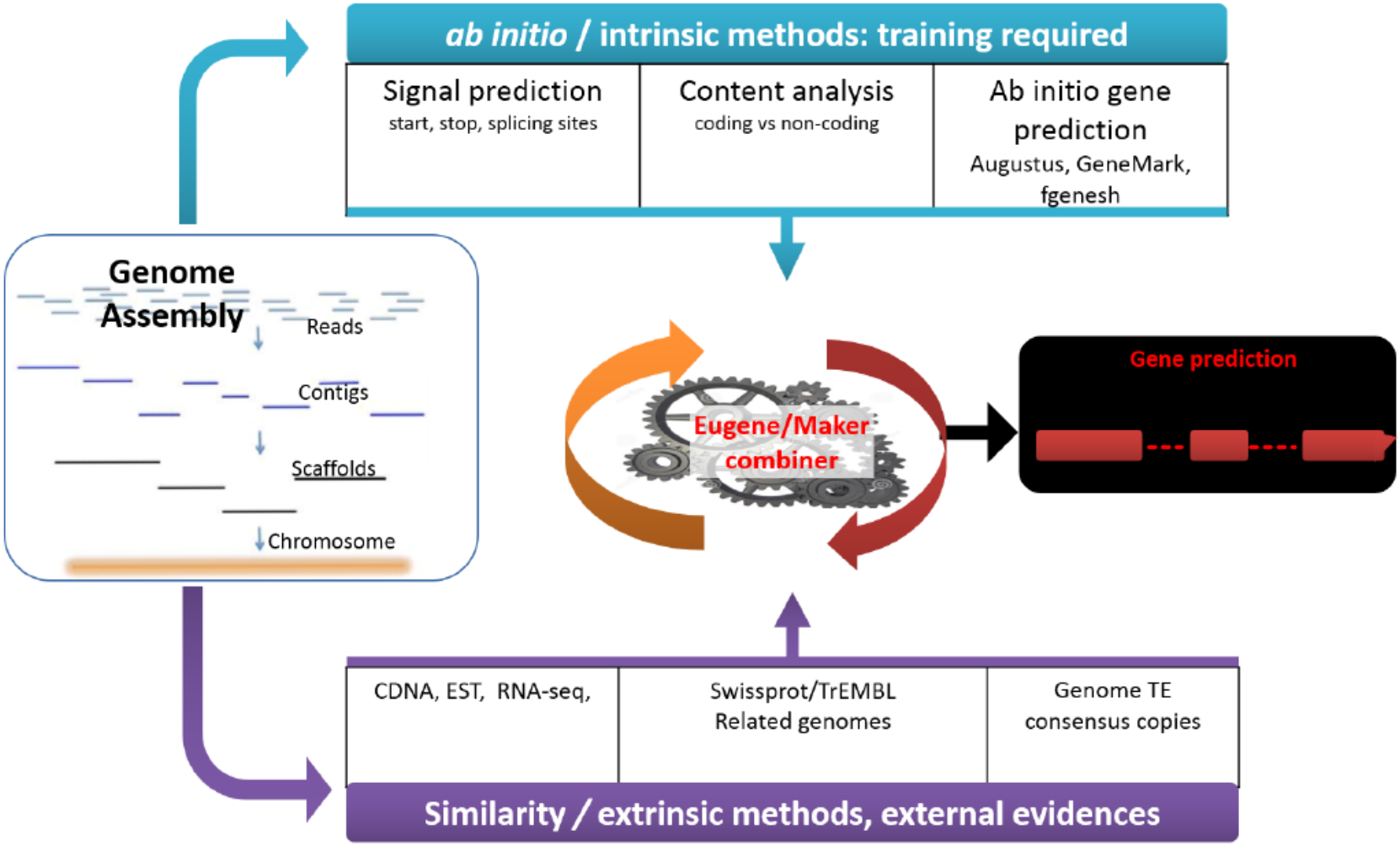
Annotation Methods

- There are different annotation algorithms for protein-coding genes, tRNAs, rRNAs, other non-coding RNAs
- Prokka (http://www.vicbioinformatics.com/software.prokka.shtml) combines all these tools

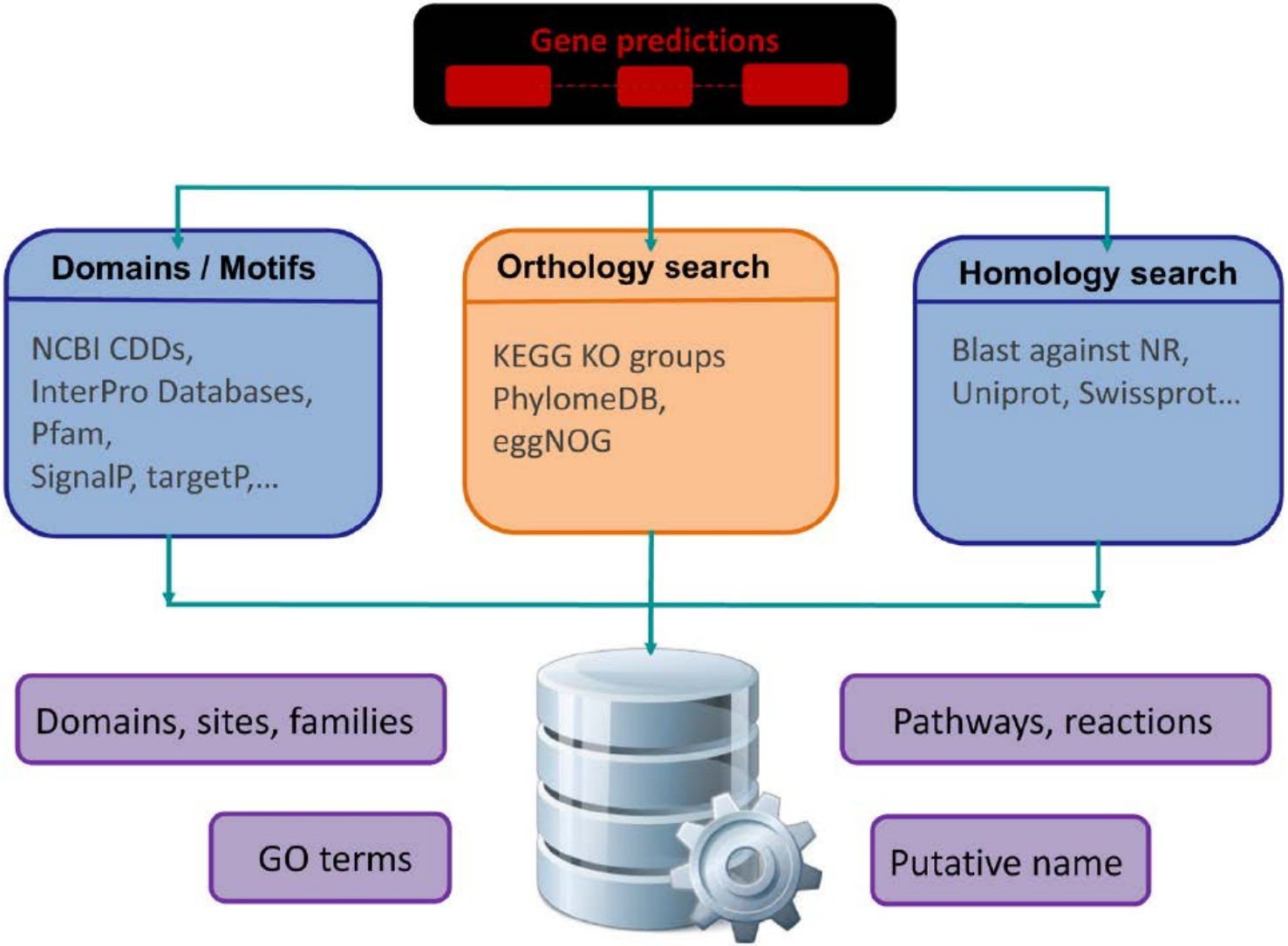
Table 1. Feature prediction tools used by Prokka

Tool (reference)	Features predicted
Prodigal (Hyatt 2010)	Coding sequence (CDS)
RNAmmer (Lagesen *et al.*, 2007)	Ribosomal RNA genes (rRNA)
Aragorn (Laslett and Canback, 2004)	Transfer RNA genes
SignalP (Petersen *et al.*, 2011)	Signal leader peptides
Infernal (Kolbe and Eddy, 2011)	Non-coding RNA

Structural genome annotation using “combiners”



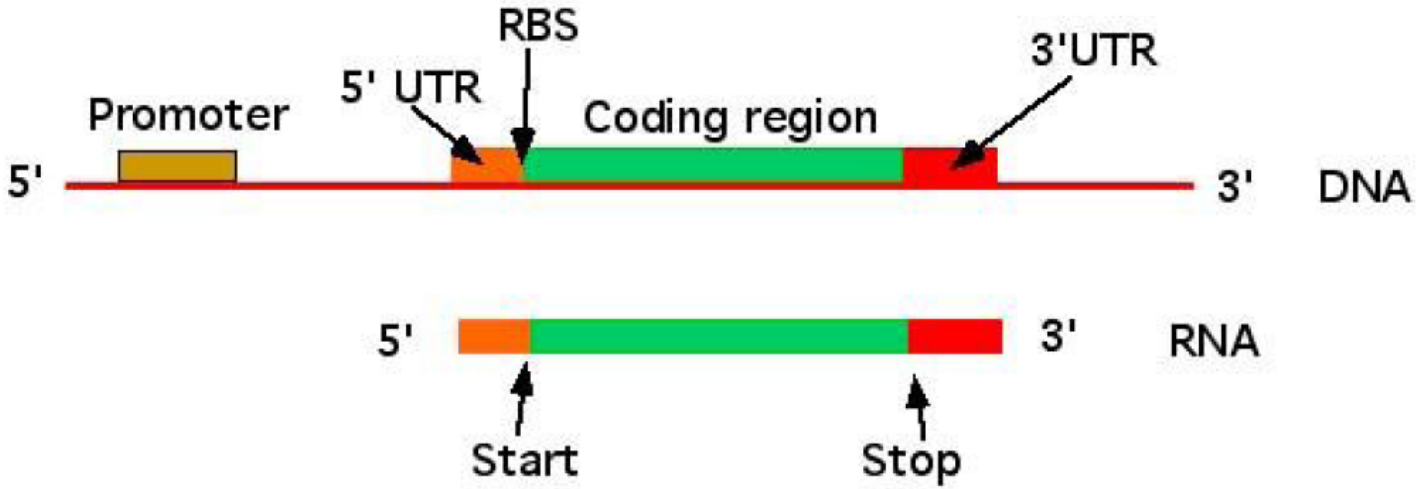
Functional genome annotation pipelines



Bacterial feature types

- protein coding genes
  - promoter (-10, -35)
  - ribosome binding site (RBS)
  - coding sequence (CDS)
    - signal peptide, protein domains, structure
  - terminator
- non coding genes
  - transfer RNA (tRNA)
  - ribosomal RNA (rRNA)
  - non-coding RNA (ncRNA)
- other
  - repeat patterns, operons, origin of replication, ...

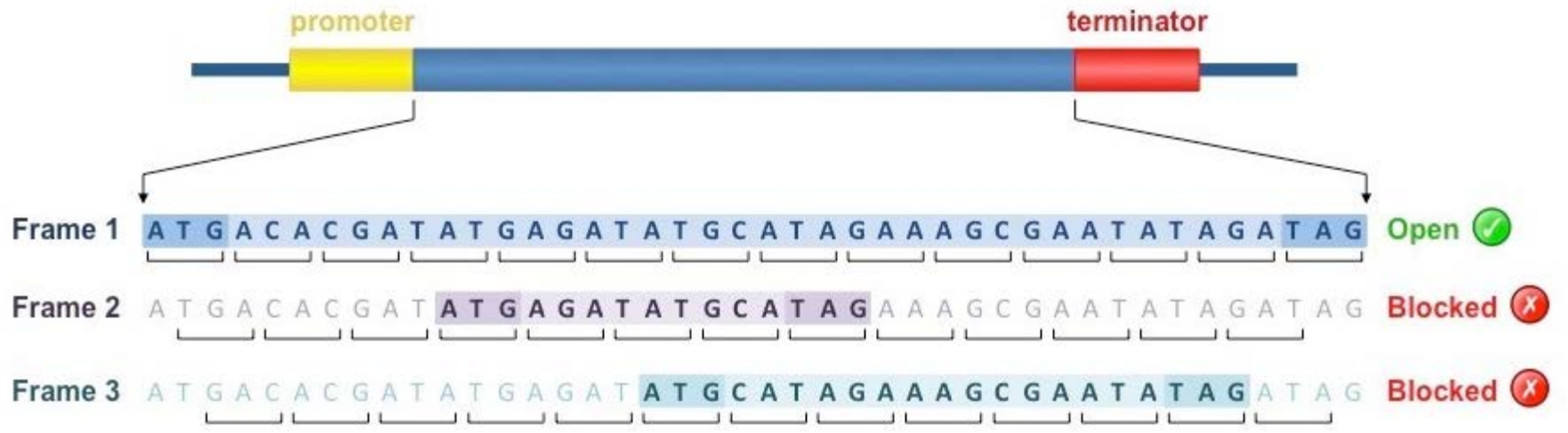
How does a bacterial gene looks like?



- have >= 3 potential start codons (species dependent)
- haploid, but lots of horizontal gene transfer
- methylation used as primitive immune system
  - restriction modification system against phage
- no introns

Identification of open reading frames

– look for ATG-Stop (+ alternatives)
– over certain size
– overlaps
– computer based and by “trained eye”



Tools: Glimmer, Orpheus

Key bacterial features

- tRNA
  - easy to find and annotate: anti-codon
- rRNA
  - easy to find and annotate: 5s 16s 23s
- CDS
  - straightforward to find candidates
    - false positives are often small ORFs
    - wrong start codon
  - partial genes, remnants
  - pseudogenes
  - assigning function is the bulk of the workload

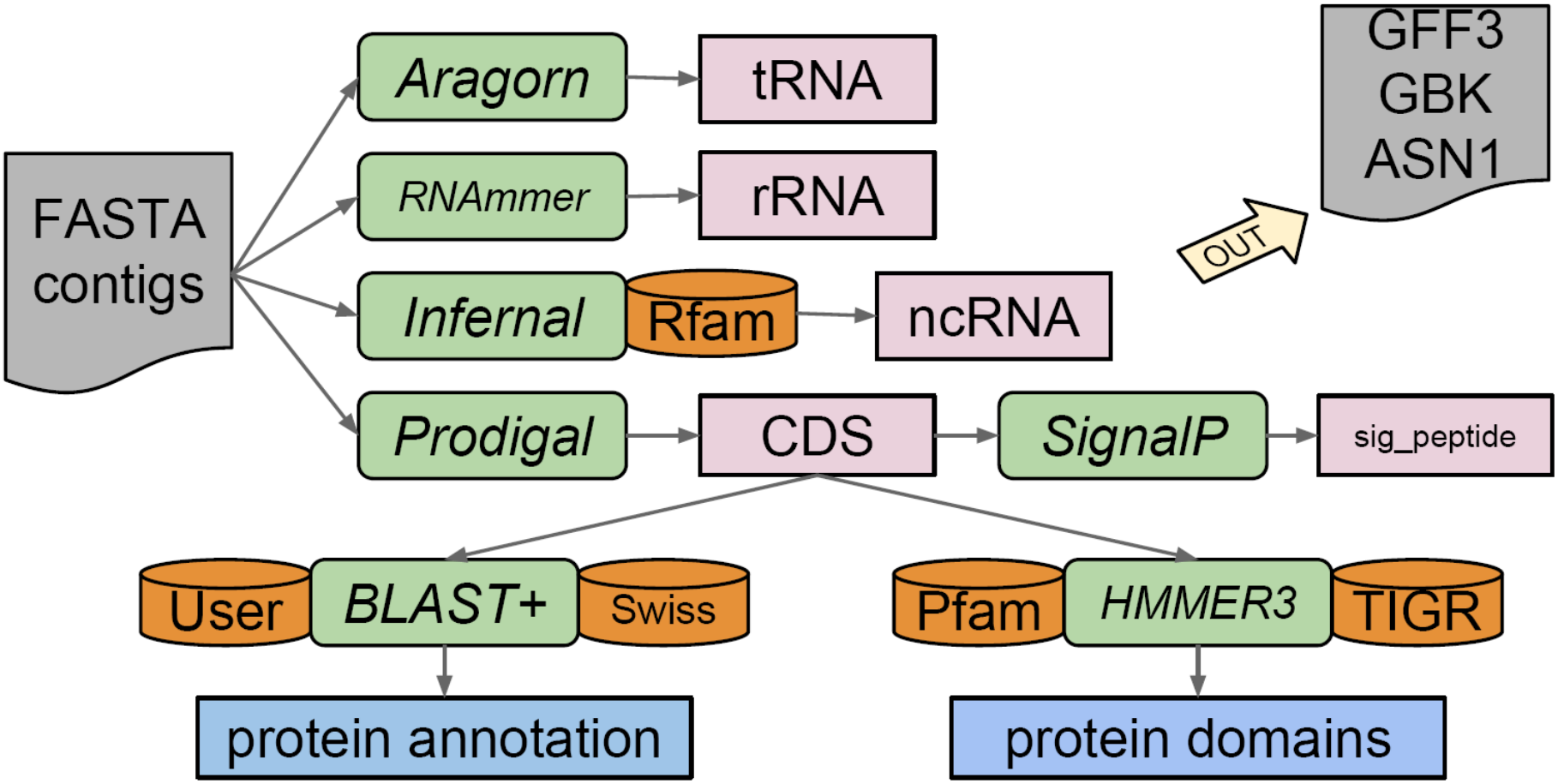
Automatic annotation

Two strategies for identifying coding genes:

- **sequence alignment**
  - find known protein sequences in the contigs
    - transfer the annotation across
  - will miss proteins not in your database
  - may miss partial proteins

- ***ab initio* gene finding**
  - find candidate open reading frames
    - build model of ribosome binding sites
    - predict coding regions
  - may choose the incorrect start codon
  - may miss atypical genes, overpredict small genes

Prokka pipeline (simplified)



Predicting protein function

Sequence similarity is a proxy for homology

- Sequence based (alignment)
  - tools: BLAST, BLAT, FASTA, Exonerate
  - databases: RefSeq, Uniprot, ...
- Model based ("fuzzy sequence" matching)
  - PSSM: position-specific scoring matrix
    - tools: RPS-BLAST, Psi-BLAST
    - databases: CDD, COG, Smart
  - HMM: hidden Markov models
    - tools: HMMER, HHblits
    - databases: Pfam, TIGRfams

Hierarchical database searching

- Facts
  - searching against smaller databases is faster
  - searching against similar sequences is faster

- Idea
  - start with small set of close proteins
  - advance to larger sets of more distant proteins

- Prokka
  - your own custom "trusted" set (optional)
  - **core bacterial proteome (default)**
  - genus-specific proteome (optional)
  - whole protein HMMs: PRK clusters, TIGRfams
  - protein domain HMMs: Pfam

Core bacterial proteome

- Many bacterial proteins are conserved
  - experimentally validated
  - small number of them
  - good annotations
- Prokka provides this database
  - derived from UniProt-Swissprot
  - only bacterial proteins
  - only accept evidence level 1 (aa) or 2 (RNA)
  - reject "Fragment" entries
  - extract /gene /EC_number /product /db_xref
- First step gets ~50% of the genes
  - BLAST+ blastp, multi-threading to use all CPUs

The flexible genome content

- Prokka has genus-specific databases
  - aim to capture "genus-specific" naming conventions
  - derived from proteins in completed genomes
  - proteins are clustered and majority annotation wins
  - some annotations are rubbish though
- Existing model databases
  - Pfam, TIGRfams are well curated

Provenance

Recording where an annotation came from

Prokka uses Genbank "evidence qualifier" tags:

Wet lab

```
/experiment="EXISTENCE:Northern blot"
```

Dry lab

```
/inference="similar to DNA sequence:INSD:AACN010222672.1"
/inference="profile:tRNAscan:2.1"
/inference="protein motif:InterPro:IPR001900"
/inference="ab initio prediction:Glimmer:3.0"
```

Example from Prokka

Feature Type:

tRNA

Location:

contig000341 @ 655..730 +

Attributes:

/gene="tRNA-Leu(UUR)"

/anticodon=(pos:678..680,aa:Leu)

/product="transfer RNA-Leu(UUR)"

/inference="profile:Aragorn:1.2"

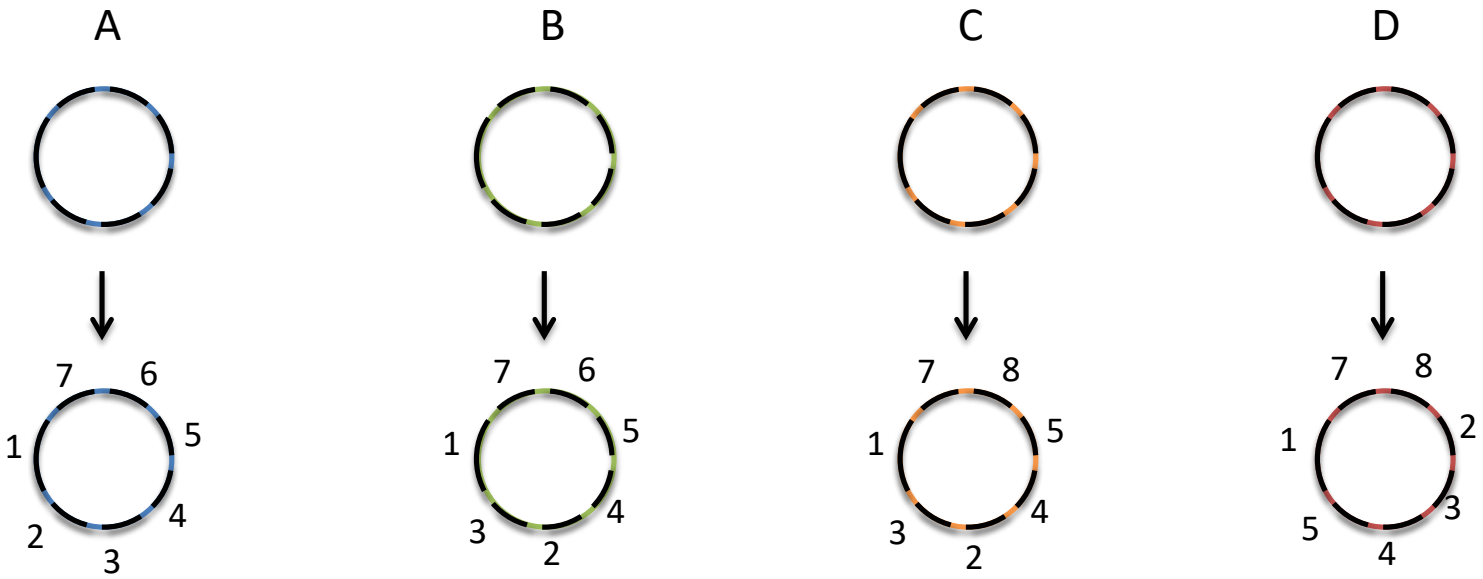
Improving Annotation

- Some annotations are wrong
  - False annotation
  - Missing annotation
  - Partially wrong annotation

- Curation
  - Manual effort to improve annotations
  - Community curation

Assembly-Based Approach

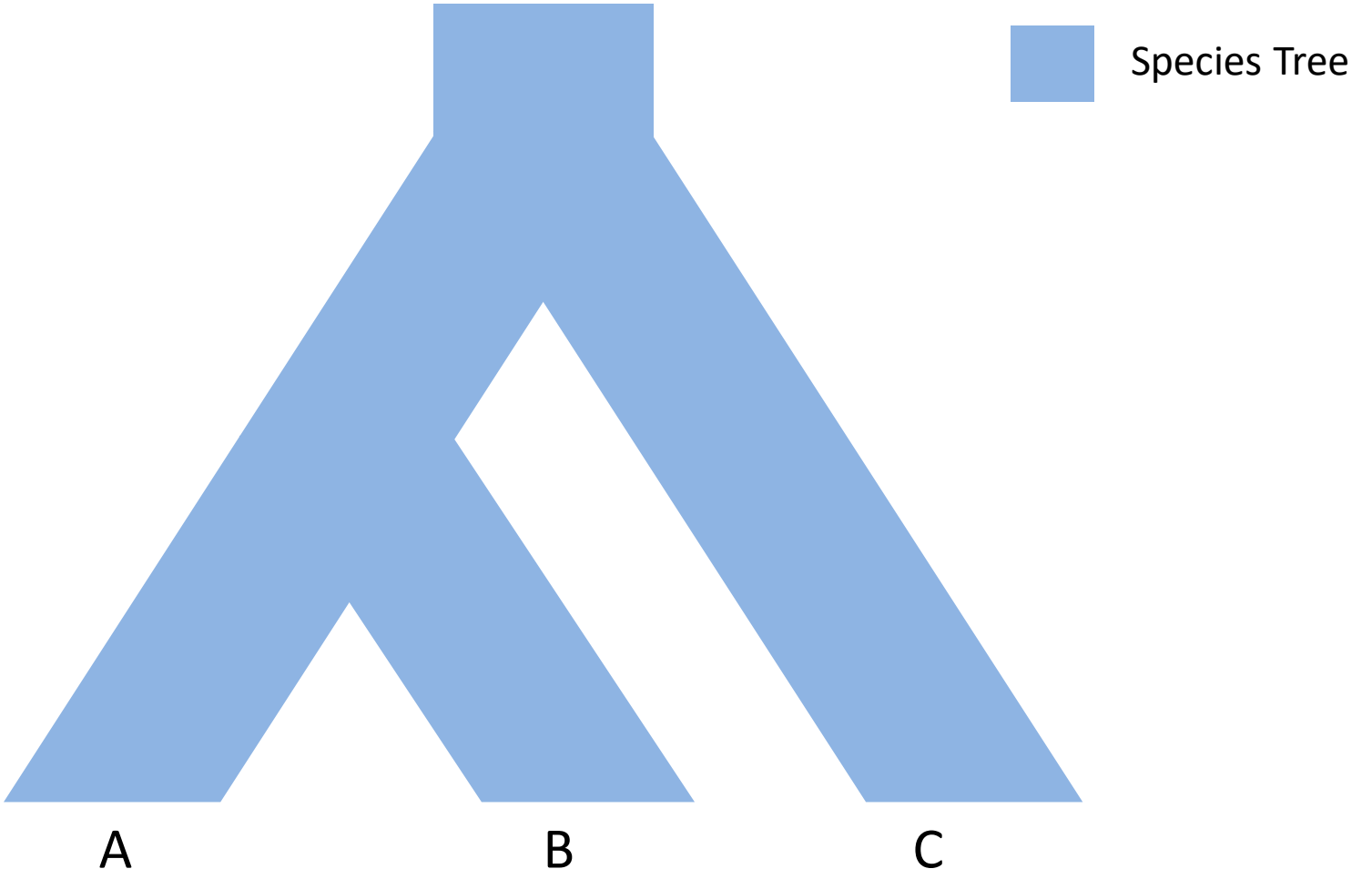
2. Cluster genes



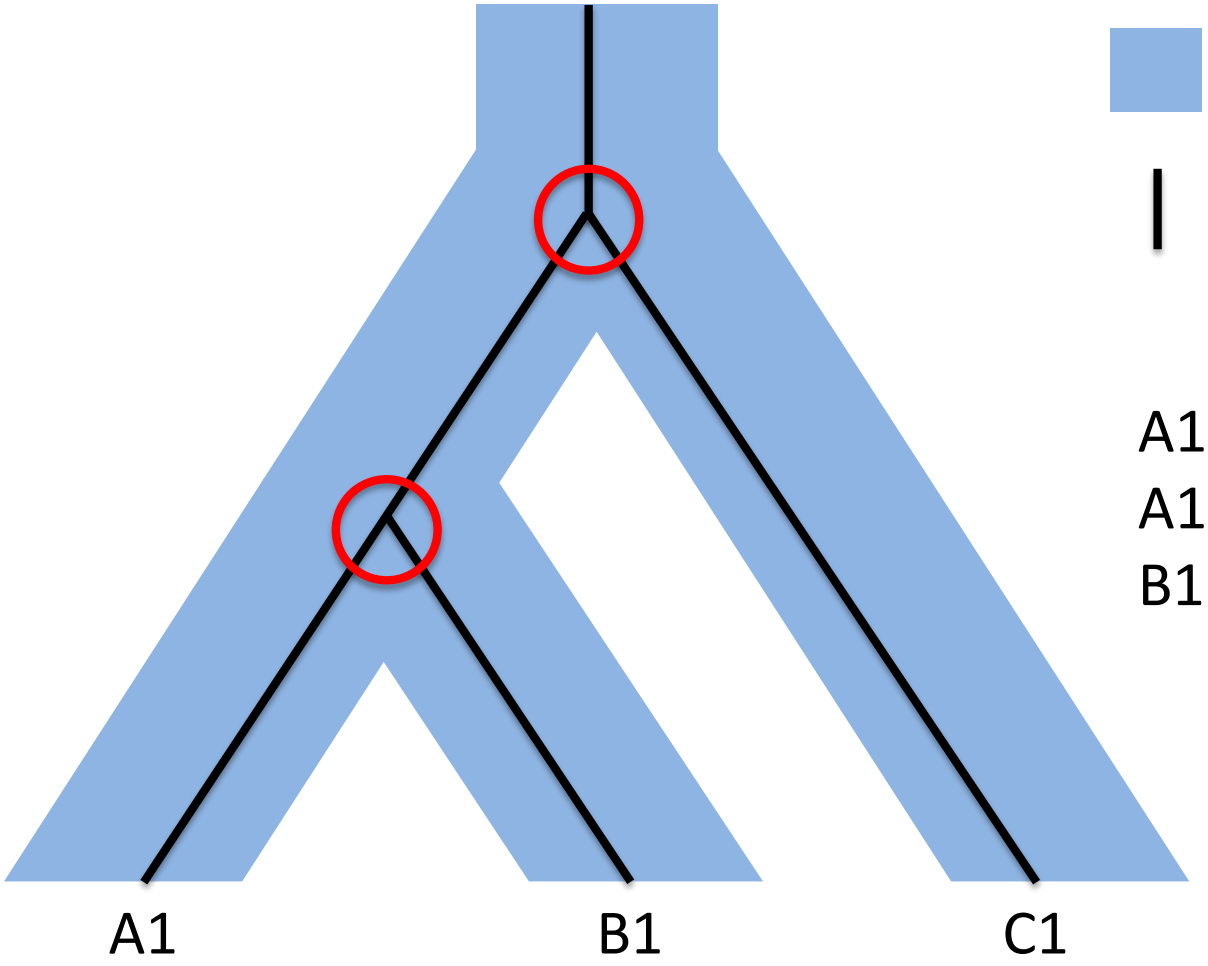
Orthology

- Orthologs are genes whose most recent divergence was a speciation event
- Paralogs are genes whose most recent divergence was a gene duplication event
- Groups of orthologous and paralogous genes are termed “ortholog clusters” or “gene clusters” or even just “genes” and form the basis of all gene-based comparative genomics

Gene Trees vs Species Trees



Gene Trees vs Species Trees

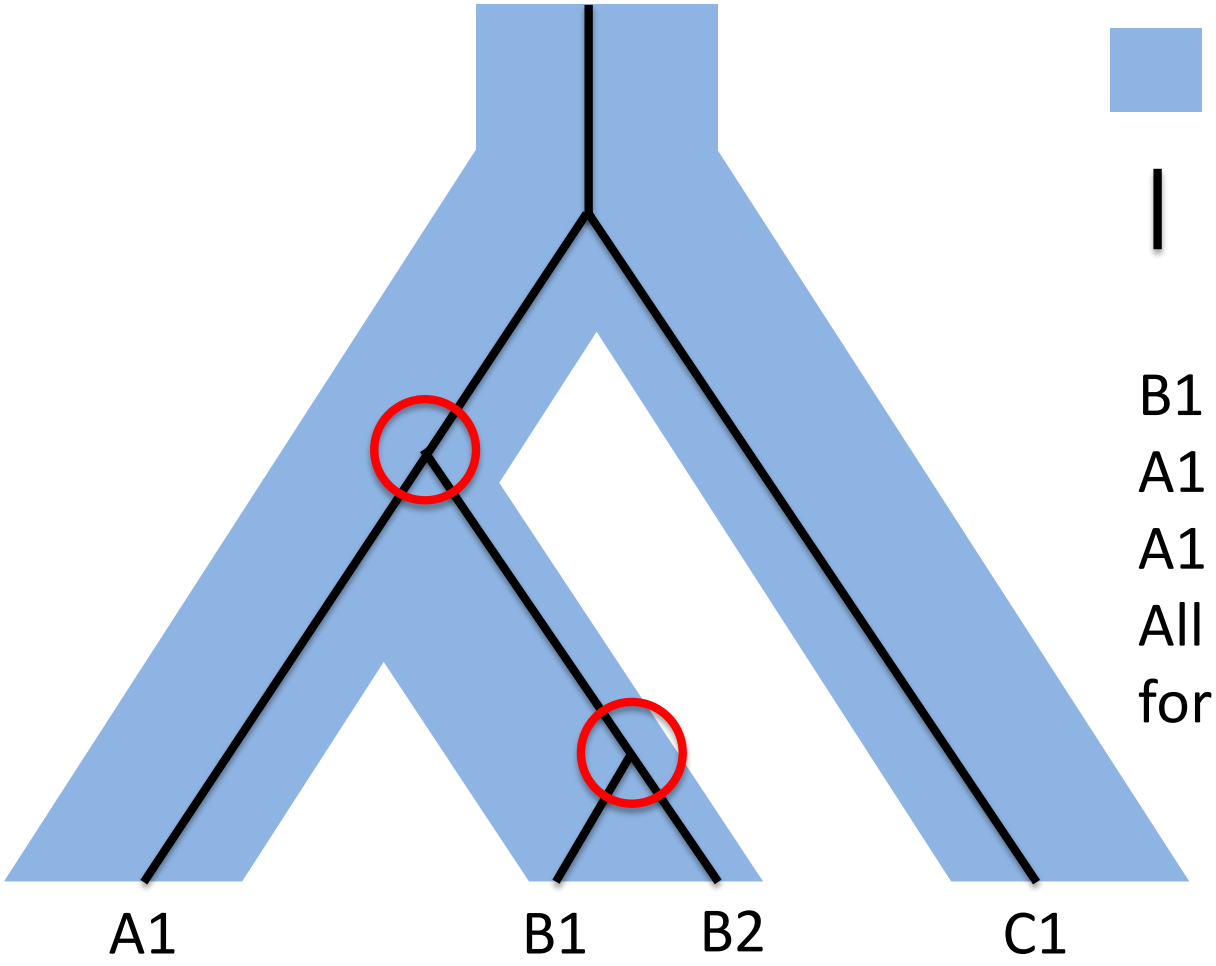


Species Tree

Gene Tree

A1 and B1 are orthologs
A1 and C1 are orthologs
B1 and C1 are orthologs

Gene Trees vs Species Trees



Species Tree

Gene Tree

B1 and B2 are paralogs
A1 and B1 are orthologs
A1 and B2 are orthologs
All of these genes would form a single gene cluster

Gene Names, Orthology, and Function

- Does strain A have an ortholog of gene X? (where gene X is characterized in another strain)

- If two genes are orthologs, that do not necessarily have same function, but they often do

- If two genes are paralogs, they are traditionally thought to often differ in function, and paralogy is thought to be one of the main sources of “new” genes

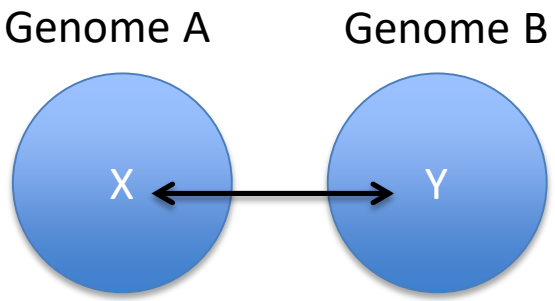
Gene Clustering

- Assess the similarity of every gene to every other gene
  - e.g., using BLAST
- Use that similarity to join pairs of genes
  - e.g., using Reciprocal Best Hits
- Connect the gene pairs into larger clusters
  - e.g., using Reciprocal Best Hits or Markov clustering

Pairwise Clustering - Reciprocal Best Hits

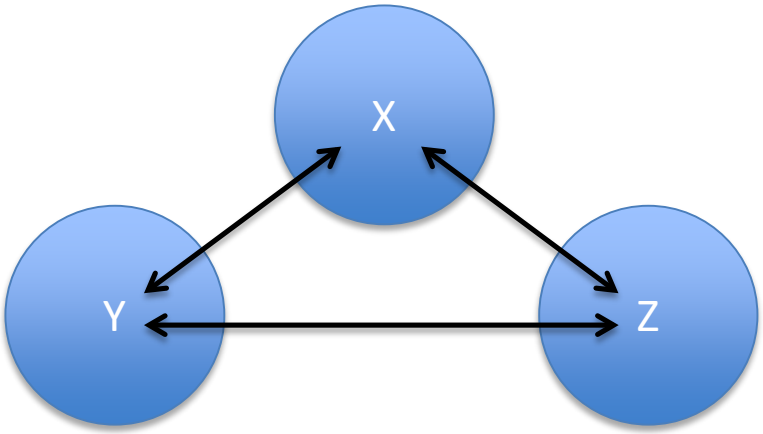
- Reciprocal Best Hits (RBH) is a simple and popular clustering algorithm

- Two proteins X and Y from species A and B, respectively, are considered orthologs if protein X is the best BLAST hit for protein Y and protein Y is the best BLAST hit for protein X (i.e., they are reciprocal best hits)



Clustering - Reciprocal Best Hits

- The logic of RBH can then be extended from pairs of genomes to three or more genomes
  - i.e., Three proteins X, Y, and Z, respectively, from species A, B, and C, respectively, are considered orthologs if each protein is the best BLAST hit for each protein all genomes



- Addition of paralogs is not part of the RBH algorithm, but can be done as post-processing step

Clustering - OrthoMCL

- OrthoMCL is an extremely popular gene clustering program
- OrthoMCL uses reciprocal best hits to identify orthologs between pairs of genomes
- Beyond genome pairs, it uses a Markov cluster algorithm (MCL) to assemble groups of orthologs and paralogs
- It does not scale well to hundreds of genomes, so as sequencing throughput continues to increase, OrthoMCL is losing popularity

Gene Content Profiles

- Orthologous gene clusters can be used to build gene content profiles - binary coding of gene presence/absence across genomes
- These profiles can then be easily queried to identify genes unique to a given set of genomes
  - easily identifies clade-specific genes
  - can also look for perfect correlations of genes with phenotypes

	Species A	Species B	Species C	Species D
Cluster W	1	1	0	0
Cluster X	0	0	1	1
Cluster Y	1	1	1	0
Cluster Z	1	1	1	1

Gene Content Profiles

	Species A	Species B	Species C	Species D	Profile Type
Cluster S	1	1	1	1	Single copy core
Cluster T	1	2	2	1	Multi-copy core
Cluster U	1	1	0	0	Auxillary
Cluster V	2	0	0	0	Unique

- Cluster terminology:
  - Core = orthologs are present in all genomes
  - Auxillary = genes with orthologs in at least two genomes but not all genomes
  - Unique = genes without orthologs
  - Sum of all of these genes is called the “pan genome”

  - Single-copy = genes without paralogs in any genome
  - Multi-copy = genes with paralogs in at least one genome

Organismal Phylogenies

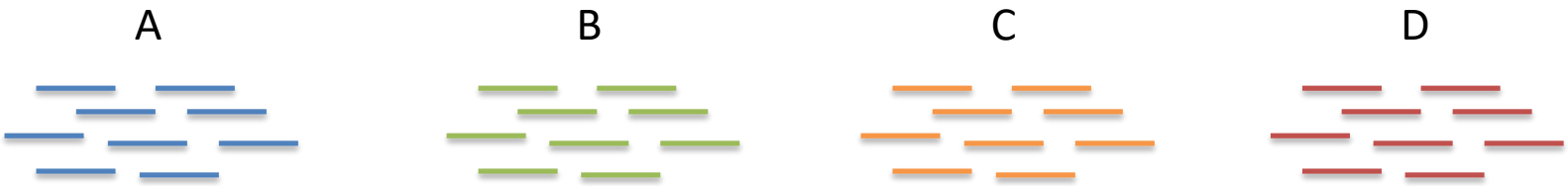
- Single-copy core genes are often used to create organismal phylogenies
- Genes can be aligned with MUSCLE or CLUSTAL
- Then sequences are concatenated, or attached together end-to-end, so that the end of gene A is followed by the beginning of gene B
- Then a phylogeny is generated using available software like RAxML or FastTree
- CAVE: horizontal gene transfer!

Other potential downstream analyses

- Look for rapidly evolving genes by calculating evolutionary rates
- Functional enrichment of genes specific to a clade
- Association tests of gene presence/absence with a specific phenotype

Two Approaches to Microbial Genomics

Starting with sets of reads representing your study isolates…



<u>Assembly-based</u>

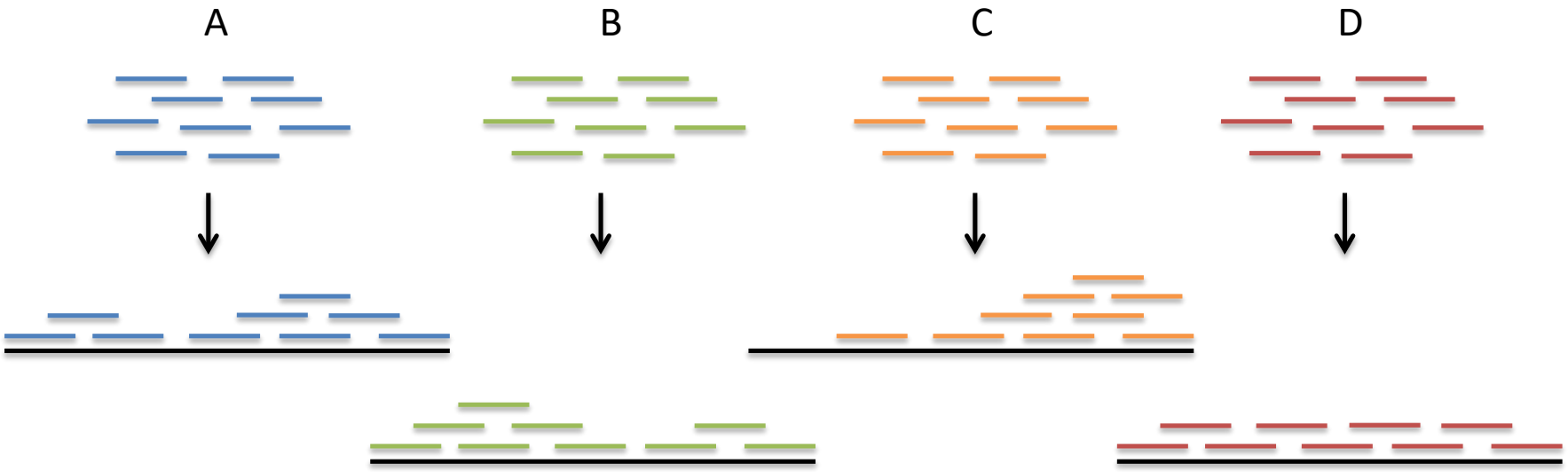
1. Assemble each set of reads into a genome sequence
2. Annotate each genome
3. Cluster genes and compare between each genome

<u>Variant-based</u>

1. Compare each read set to a reference genome assembly
2. Directly compare variants between each genome

Variant-Based Approach

1. Align reads to a reference genome



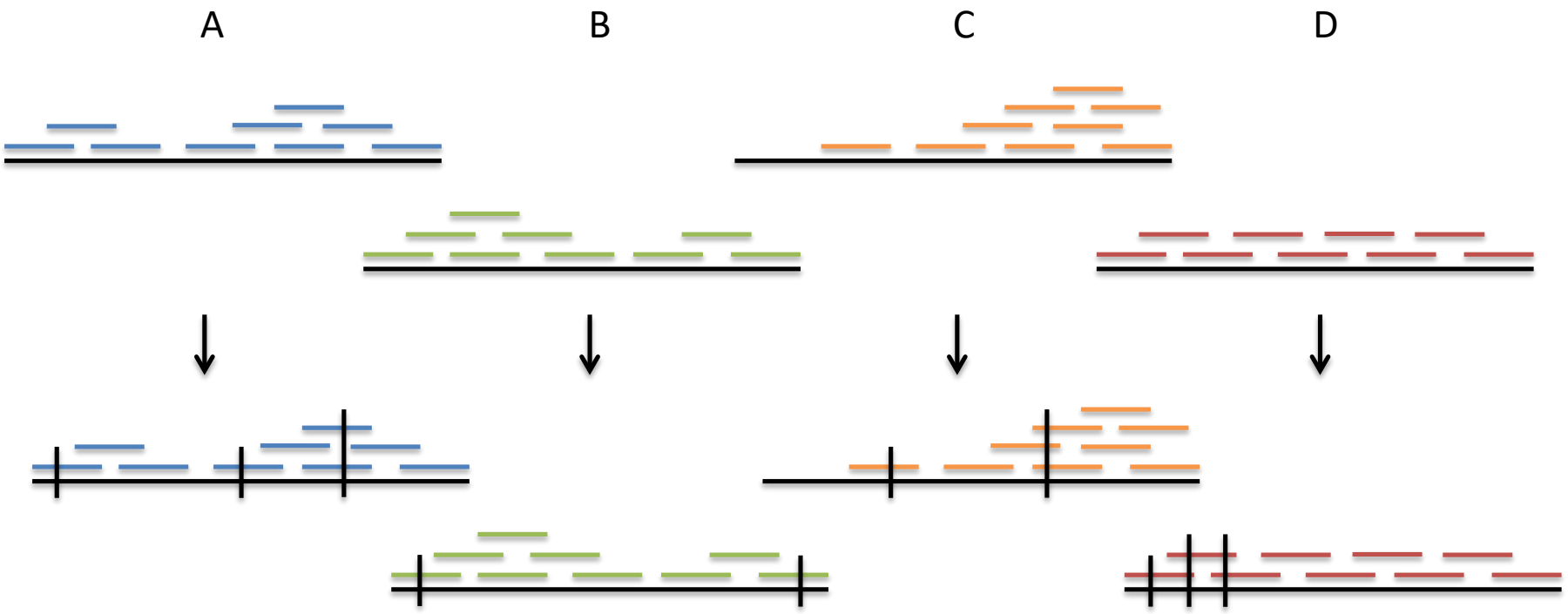
Read Alignment Methods

- Goal: to find the best match or matches of a read to reference genome
- While it seems simple, it's actually a difficult problem since you cannot check all possibilities (need heuristics)

- Un-spliced aligners (DNA to DNA, cDNA to cDNA)
  - BWA (http://bio-bwa.sourceforge.net/)
  - Bowtie2 (http://bowtie-bio.sourceforge.net/bowtie2/index.shtml)

Variant-Based Approach

2. Call variants



Variants

- Single nucleotide polymorphisms (SNPs)

```
Ref AGGTCGT
Alt AGGCCGT
```

- Insertion

```
Ref AGGT---CGT
Alt AGGTCCCCGT
```

- Deletion

```
Ref AGGTCGT
Alt AGG-CGT
```

- Substitution

```
Ref AGGTATGCGT
Alt AGGCCC-CGT
```

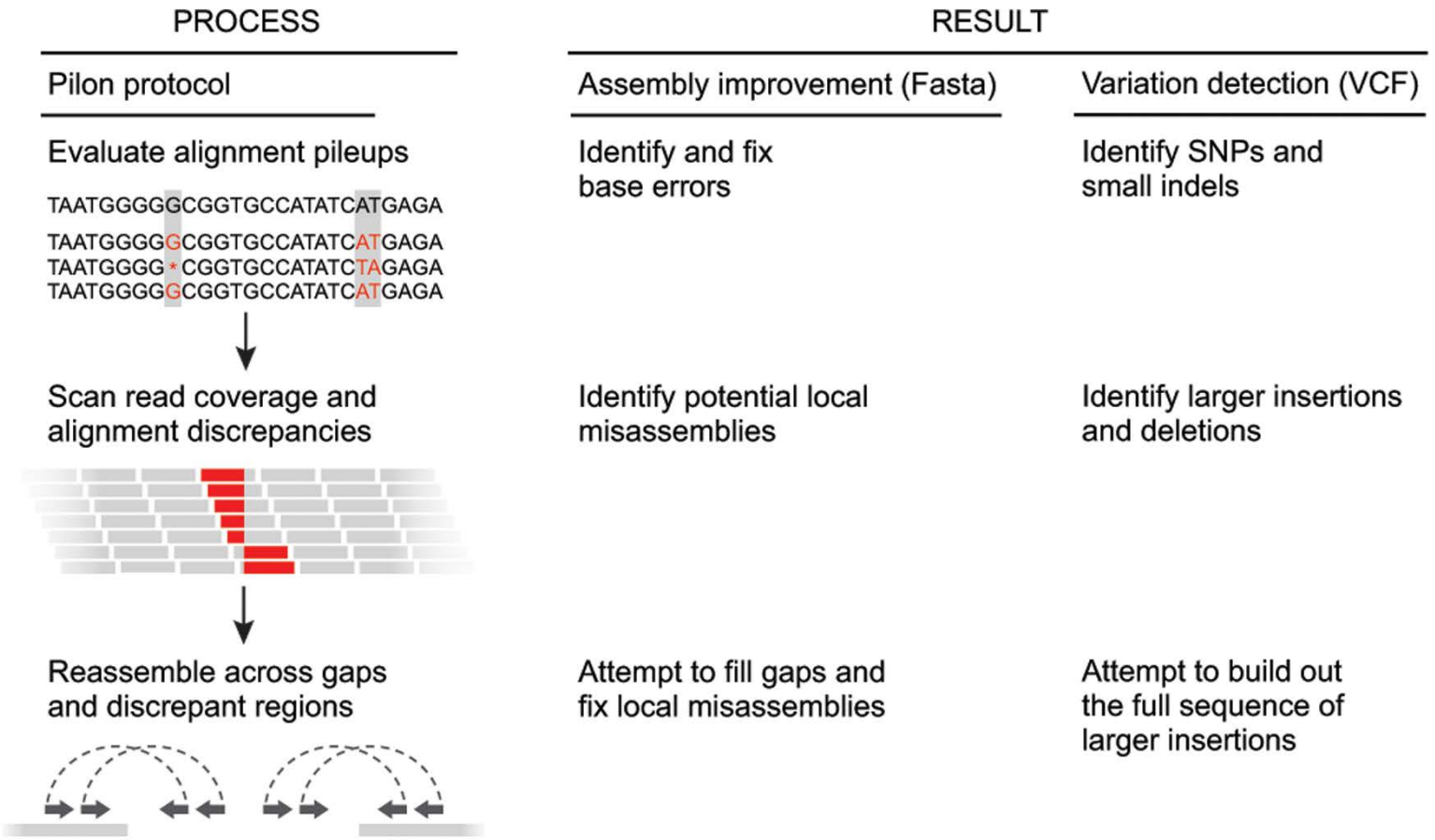
Variant Calling Methods

- Variant calling process: decide which differences in an alignment to a reference represent real differences and not errors in alignment or sequencing

- Pilon (https://github.com/broadinstitute/pilon/wiki):
  - Program for assembly improvement and also SNP calling
  - Initially developed for haploid genomes but now also works on diploid genomes
  - Uses internal heuristics for quality control

- GATK (https://software.broadinstitute.org/gatk/):
  - Program for SNP calling only
  - Initially developed for diploid genomes but has been adapted to other ploidies
  - Requires “truth set” or hard filters for quality control

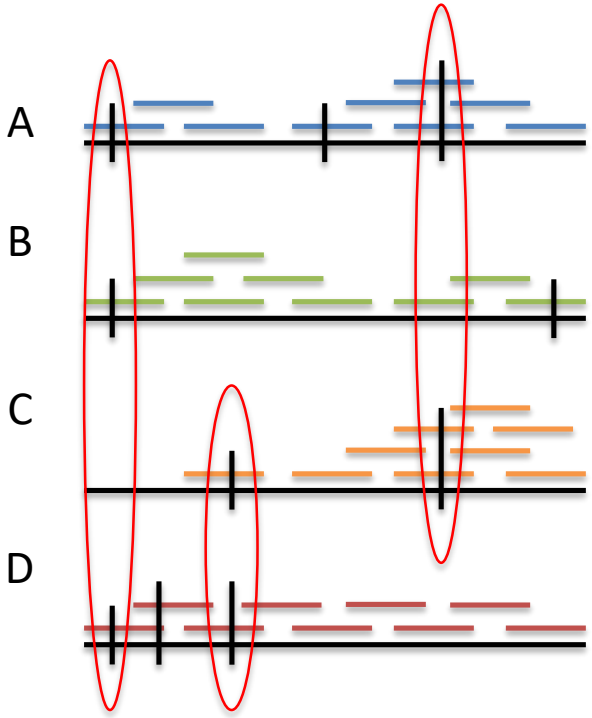
Pilon



Walker, Abeel, *et. al* 2014 *PLOS ONE*

Variant-Based Approach

3. Compare variants directly



Downstream analyses of variants

- Annotation of variant effects
  - Captures very different information than gene presence/absence: nonsynonymous and synonymous changes, frameshifts and introduced stop codons, promoter mutations
- SNP-based phylogenetic analysis
- SNP-based analysis of evolutionary rates
- Enrichment of variant types in specific sets of genes
- Association tests of variants with a specific phenotype (GWAS)

Exploring deeper lineages

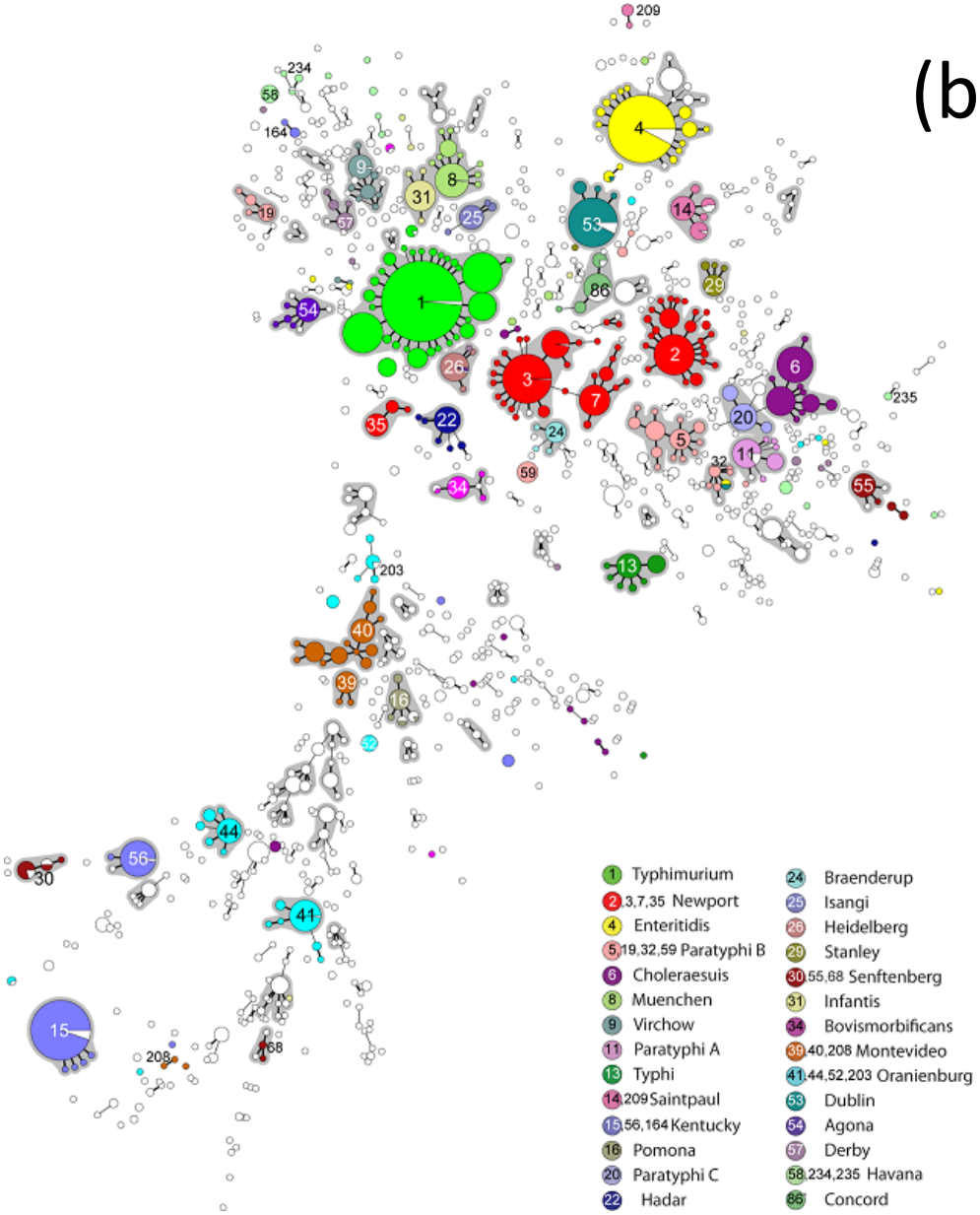
- Typing methods based around antigenicity, pathotyping and other typing methods, some of which are the *de jure* standard in many reference labs, do not always correlate with the relativity of individual strains.
- A bacterial species consists of multiple discrete lineages; to treat it as uniform is misleading.
- To place strains within a population, a neutral set of markers from across the genome should be used.
- Increasing the number of genes, or the use SNPs, as the informative sites, increases the resolution.

Sequence-based typing schemes

MLST Classic	Ribosomal MLST	Core Genome MLST
7-8 Loci	53 Loci	~ 1500-3000 for Salmonella
Conserved Housekeeping genes	Ribosomal proteins	Any conserved coding sequence
Highly conserved; Low resolution	Highly conserved; Medium resolution	Variable; High resolution
Different scheme for each Species/genus	Single scheme across tree of life	Different scheme for each Species/genus

Salmonella enterica population structure

(based on MLST)



Deciding on an Approach

- Does my reference contain most of my genes of interest?
- Are my strains closely related to a reference (>=95% identity)

- If the answer to both questions is yes, the variant-based approach is favored
- If the answer to either question is no, the assembly-based approach is favored

Pros and Cons of Approaches

Assembly-based

- \- Results are not directly comparable and must be clustered
- \- Large number of steps increases chance of error (n + n + (n-1)!)
- \+ Captures unique regions in each strain
- \+ Works on both closely and distantly related strains

Variant-based

- \+ Can compare variants directly without clustering
- \+ Small number of steps decreases chance of error (1 + 1 + n)
- \- Only captures regions present in reference
- \- Works only on closely related strains

Genome-Wide Association Studies (GWAS)

Basic anatomy of GWAS:

- Count alleles for each polymorphic site
- Evaluate allele with Chi-squared or Fisher's exact test
- Correct for multiple comparisons

Countless more complex variations of GWAS exist

Fundamentally the same idea as an "enrichment test"

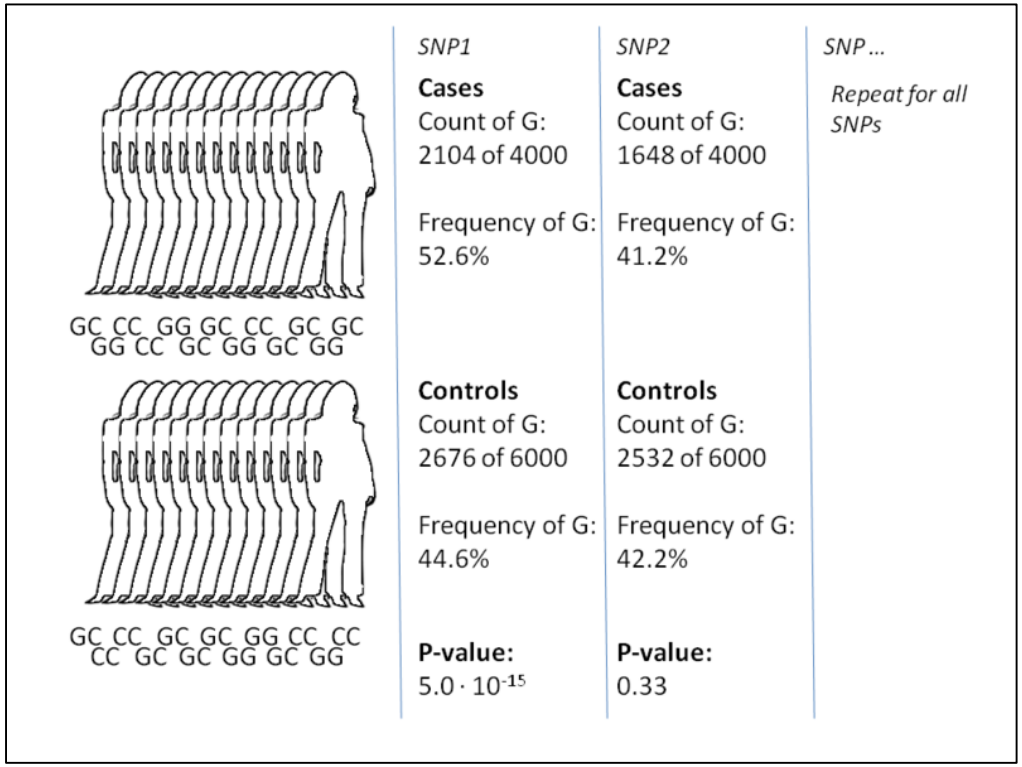


image from Wikipedia

Bacteria and GWAS

- Most GWAS methods depend on linkage disequilibrium being slowly broken up by meiotic recombination, such that alleles physically distant from each other are independent

- Many bacteria have limited or no recombination, making GWAS difficult

- Adapting GWAS to bacteria is an active area of research

"Tech support says the problem is located somewhere between the keyboard and my chair."