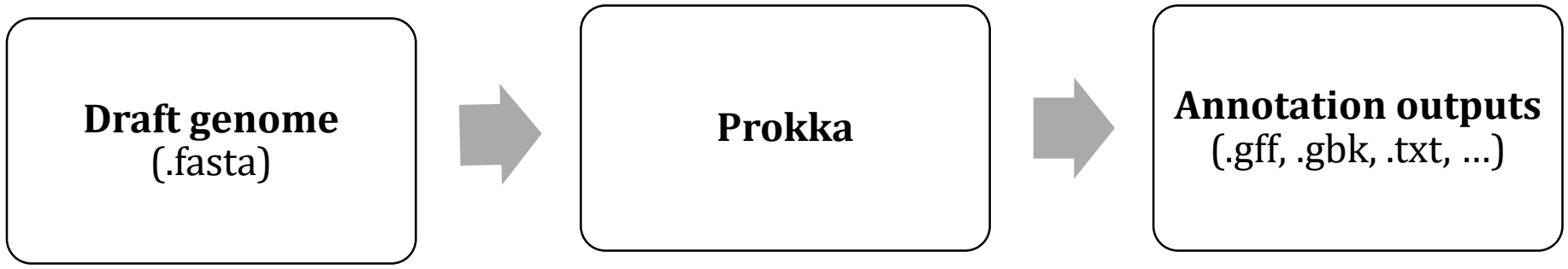
Prokka: rapid prokaryotic genome annotation

Seemann Torsten, *Bioinformatics* 30.14 (2014): 2068-2069.

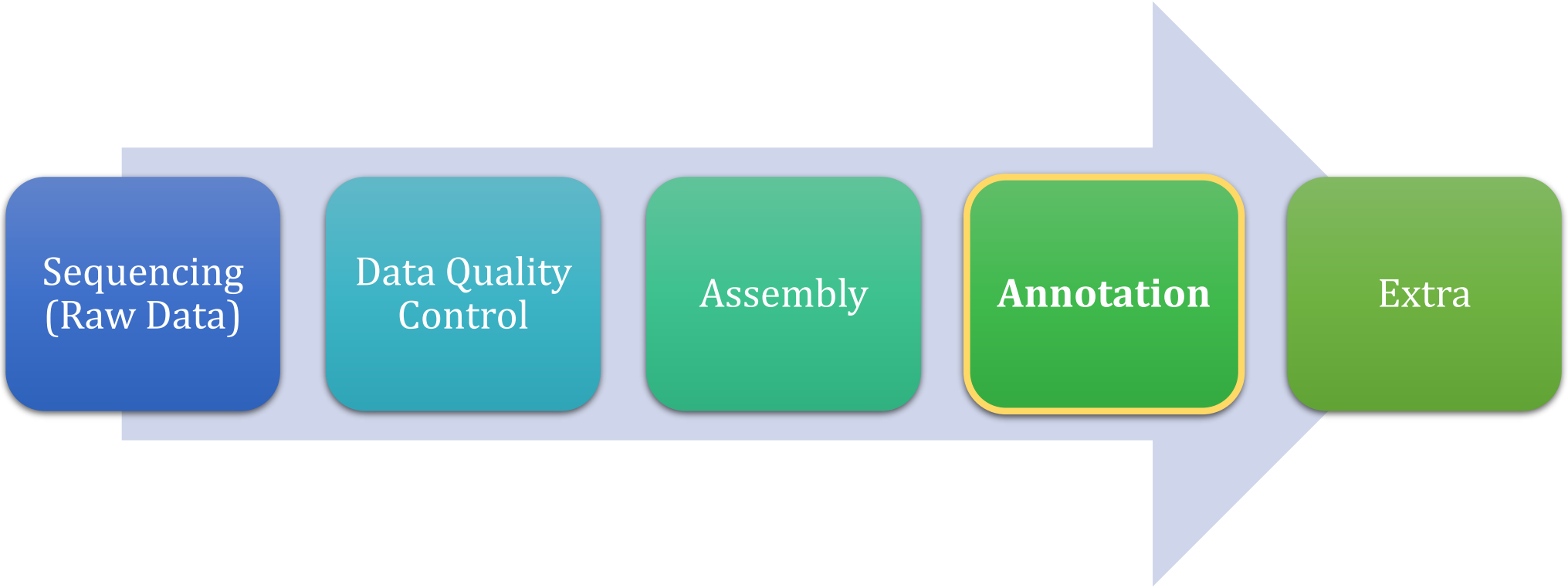
Presented by Sohyoung Won

What is Prokka?

- **Prok**aryotic **a**nnotation
- A command line software tool to annotate bacterial, archaeal and viral genomes quickly and produce standards-compliant output files

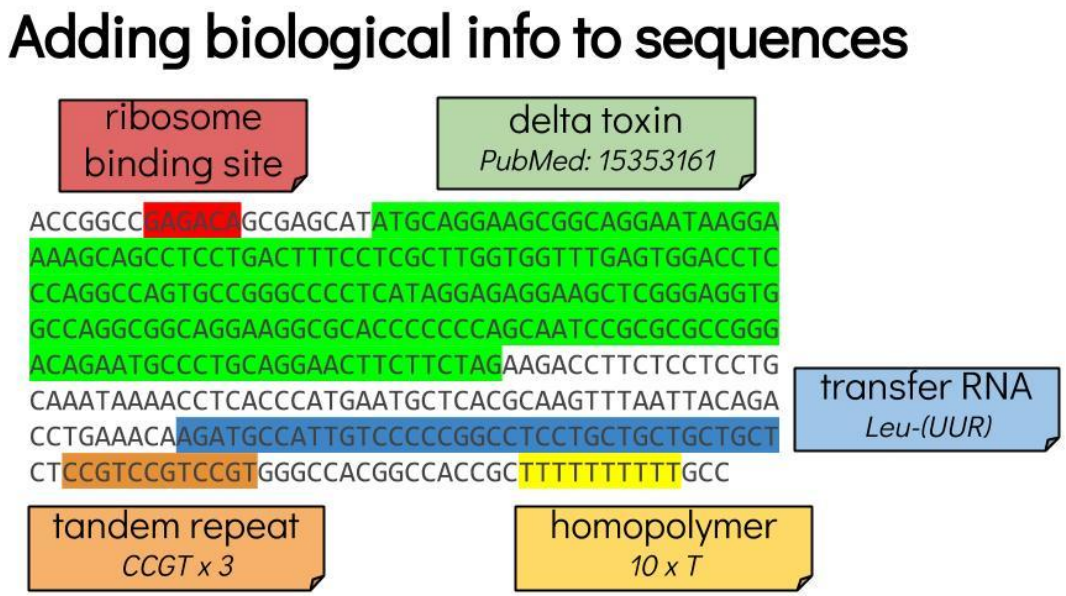


Workflow of genome analyses



Genome annotation

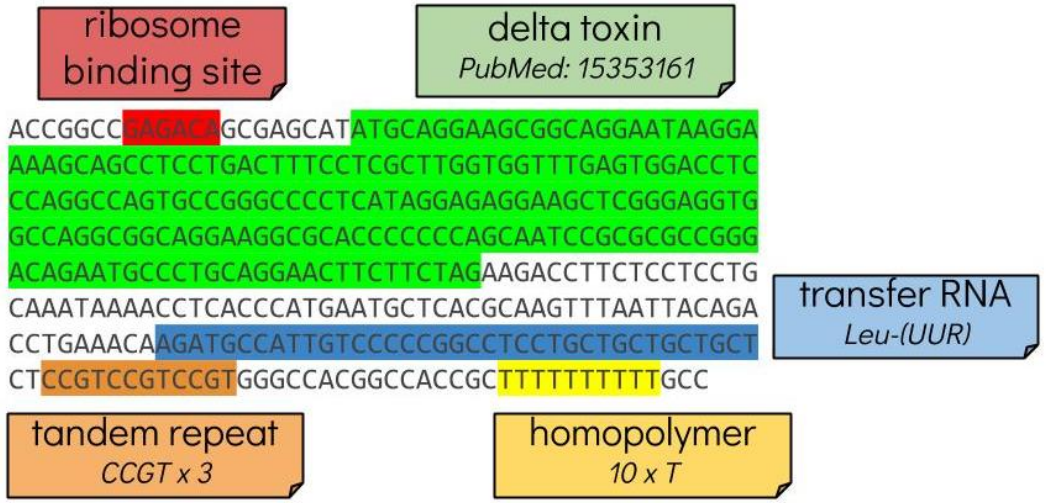
- Process of **identifying and labeling genetic elements** on a genome sequence
- **Genetic elements**: coding regions, tRNAs, rRNAs, non-coding RNAs, signal peptides, repeats, …



Torsten Seemann, https://www.slideshare.net/torstenseemann/prokka-rapid-bacterial-genome-annotation-abphm-2013

Genome annotation

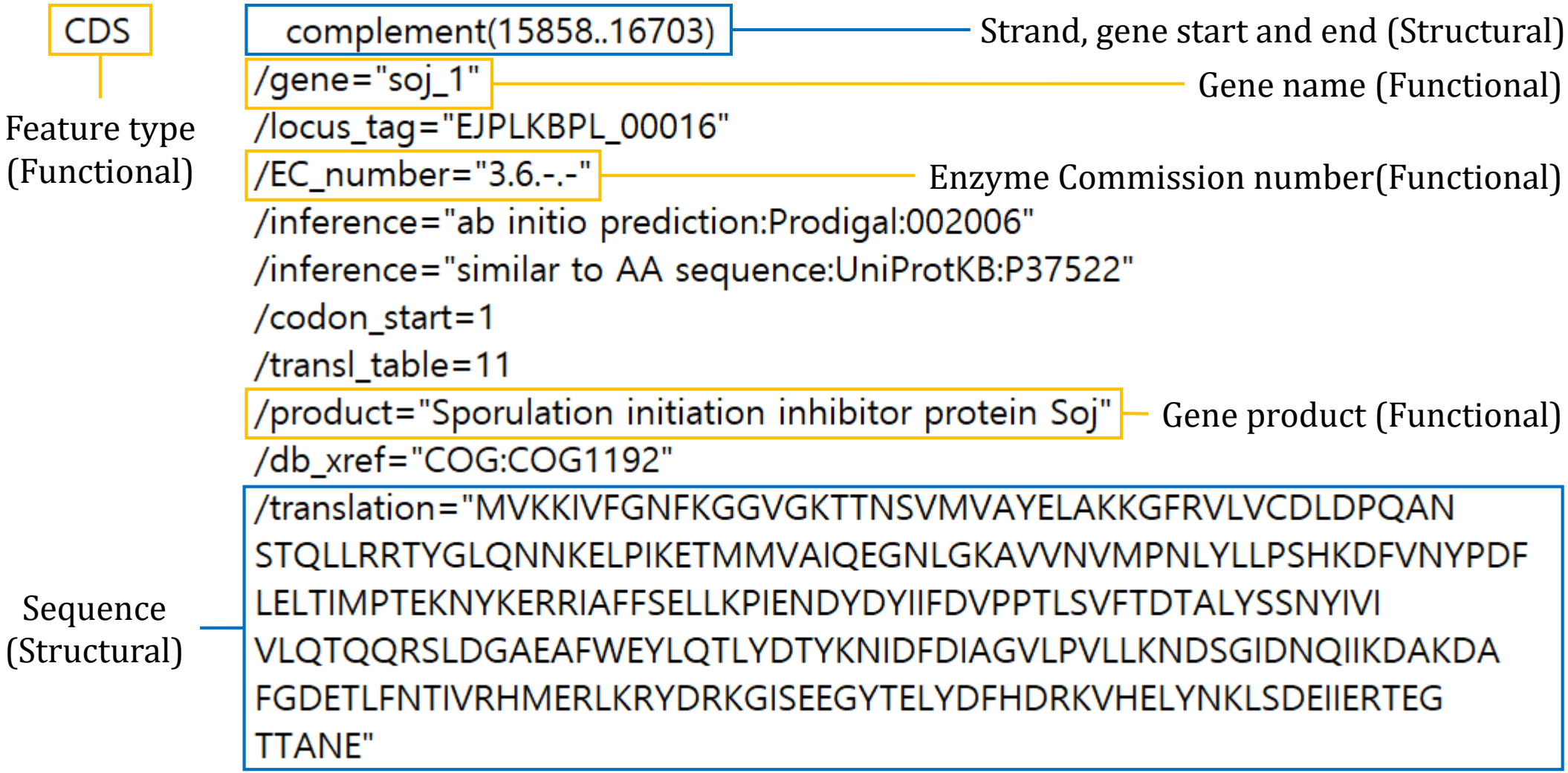
- **Structural annotation**: identifying **coordinates** of genetic elements
- **Functional annotation**: associating **functional data** with the elements



Genome annotation

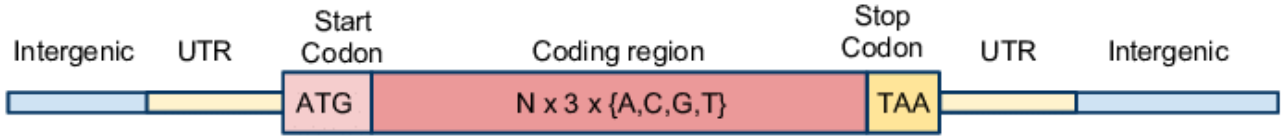
- **Coordinates**: which sequence is on which location and strand
- **Feature types**
  - Protein coding genes
    - Coding sequence, signal peptide
  - Non-coding genes
    - Transfer RNA (tRNA), ribosomal RNA (rRNA), non-coding RNA (ncRNA)
  - Others
    - Repeats
- **Attributes:** protein products, enzyme codes, subcellular location, **...**

Genome annotation example



Genome annotation strategies

- **Ab initio gene prediction**
  - Model based method using only the sequence information
  - Use patterns of promoter sequences, open reading frames, stop codons, …



- **Homology based gene prediction**
  - Search for sequences that are similar to extrinsic evidence
  - Find biologically similar sequences from databases

```
Query   1       ATAAAAAAGGAAATGTTGGAAAAGAAGAGTTTATTTATAGGGTTAATAATGTTTACTATA   60
                ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct   408534  ATAAAAAAGGAAATGTTGGAAAAGAAGAGTTTATTTATAGGGTTAATAATGTTTACTATA   408593
```

— Sequence from my assembly

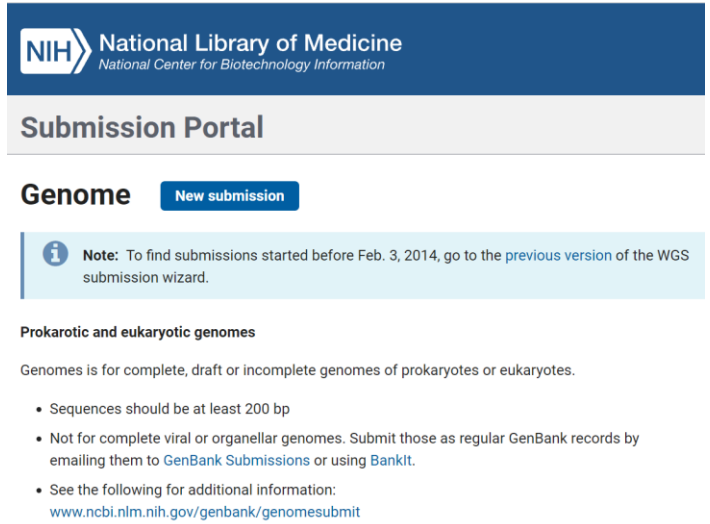
— Known sequence of a feature

Görnitz, Nico, et al. "Hierarchical Multitask Structured Output Learning for Large-scale Sequence Segmentation." NIPS. 2011.

Annotation tools

	NCBI*	RAST	xBASE2	Prokka
Availability	Web server	Web server	Web server	Stand anlone
Time	In days	Under a day	Few hours	~10 minutes

- Limitations of online annotation servers
  - Not useful where throughput or privacy is critical
  - Difficult to iterate and integrate into pipeline



*NCBI Prokaryotic Genomes Automatic Annotation Pipeline

https://submit.ncbi.nlm.nih.gov/subs/genome/

What Prokka does

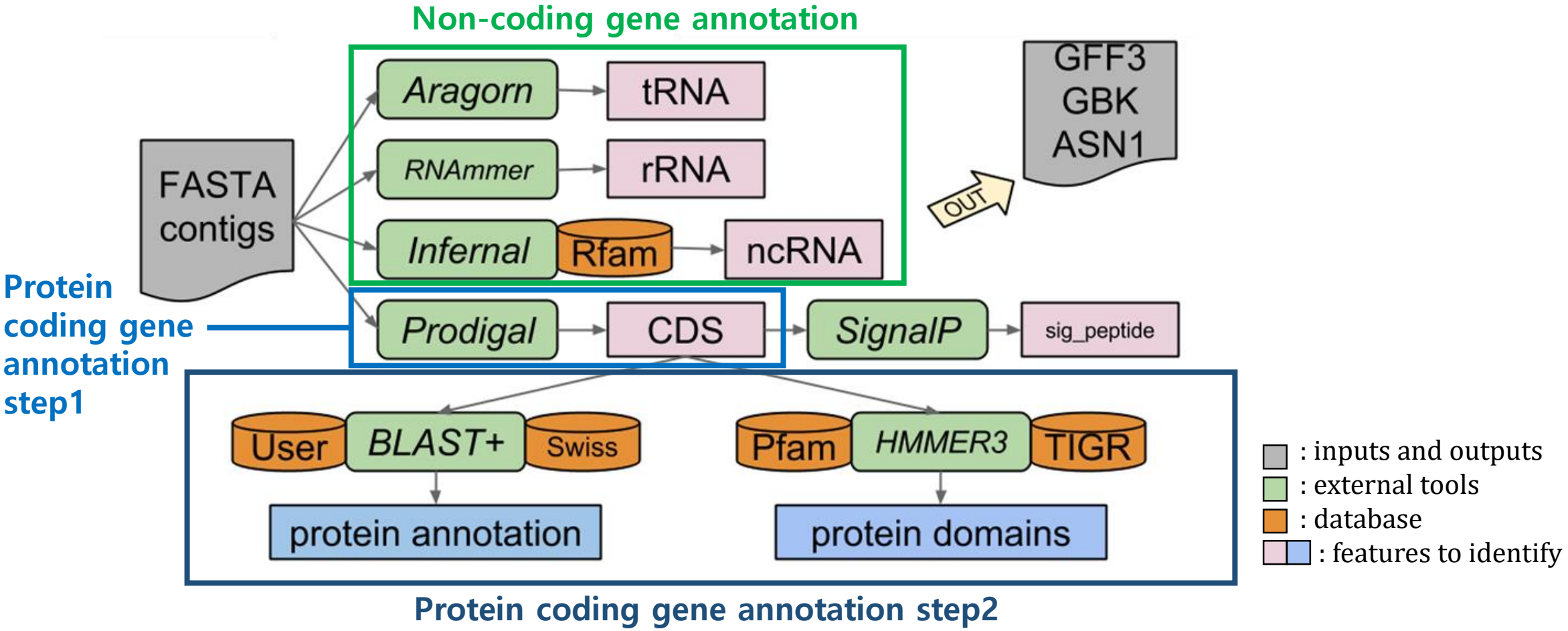
- Coordinates a suite of **existing software tools**
  - Use both **ab initio and homology based** gene prediction
- **Rich and reliable annotation** of genomic bacterial sequences
  - Can annotate protein coding genes and non-coding genes at once
- Can be installed on any Unix system and exploit multiple processing cores
- A typical bacterial genome can be annotated in 10 min on a quad core desktop computer - **Fast**

Pipeline

- Relies on external feature prediction tools

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

Pipeline

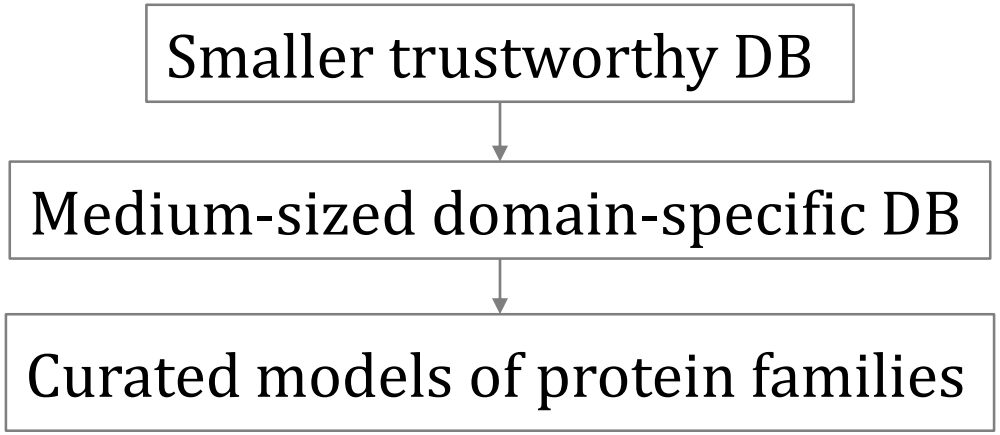


Torsten Seemann, https://www.slideshare.net/torstenseemann/prokka-rapid-bacterial-genome-annotation-abphm-2013

Protein coding gene annotation

1. **Prodigal** identifies the **coordinates of candidate genes**
2. Compares the sequence with a **database of known sequences**

- Prokka uses databases in a **hierarchical** manner



Databases for protein coding genes

1) Optional user-provided set of annotated proteins
2) All bacterial proteins in UniProt that have real protein or transcript evidence and are not fragmented (~16,000 proteins)
3) All proteins from finished bacterial genomes in RefSeq for a specified genus

BLAST+

4) Hidden Markov model profile databases (Pfam and TIGRFAMs) — **HMMER3**
5) No matches → 'hypothetical protein'

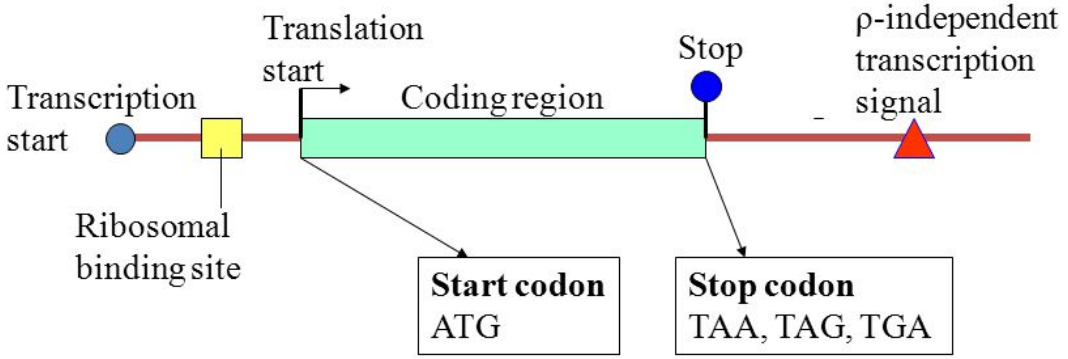
RefSeq

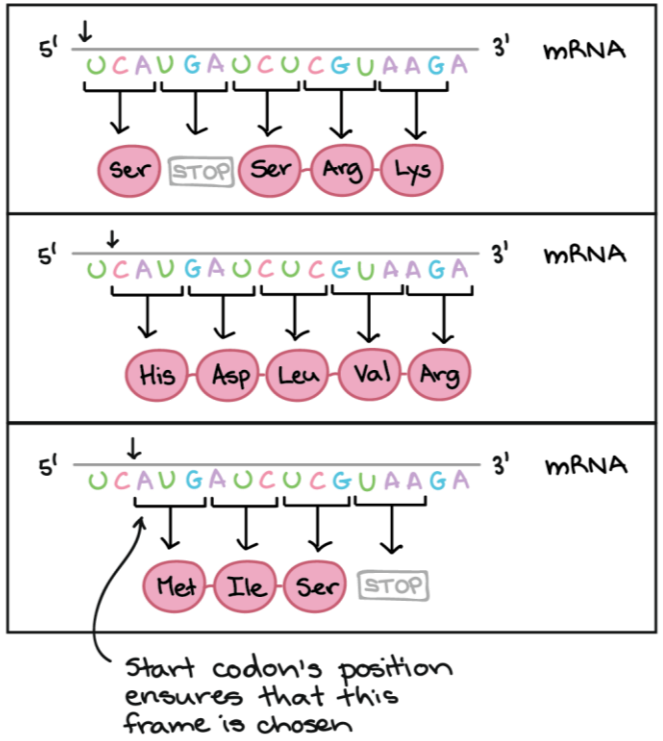
Pfam

uniprot.org; ensemble.info; pfam.xfam.org;

Prodigal – For protein coding gene annotation

1. Read in the sequence
2. Locate all starts and stops in the genome
3. Scan all open reading frames and record numbers of G's and C's in each codon position
4. Build a frame bias model based on ORF length and G/C codon position within each ORF
5. Record the highest scoring start nodes in each frame that overlap a stop codon by <= 60 bp
6. Do the first pass dynamic programming, connecting nodes based on frame bias scores
7. Create a hexamer background of all 6-mers in the entire sequence
8. FOR each gene model in the dynamic programming output:
    1. Gather all hexamer statistics
9. Create log table of hexamer coding scores
10. FOR each gene model in the dynamic programming output:
    1. Calculate a coding score based on hexamer statistics
    2. Penalize the score if there is a higher scoring start upstream in the same ORF
    3. IF the gene is very long but has a negative score, THEN give it a barely positive score

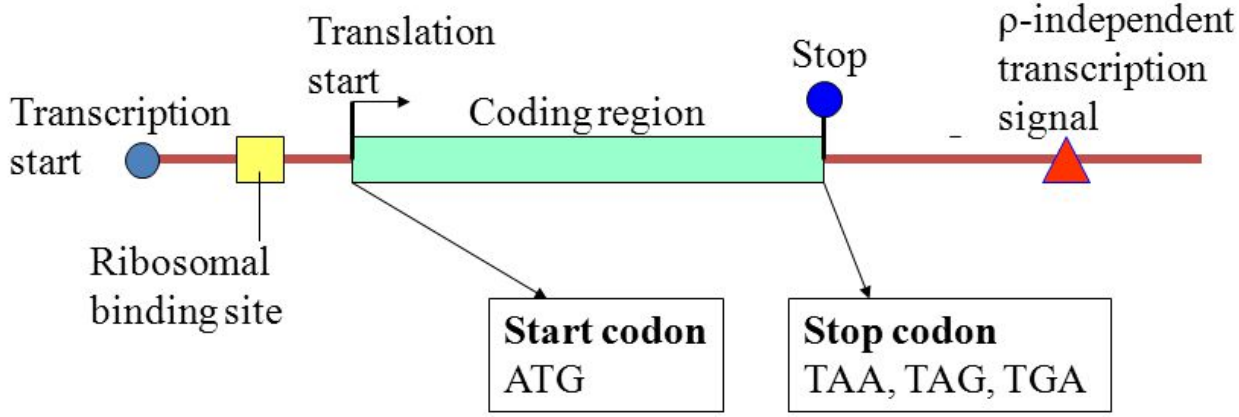




Hyatt, D., Chen, GL., LoCascio, P.F. et al. Prodigal: prokaryotic gene recognition and translation initiation site identification. BMC Bioinformatics 11, 119 (2010). 15
https://slideplayer.com/slide/5750865/; www.khanacademy.org/science/ap-biology/gene-expression-and-regulation/translation/a/the-genetic-code-discovery-and-properties

Prodigal – For protein coding gene annotation

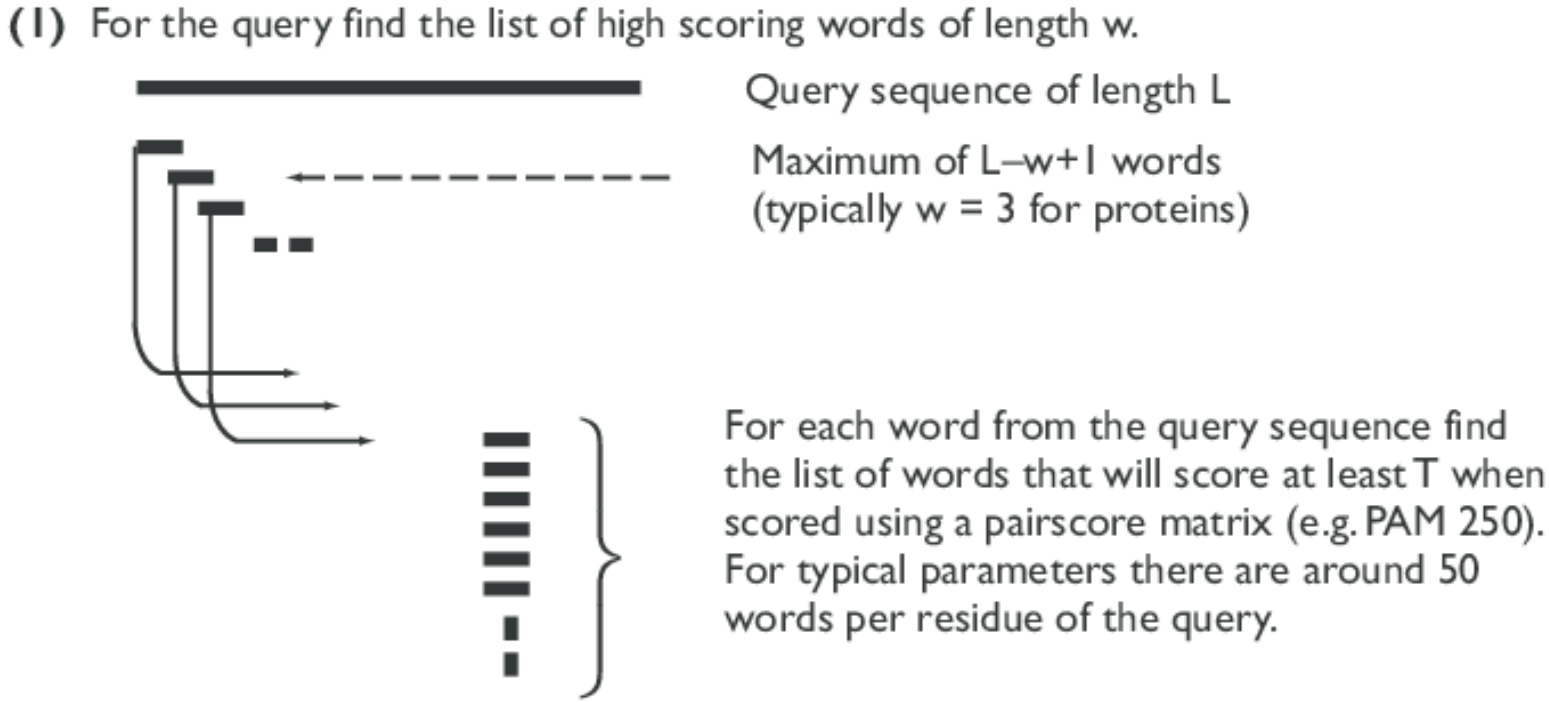
```
11. FOR 10 iterations
      1.  Build a ribosomal binding site and ATG/GTG/TTG background for all nodes
      2.  FOR each gene with a score of > 35.0:
            1.  Gather its Shine-Dalgarno RBS motif data and ATG/GTG/TTG data
      3.  Modify RBS and ATG/GTG/TTG weights by the observations
12. IF organism is not determined to use Shine-Dalgarno THEN run the non-SD finder
13. FOR each gene model:
      1.  Assign a final score of start score + coding score
      2.  Penalize the final score of genes < 250bp
14. Do the second pass dynamic programming, connecting nodes based on hexamer coding
15. FOR each gene model in the final dynamic programming:
      1.  Eliminate negative scoring models
      2.  Resolve very close start pairs (<= 15 bp from each other)
16. Print final output
```



Hyatt, D., Chen, GL., LoCascio, P.F. et al. Prodigal: prokaryotic gene recognition and translation initiation site identification. BMC Bioinformatics 11, 119 (2010). 16
https://slideplayer.com/slide/5750865

BLAST – For protein coding gene annotation

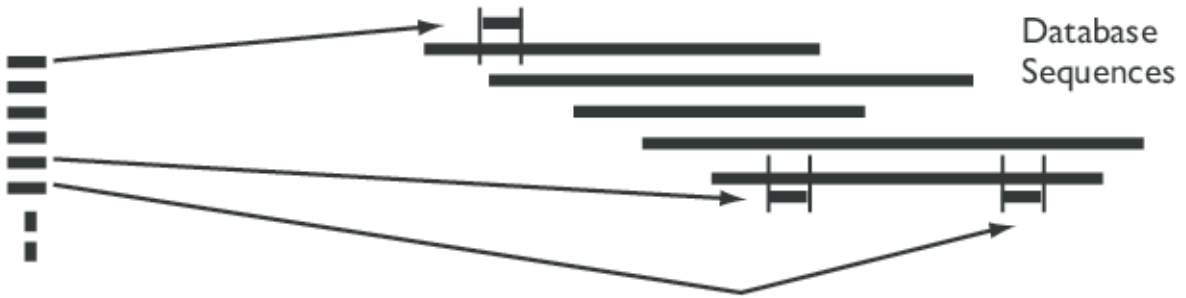
- Compares query and database sequences to find similar sequences



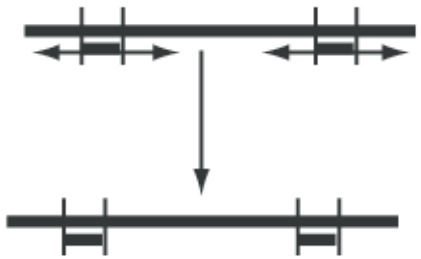
Sansom, Clare. (2000). Database searching with DNA and protein sequences: An introduction. Briefings in bioinformatics. 1. 22-32. 10.1093/bib/1.1.22.

BLAST – For protein coding gene annotation

(2) Compare the word list to the database and identify exact matches.



(3) For each word match, extend alignment in both directions to find alignments that score greater than score threshold S.



Sansom, Clare. (2000). Database searching with DNA and protein sequences: An introduction. Briefings in bioinformatics. 1. 22-32. 10.1093/bib/1.1.22.

BLAST – For protein coding gene annotation

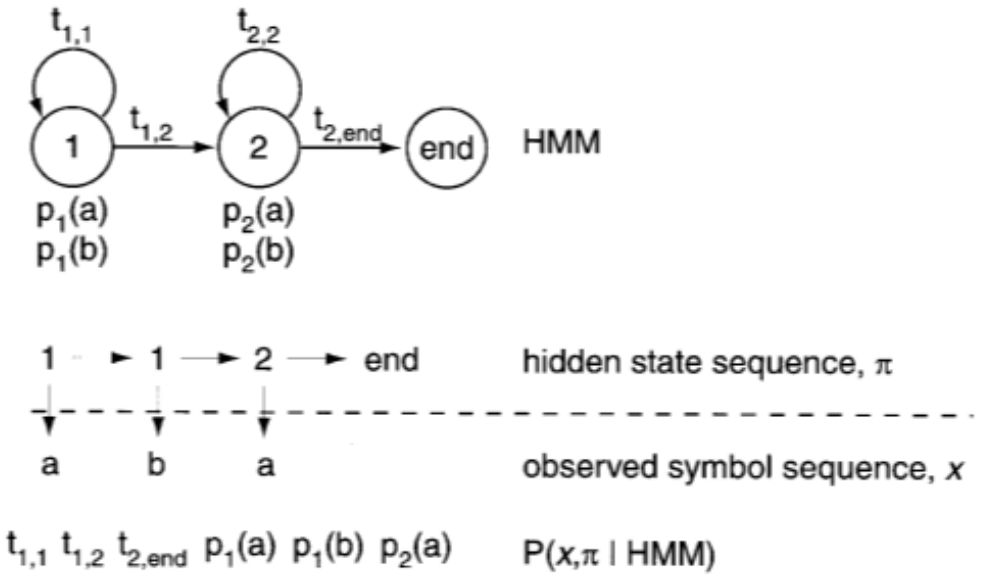
- Query: Coding sequence identified from Prodigal
- Subject: Gene sequence from database
- Database is used hierarchically (as the order in slide 13)
  - If there is a match from DB 1) then 2) is not searched
  - The gene from DB 1) is annotated
- Prokka identifies genes of e-value < 10^{-6}

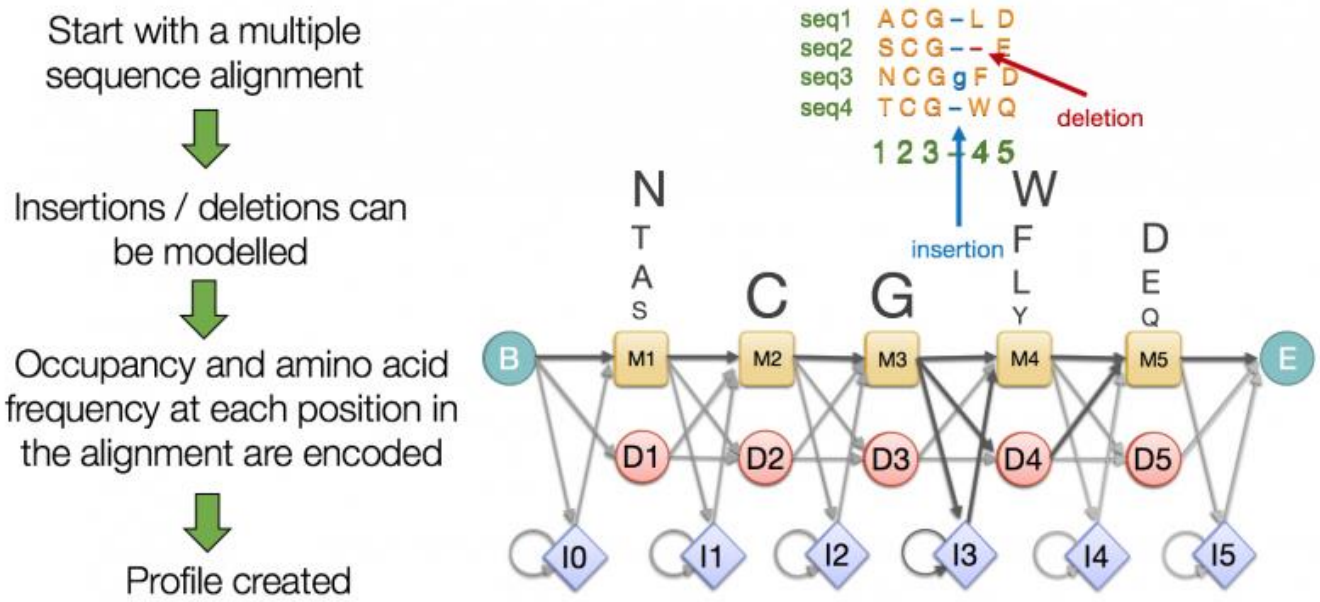
Score	Expect	Identities	Gaps	Strand
542 bits(293)	9e-150	296/297(99%)	1/297(0%)	Plus/Plus

```
Query  1      GTGGTCTGGCAGGCAGATTATCCTGACCCTATGAGTTTTTTAGGTAACTTTGAAAGTAAC  60
              ||||  ||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  54826  GTGG-CTGGCAGGCAGATTATCCTGACCCTATGAGTTTTTTAGGTAACTTTGAAAGTAAC  54884

Query  61     AGTGTGTTGAATTTTGGAGGTTATAGCAATACTAAATATGATGAATACTTAAAAGATACC  120
              ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  54885  AGTGTGTTGAATTTTGGAGGTTATAGCAATACTAAATATGATGAATACTTAAAAGATACC  54944
```

HMMER – For protein coding gene annotation



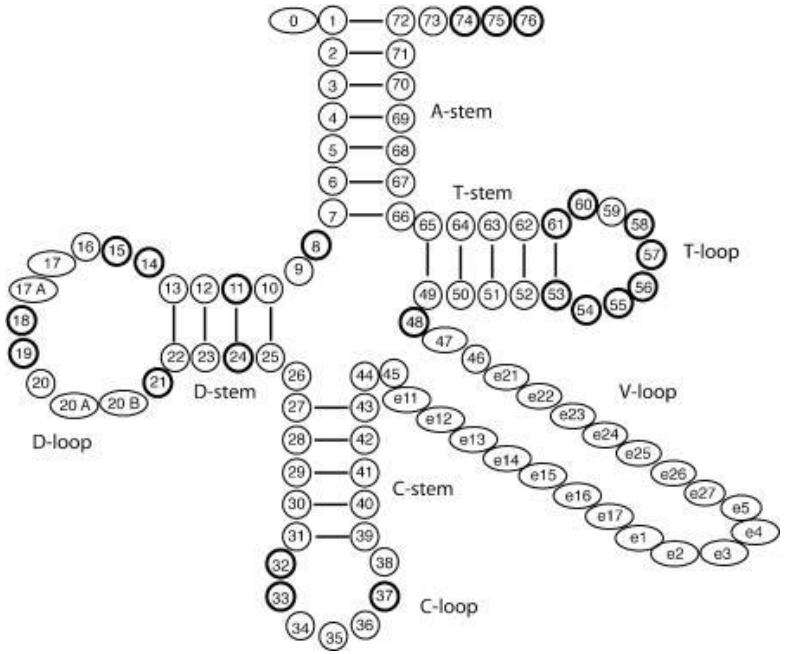


- Toy example of HMM (hidden Markov model)
- Profile HMM modelling a multiple sequence alignment

Profile Hidden Markov Models. S. R. Eddy. Bioinformatics, 14:755-763, 1998.; https://www.ebi.ac.uk/training/online/

tRNA gene annotation

- **tRNA gene: Aragorn**
  - **Heuristic algorithms** to predict tRNA secondary structure
  - Based on **homology** with recognized tRNA consensus sequences and ability to form a base-paired cloverleaf
  - Heuristic algorithm: adapting the approach to a problem based on previous solutions to similar problems



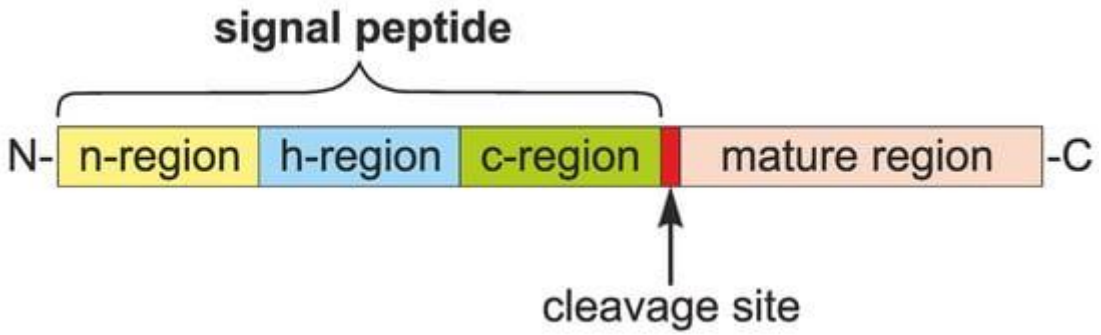
Laslett D, Canback B. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. Nucleic Acids Res. 2004;32(1):11-16

rRNA and ncRNA gene annotation

- **rRNA gene: RNAmmer**
  - **HMMs** trained on data from the 5S ribosomal RNA database and the European ribosomal RNA database project
- **ncRNA: Infernal**
  - Builds consensus RNA secondary structure profiles called **covariance models** (CMs), and uses them to search nucleic acid sequence databases for homologous RNAs
  - A CM is **like a sequence profile**, but it scores a combination of sequence consensus and **RNA secondary structure consensus**

Signal peptide annotation

- **Signal peptide: SignalP**
  - Short amino acid sequences in the amino terminus of many newly synthesized proteins that target proteins into, or across, membranes
  - **Neural network–based** method



Burdukiewicz M, et al. Prediction of Signal Peptides in Proteins from Malaria Parasites. International Journal of Molecular Sciences. 2018; 19(12):3709.

Outputs

Suffix	Description of file contents
.fna	FASTA file of original input contigs (nucleotide)
.faa	FASTA file of translated coding genes (protein)
.ffn	FASTA file of all genomic features (nucleotide)
.fsa	Contig sequences for submission (nucleotide)
.tbl	Feature table for submission
.sqn	Sequin editable file for submission
.gbk	Genbank file containing sequences and annotations
.gff	GFF v3 file containing sequences and annotations
.log	Log file of Prokka processing output
.txt	Annotation summary statistics

.fna, .ffn, .fas

```
>contig_1_pilon
GTGCTAAATAAGGAAAATCAAGAAAACTATTACT
TCTTTTGAAGAGTTAAGTTTAGAAGAAATGGAAG
GCTGAGACAACTCCAGCATGTTTTACCATAGGCTT
```

.faa

```
>EJPLKBPL_00001 hypothetical protein
MEDNLINVLSINERCFLLKQSGNEKYDIKNLQAWKER
GLGITPIENFPDKEVAIQYIKDQSWYIFFESILDSYNDSE
LFLLDLNSELNICTKEFIINLLENLTQELIHLTSKTLVLDL
```

.tbl

```
>Feature contig_1_pilon
253         3234        gene
                        locus_tag  EJPLKBPL_00001
253         3234        CDS
                        inference  ab initio prediction:Prodigal:002006
                        locus_tag  EJPLKBPL_00001
                        product    hypothetical protein
```

.sqn

```
Seq-entry ::= set {
  class genbank ,
  seq-set {
    set {
      class nuc-prot ,
      descr {
        source {
          org {
            taxname "Genus species" ,
```

.gbk

```
LOCUS       contig_1_pilon         54297 bp    DNA     linear       10- 8월-2021
DEFINITION  Genus species strain strain.
ACCESSION
VERSION
KEYWORDS    .
SOURCE      Genus species
  ORGANISM  Genus species
            Unclassified.
COMMENT     Annotated using prokka 1.14.6 from
            https://github.com/tseemann/prokka.
FEATURES             Location/Qualifiers
     source          1..54297
                     /organism="Genus species"
                     /mol_type="genomic DNA"
                     /strain="strain"
     gene            253..3234
                     /locus_tag="EJPLKBPL_00001"
     CDS             253..3234
                     /locus_tag="EJPLKBPL_00001"
                     /inference="ab initio prediction:Prodigal:002006"
                     /codon_start=1
                     /transl_table=11
                     /product="hypothetical protein"
                     /translation="MEDNLINVLSINERCFLLKQSGNEKYDIKNLQAWKERKSVLKQD
```

.log

```
[13:49:22] This is prokka 1.14.6
[13:49:22] Written by Torsten Seemann <torsten.seemann@gmail.com>
[13:49:22] Homepage is https://github.com/tseemann/prokka
[13:49:22] Local time is Tue Aug 10 13:49:22 2021
```

Outputs - .gff

```
contig_1_pilon  prokka  gene     253 3234    .   +   .   ID=EJPLKBPL_00001_gene;locus_tag=EJPLKBPL_00001
contig_1_pilon  Prodigal:002006 CDS 253 3234    .   +   0   ID=EJPLKBPL_00001;Parent=EJPLKBPL_00001_gene;in
contig_1_pilon  prokka  gene     3246 5390    .   +   .   ID=EJPLKBPL_00002_gene;Name=lagD_1;gene=lagD_1;
```

1. Sequence name
2. Source of annotation
3. Type of feature
4. Start coordinate (1-based, inclusive)
5. End coordinate (1-based, inclusive)
6. Score
7. Strand (compared to reference)
8. Frame (0, 1, 2)
9. Attributes (separated by ;)

Outputs - .txt

- Summary of annotated feature types

organism: Genus species strain
contigs: 4
bases: 3089071
CDS: 3011
gene: 3090
rRNA: 12
repeat_region: 1
tRNA: 66
tmRNA: 1

Results

- Comparison of annotation of *E. Coli K-12* accession U00096.2

Feature	Reference	Prokka	RAST	xBase2
Total CDS	4321	**4305**	4512	4444
Matching start	–	**3828**	3571	3025
Different start	–	**318**	533	1052
Missing CDS	–	**172**	214	241
Extra CDS	–	**159**	405	367
Hypothetical protein	18	276	638	**156**
With EC number	1114	1050	**1118**	0
Total tRNA	89	**88**	86	**88**
Total rRNA	22	**22**	**22**	**22**

The bold denotes the best performing tool (column) for that attribute (row). The italics are "subsets" of the "Total CDS" section.

Summary

- **Annotation** is a process of **identifying and labeling genetic elements** (protein coding genes, non-coding RNAs, …) on a genome sequence
- There are web- or email-based systems for prokaryotic genome annotation, but these are not applicable for sensitive data or integrating into computational pipelines
- **Prokka** is a **command line software tool to fully annotate a draft bacterial genome** in about 10 min on a typical desktop computer
- The input for Prokka is a genome assembly (.fasta) and the outputs are standards-compliant files such as .gff files

Thank you for listening!