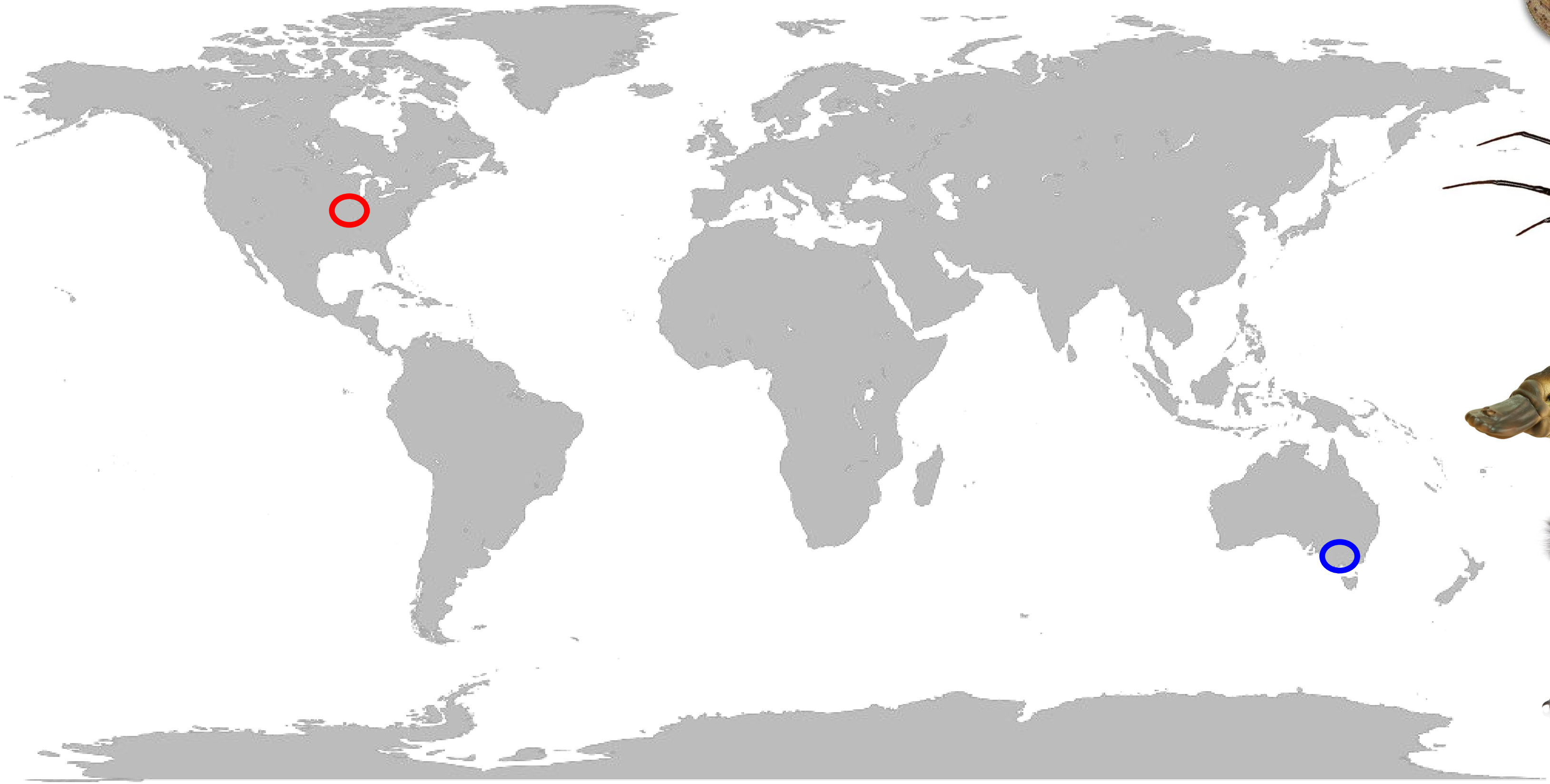
# Bacterial genome annotation

Torsten Seemann
Annette McGrath
Simon Gladman
Anna Syme

Victorian Life Sciences Computation Initiative (VLSCI)
The University of Melbourne

# About us

# Melbourne, Australia



# Bacterial genomes

# Small genome

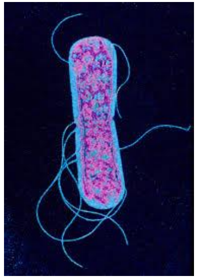
6,000,000,000

letters

30,000 genes

**Genome**

**A T G C**

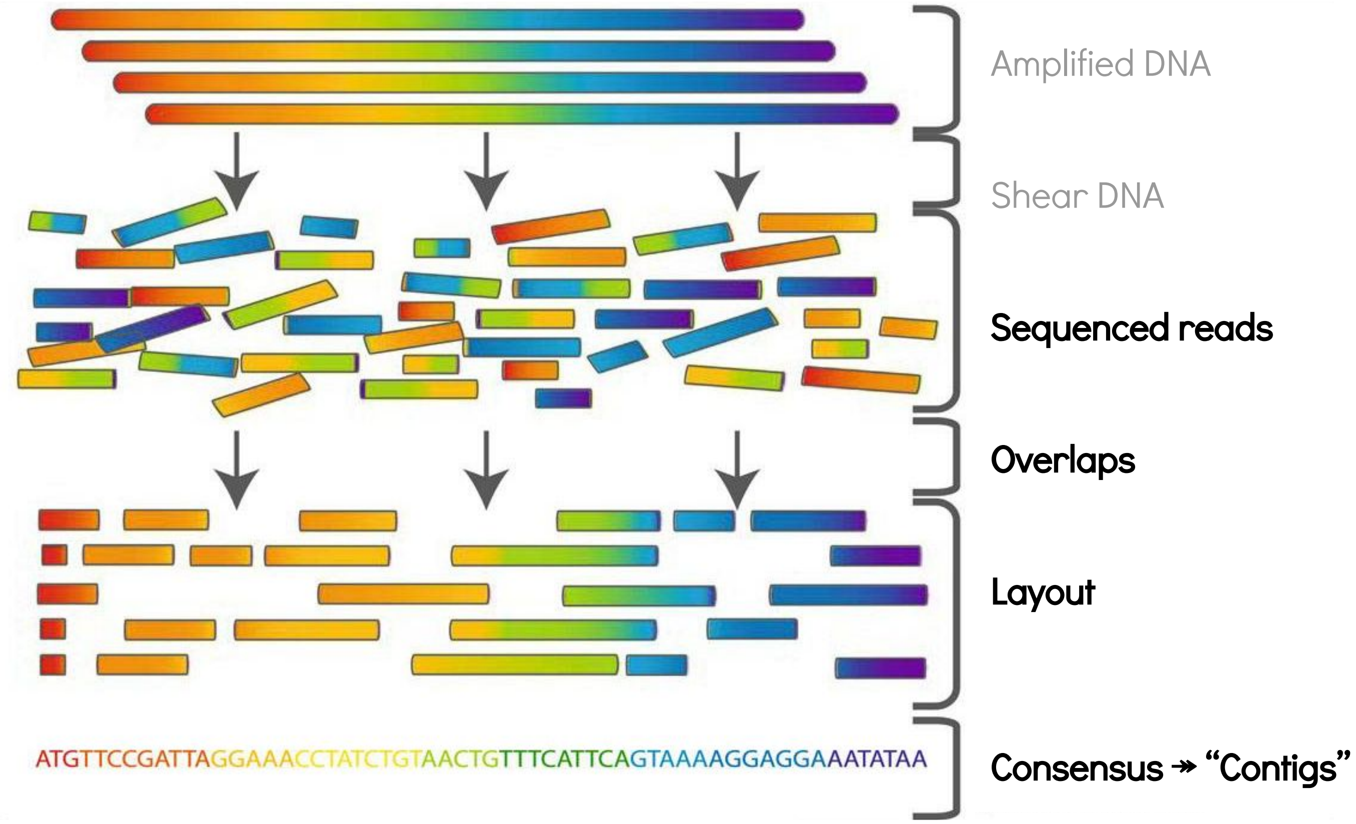


3,000,000

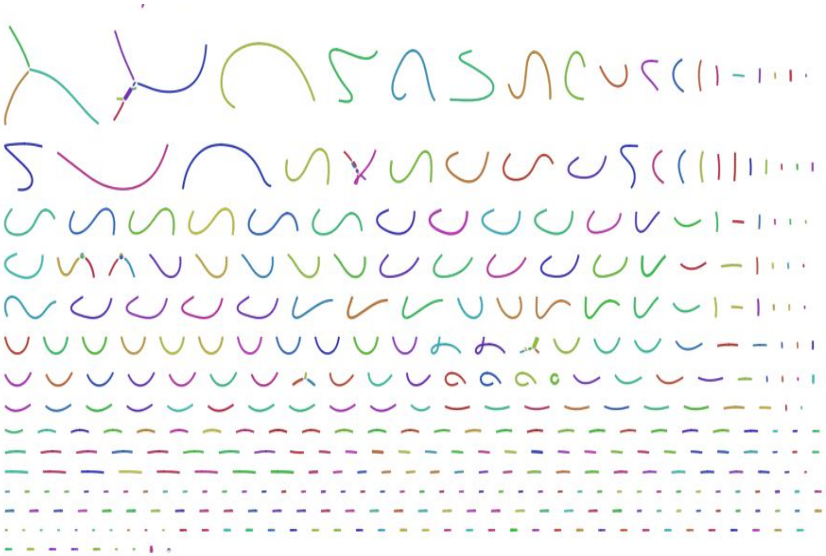
letters

3,000 genes

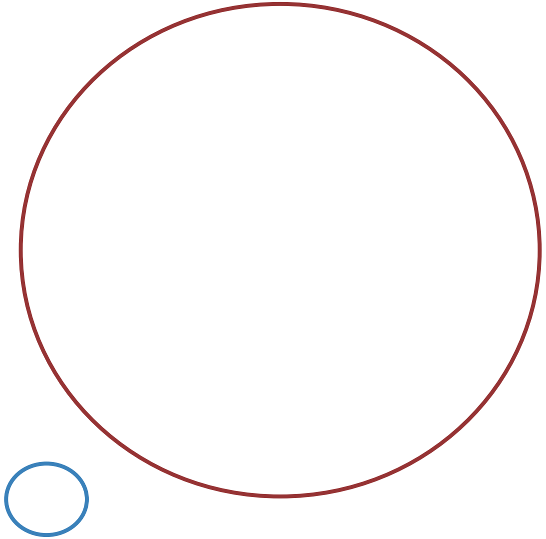
# Overlap - Layout - Consensus



# Draft vs Finished genomes



Lots of contigs



One contig per replicon

# Annotation

# Adding biological info to sequences

ribosome binding site

delta toxin
*PubMed: 15353161*

ACCGGCCGAGACAGCGAGCATATGCAGGAAGCGGCAGGAATAAGGA
AAAGCAGCCTCCTGACTTTCCTCGCTTGGTGGTTTGAGTGGACCTC
CCAGGCCAGTGCCGGGCCCCTCATAGGAGAGGAAGCTCGGGAGGTG
GCCAGGCGGCAGGAAGGCGCACCCCCCCAGCAATCCGCGCGCCGGG
ACAGAATGCCCTGCAGGAACTTCTTCTAGAAGACCTTCTCCTCCTG
CAAATAAAACCTCACCCATGAATGCTCACGCAAGTTTAATTACAGA
CCTGAAACAAGATGCCATTGTCCCCCGGCCTCCTGCTGCTGCTGCT
CTCCGTCCGTCCGTGGGCCACGGCCACCGCTTTTTTTTTTGCC

transfer RNA
*Leu-(UUR)*

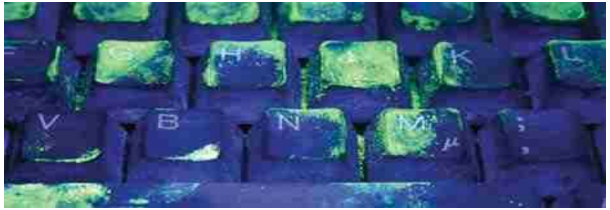
tandem repeat
*CCGT x 3*

homopolymer
*10 x T*

# What's 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*

# Bacterial feature types

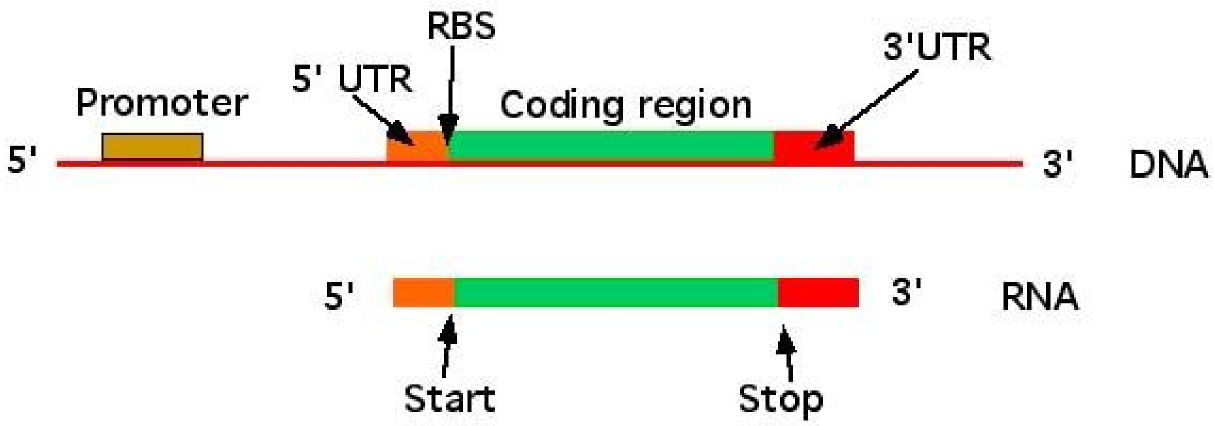


- protein coding genes
  - promoter (-10, -35)
  - ribosome binding site (RBS)
  - coding sequence (CDS)
    - signal peptide, protein domains, structure
  - terminator

- non coding genes
  - transfer RNA (tRNA)
  - ribosomal RNA (rRNA)
  - non-coding RNA (ncRNA)

- other
  - repeat patterns, operons, origin of replication, ...

# Look mum, no introns!




- 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

# Automatic annotation

# 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

# Some good existing tools

| Software | *ab initio* | align-ment | Availability | Speed |
|---|---|---|---|---|
| RAST | yes | yes | web only | 12-24 hours |
| xBASE | yes | no | web only | >4 hours |
| BG7 | no | yes | standalone | >10 hours |
| PGAAP (NCBI) | yes | yes | email / we | >1 month |

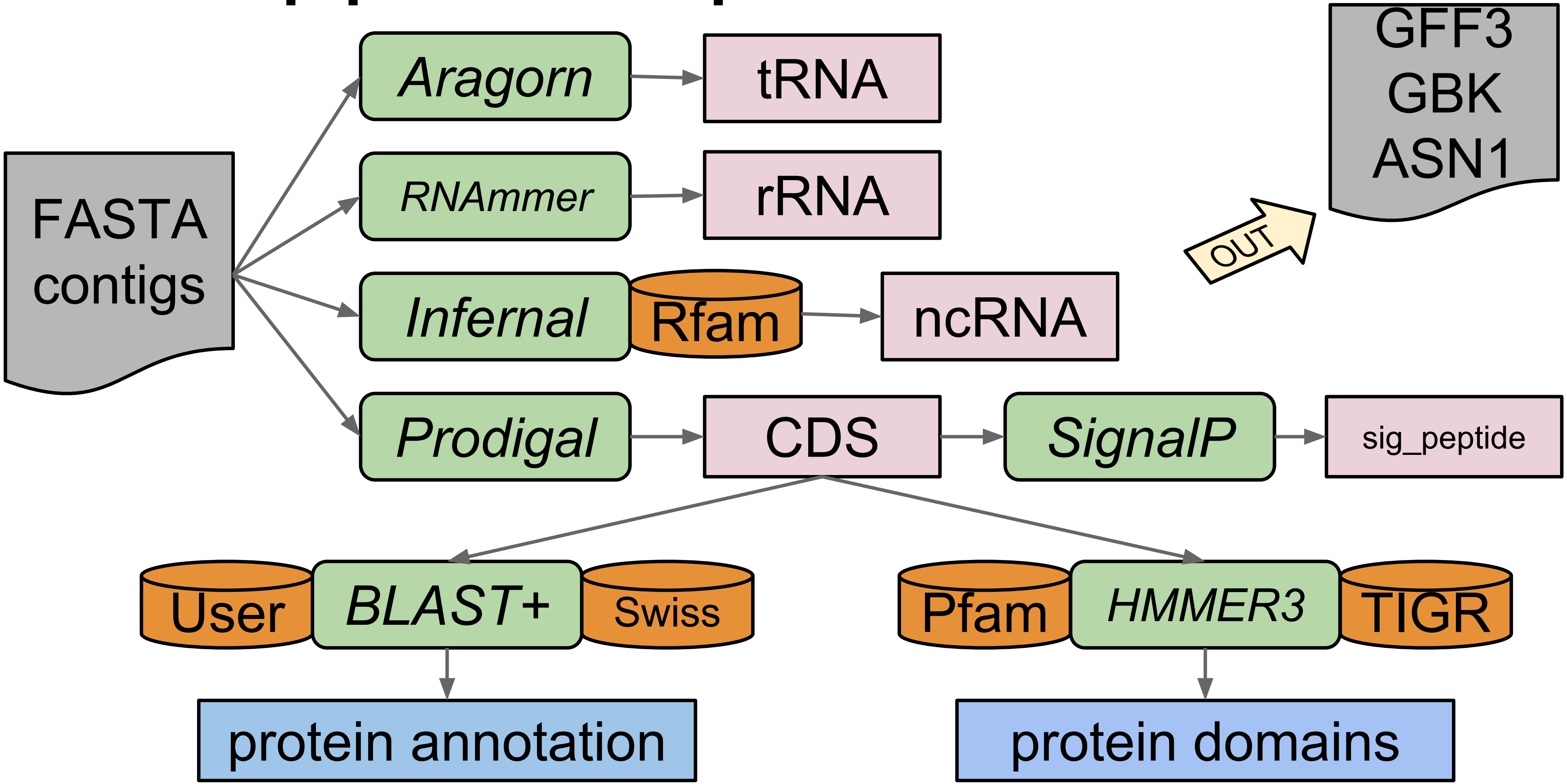
# Why another tool?

- Convenience
    - I have sequence, just tell me what's in it, please.
- Speed
    - exploit multi-core computers (aim < 15min)
- Standards compliant
    - GFF3/GBK for viewing, TBL/FSA for Genbank sub.
- Rich consistent trustworthy output
    - /product /gene /EC_number
- Provenance
    - a record of where/how/why it was annotated so

# Why "Prokka" ?

- Unique in Google
- I like the letter "k"
- Easy to type
- It sounds Aussie
- Loosely fits "Prokaryotic Annotation"
- It rhymes with "Quokka"
  - Australian cat-sized nocturnal marsupial herbivore
  - first Aussie mammal seen by Europeans - "giant rat"

# Prokka pipeline (simplified)



# What can you trust?

# 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

# Sequence databases

*I'll just BLAST against the non-redundant database.*

*-- Anonymous*

- Which one?
  - nucleotide (nt) or protein (nr)
- It's actually quite redundant
  - only eliminates exact matching sequences
- It's not picky
  - nearly anything is admitted, garbage in garbage out
- It's too big
  - searching takes too long

# Hierarchical 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 remainder

- 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

- Custom model databases
  - I took COG/PRK MSAs and made HMMs

- Existing model databases
  - Pfam, TIGRfams are well curated

- And if all else fails
  - we always have our friend "*hypothetical protein*"

# Provenance

# 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 +`

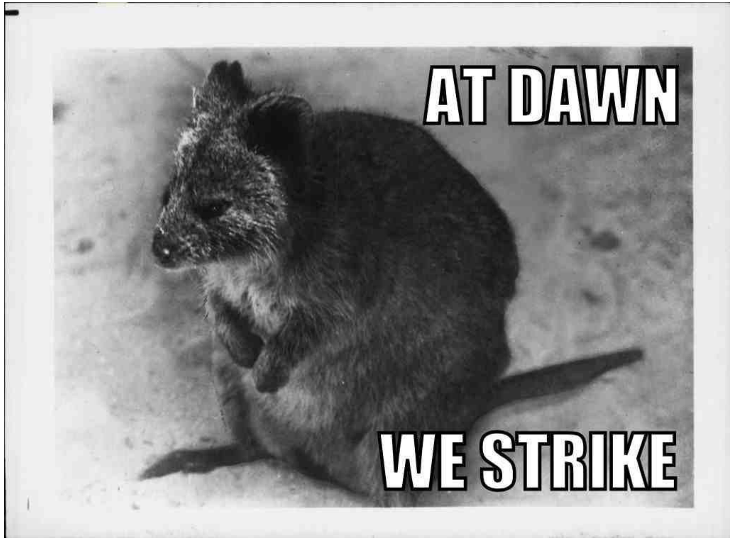
## Attributes:

```
/gene="tRNA-Leu(UUR)"

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

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

/inference="profile:Aragorn:1.2"
```

# Software quality

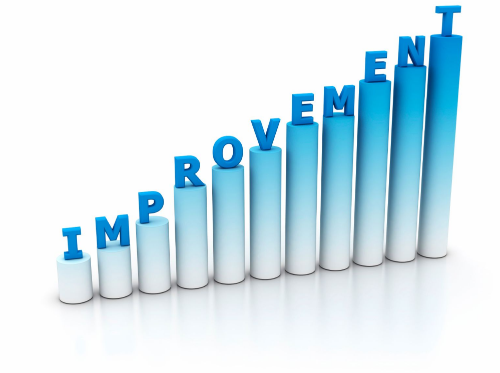
# Prokka in the wild

- Sanger Institute UK
  - Pathogen Informatics Unit
  - 50,000 draft genomes in 2 weeks (24 sec each!)
  - Now done >100,000 genomes



# Curating genomes

# Improving annotations




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

- Curation
  - Manual effort to improve annotations
  - Community curation

# Web curation demo

*WebApollo*

# The end.

Bioinformatics Advance Access published March 18, 2014

*Genome Analysis*

# Prokka: rapid prokaryotic genome annotation

Torsten Seemann[1,2,*]

[1] Victorian Bioinformatics Consortium, Monash University, Melbourne, Australia.

[2] Life Sciences Computation Centre, Victorian Life Sciences Computation Initiative, Melbourne, Australia.

Associate Editor: Prof. Alfonso Valencia

**ABSTRACT**

**Summary:** The multiplex capability and high yield of current day DNA sequencing instruments has made bacterial whole genome sequencing a routine affair. The subsequent *de novo* assembly of reads into contigs has been well addressed. The final step of annotating all relevant genomic features on those contig can be achieved slowly using existing web and email-based systems, but these are not applicable for sensitive data or integrating into computational pipelines. Here we introduce Prokka, a command line software tool to fully annotate a draft bacterial genome in about ten minutes on a typical desktop computer. It produces standards-compliant output files for further analysis or viewing in genome browsers.

**Availability and Implementation:** Prokka is implemented in Perl and is freely available under an open source GPLv2 license from http://vicbioinformatics.com/.

**Contact:** torsten.seemann@monash.edu

## 2 DESCRIPTION

### 2.1 Input

Prokka expects pre-assembled genomic DNA sequences in FASTA format. Finished sequences without gaps are the ideal input, but it is expected that the typical input will be a set of scaffold sequences produced by *de novo* assembly software. This sequence file is the only mandatory parameter to the software.

### 2.2 Annotation

Prokka relies on external feature prediction tools to identify the coordinates of genomic features within contigs. These tools are listed in Table 1, and all of them, except for Prodigal, provide co-