

Genome Annotation

Лекция 7

Валерий Павлов



Цель занятия

Аннотация генома (prokaryotic): структура → функция → доказательство

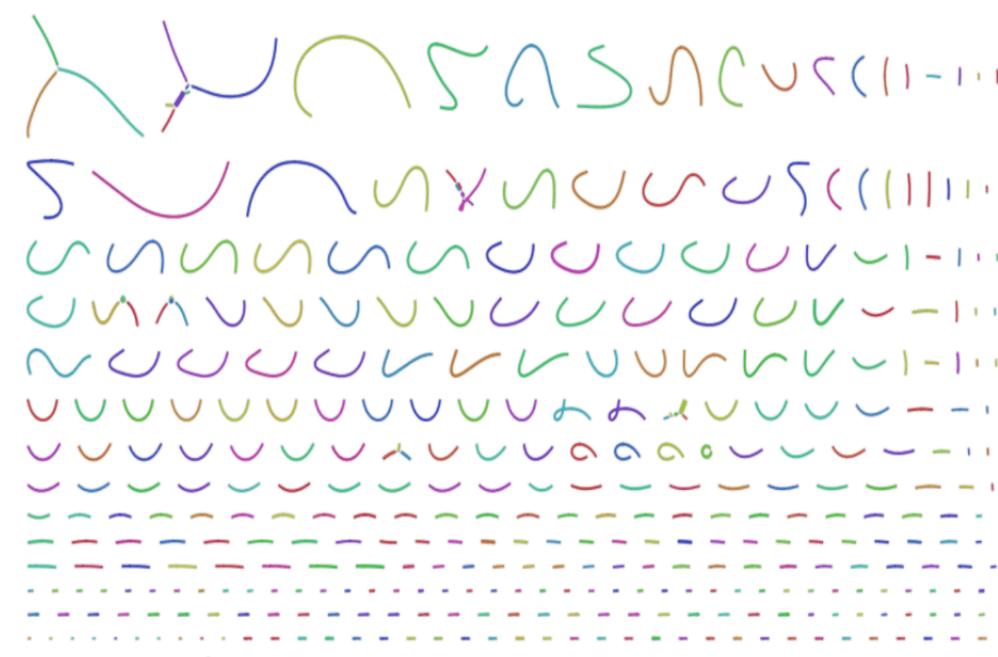
- Где гены?
- Что делают?
- Как проверить?

Prerequisites

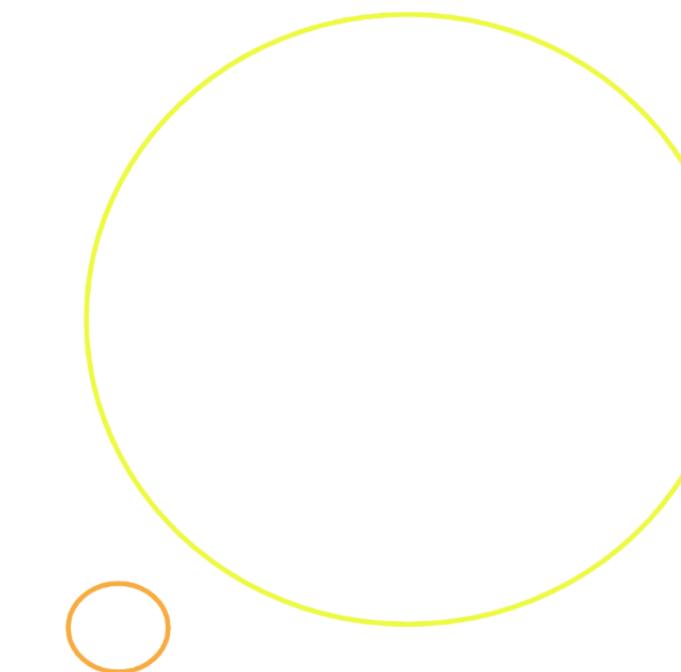
- Ген / ORF / CDS - разные у бактерий и прокариот
- Риды / Контиги / FASTQ / FASTA / QC
- Выравнивание, геном, сборка, NP50
- Гомологи / паралоги

Аннотация

Reads → QC → Assembly → Annotation → downstream (variants / pathways / comparative)



May be in lots of contigs



Or, one contig per replicon

Adding biological info to sequences

ribosome
binding site

ACCGGCCGAGAACAGCGAGCATATGCAGGAAGCGGCAGGAATAAGGA
AAAGCAGCCTCCTGACTTTCTCGCTTGGTGGTTGAGTGGACCTC
CCAGGCCAGTGCCGGGCCCTCATAGGAGAGGAAGCTCGGGAGGTG
GCCAGGCCAGGAAGCGCACCCCCCAGCAATCCGCGCGCCGGG
ACAGAATGCCCTGCAGGAACCTCTTAGAAGACCTCTCCTCTG
CAAATAAAACCTCACCCATGAATGCTCACGCAAGTTAATTACAGA
CCTGAAACAAGATGCCATTGTCCCCCGGCCCTGCTGCTGCTGCT
CTCCGTCCGTCCGTGGGCCACGGCCACCGCTTTTTTTGCC

delta toxin
PubMed: 15353161

transfer RNA
Leu-(UUR)

tandem repeat
CCGT x 3

homopolymer
10 x T

Аннотация

Reads → QC → Assembly → Annotation → downstream (variants / pathways / comparative)

Annotation

def: a note added by way of explanation or commentary

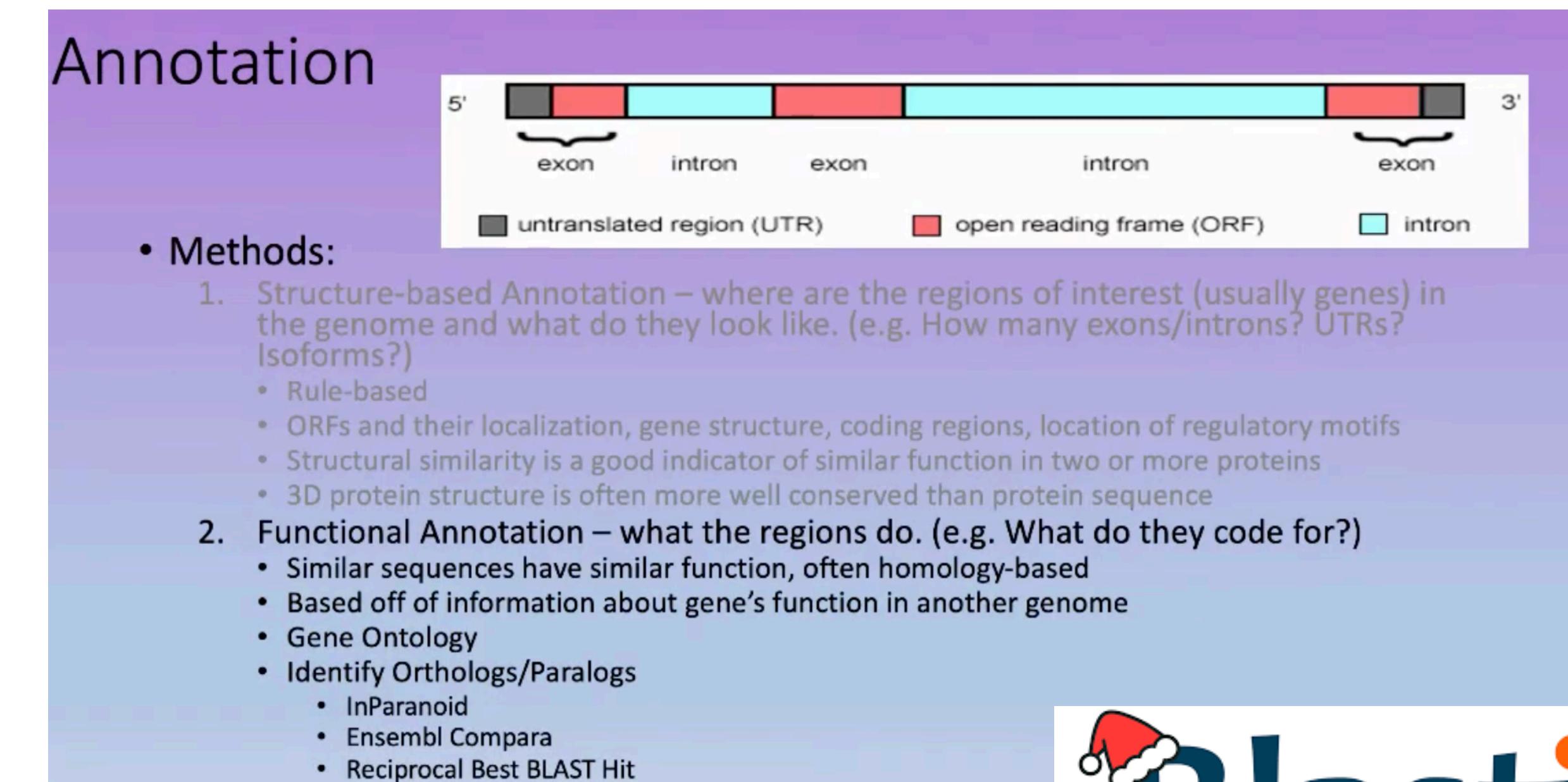
Genome annotation: Assigning a (possible) role (**functional?**) to a string of nucleotides

Genome annotation consists of three main steps:

1. identifying portions of the genome that do not code for proteins
2. identifying elements on the genome (prediction)
3. attaching biological information to these elements.

Аннотация - структурная и функциональная

- Структурная: CDS/tRNA/rRNA
- Функциональная: продукт / семейство / домены
- ФУНКЦИЯ - ГИПОТЕЗА



Evidence ladder (лестница доказательств)

ORF → coding potential/model → homology hits → domains → curated rules/taxonomy → human sanity

Annotation



The diagram illustrates a gene structure with a 5' end on the left and a 3' end on the right. It consists of alternating red boxes labeled 'exon' and light blue boxes labeled 'intron'. Brackets below the exons indicate they are 'untranslated region (UTR)'. A legend at the bottom defines the colors: dark grey for UTR, red for ORF, and light blue for intron.

- Methods:
 1. Structure-based Annotation – where are the regions of interest (usually genes) in the genome and what do they look like. (e.g. How many exons/introns? UTRs? Isoforms?)
 - Rule-based
 - ORFs and their localization, gene structure, coding regions, location of regulatory motifs
 - Structural similarity is a good indicator of similar function in two or more proteins
 - 3D protein structure is often more well conserved than protein sequence
 2. Functional Annotation – what the regions do. (e.g. What do they code for?)
 - Similar sequences have similar function, often homology-based
 - Based off of information about gene's function in another genome
 - Gene Ontology
 - Identify Orthologs/Paralogs
 - InParanoid
 - Ensembl Compara
 - Reciprocal Best BLAST Hit

Как понять, что это за ген?

Найти похожий из уже описанных

Adding biological info to sequences

ribosome
binding site

delta toxin
PubMed: 15353161

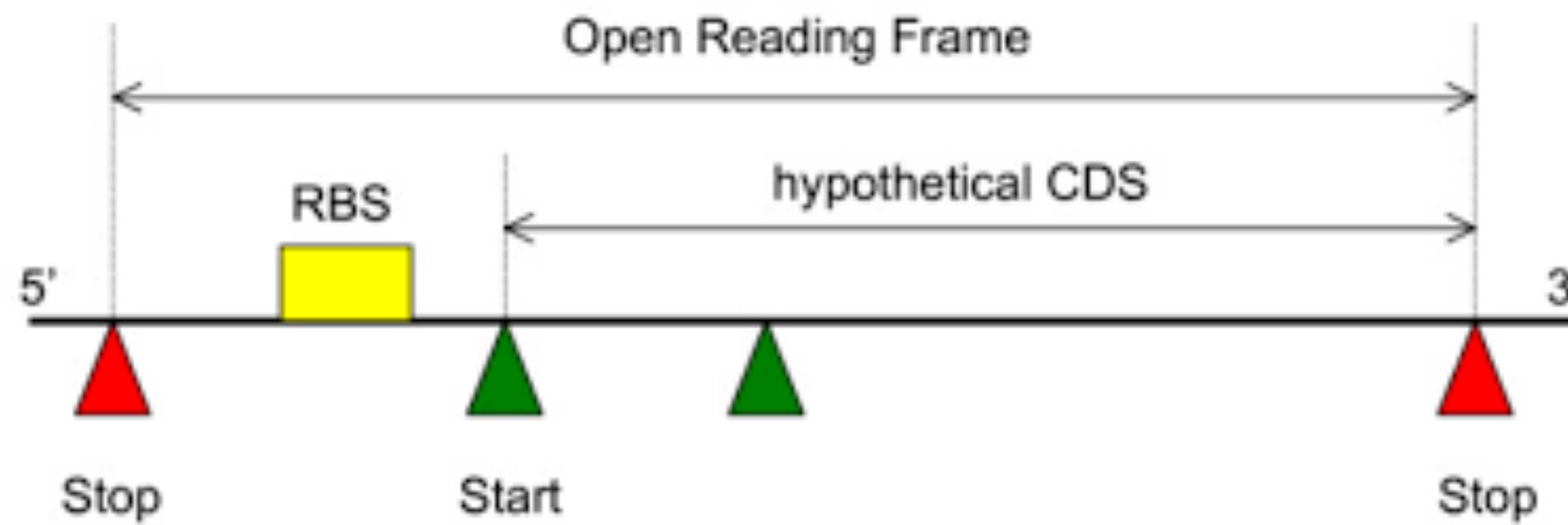
ACCGGCCGAGACAGCGAGCATATGCAGGAAGCGGCAGGAATAAGGA
AAAGCAGCCTCCTGACTTTCCCTCGCTTGGTGGTTGAGTGGACCTC
CCAGGCCAGTGCCGGGCCCCCTCATAGGAGAGGAAGCTCGGGAGGTG
GCCAGGGGGCAGGAAGGCGCACCCCCCCCAGCAATCCGCGCGCCGGG
ACAGAATGCCCTGCAGGAACCTCTTCTAGAAGACCTCTCCTCCTG
CAAATAAAACCTCACCCATGAATGCTCACGCAAGTTAATTACAGA
CCTGAAACAAGATGCCATTGTCCCCCGGCCTCTGCTGCTGCTGCT
CTCCGTCCGTCCGTGGGCCACGGCCACCGCTTTTTTTTGTGCC

transfer RNA
Leu-(UUR)

tandem repeat
 $CCGT \times 3$

homopolymer
 $10 \times T$

ORF vs CDS



Базы данных для аннотации

- Gene Ontology
 - Most well-known
 - Mostly US, but some international partners
- Enzyme Codes
 - International Union of Biochemistry and Molecular Biology
 - EU-based
- KEGG maps
 - collection of manually drawn pathway maps representing our knowledge on the molecular interaction and reaction networks
 - Japan-based
- InterPro
 - integrated database of predictive protein "signatures" used for the classification and automatic annotation of proteins and genomes
 - Associated with EMBL (i.e. EU)
- Protein Data Bank (PDB)
 - repository of information about the 3D structures of proteins, nucleic acids, and complex assemblies (US-based)

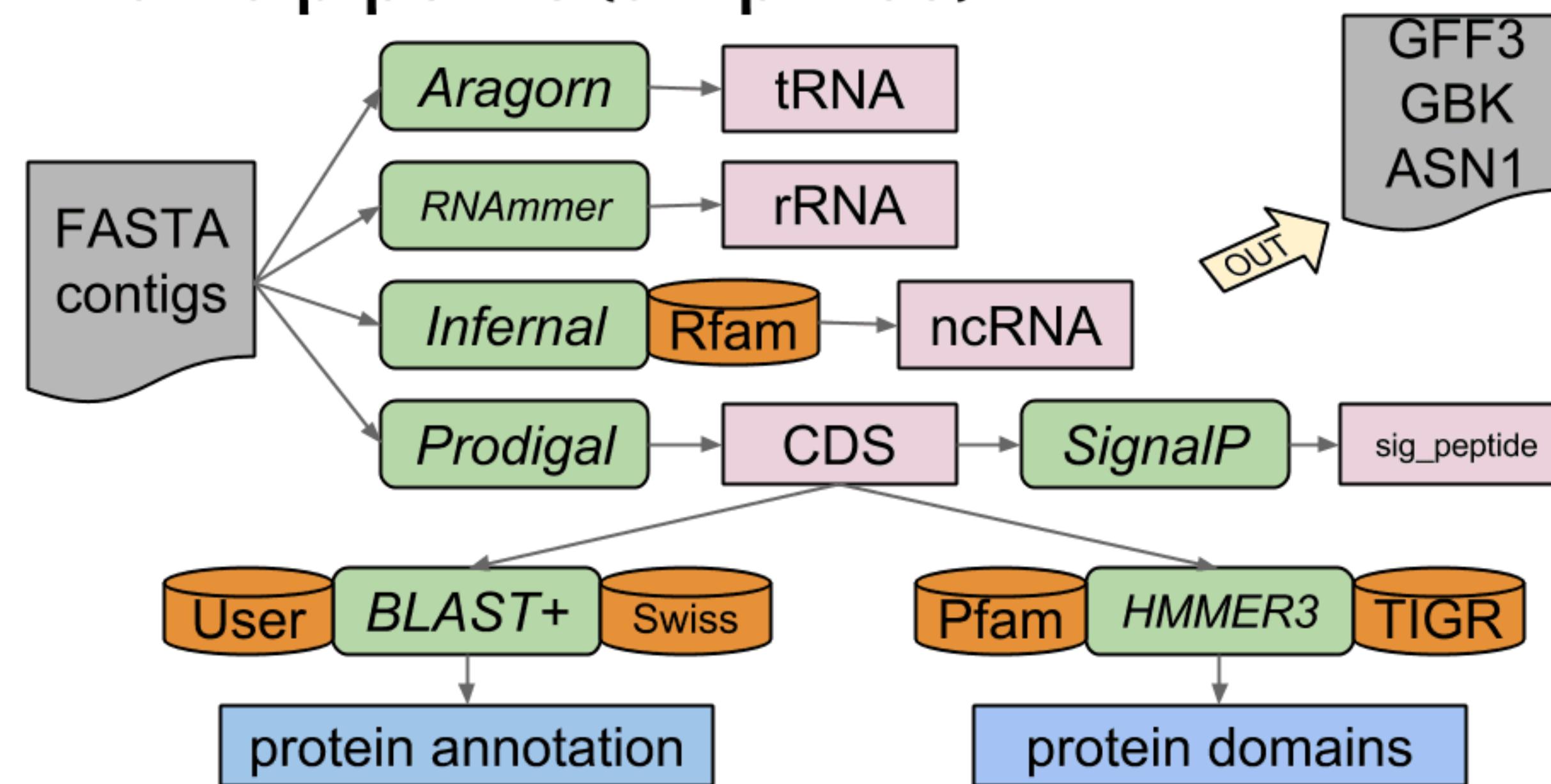
Как работает gene calling

На примере Prodigal

Триада интерпретации: identity / coverage / e-value

Как работает Prokka

Prokka pipeline (simplified)



Мы закрываем “хвосты” и идём дальше

- В прошлый раз мы **запустили пайплайн**, но не успели разобрать “как оно устроено внутри”
- Сегодня быстро закрываем 3 вещи
- алгоритмика Prokka и откуда берётся .faa
- почему эукариотическая аннотация — другая задача
- что такое GO/KEGG и почему “аннотация” ≠ просто название гена
- Потом переключаемся на variant calling (SNV/indel)

Что такое аннотация (взрослая версия)

- Аннотация = не “истина”, а гипотеза с уровнем доказательности
- Есть 2 слоя: **структурная**: где гены/фичи (CDS, rRNA, tRNA...),
функциональная: что они делают (гомология, домены, контекст)
- Любая автоматическая аннотация может ошибаться системно: плохая сборка
→ обрезанные гены → слабая гомология → много hypothetical

Prokka: что это и почему он удобен в обучении

- Prokka – быстрый аннотатор прокариот: “FASTA → готовый набор файлов”
- Он делает: gene calling (CDS), поиск rRNA/tRNA, базовую функциональную аннотацию (product/genes)
- Prokka хорош в курсе потому что: один запуск даёт все ключевые форматы

Как Prokka устроен “внутри” (конвейер)

- Вход: **сборка** (contigs FASTA)
- Внутри примерно:
 1. предсказание CDS (обычно Prodigal)
 2. поиск tRNA (обычно Aragorn) и rRNA (обычно Barrnap)
 3. сопоставление белков к базам/сигнатурам → назначение productgenes
 4. упаковка в стандартизированные outputs

Откуда берётся .faa (ключевой момент)

- Вход: **сборка** (contigs FASTA)
- Внутри примерно:
 1. gene caller даёт координаты CDS на контигах
 2. Prokka берёт нуклеотидную CDS и **переводит** в аминокислоты
 3. результат и есть .faa

Почему эукариоты — принципиально другой класс аннотации

- У эукариот: интроны/экзоны, альтернативный сплайсинг → изоформы, повторы и большие гены
- “Поиск ORF” не решает задачу: нужно восстановить структуру транскриптов
- Поэтому эукариотическая аннотация почти всегда evidence-based

Evidence-based аннотация (что значит “evidence”)

- Evidence = независимые “подсказки”, которые подтверждают модель гена:
- RNA-seq выравнивания → границы инtronов/экзонов
- long-read (Iso-Seq/ONT) → целые транскрипты и изоформы
- белки близких видов → гомология экзонных структур
- Итог: мы строим **модель транскриптома**, а не просто список ORF

Типовой эукариотический pipeline (концептуально)

- repeat masking (чтобы повторы не выглядели как гены)
- выравнивание RNA-seq / long-read к геному
- сборка/подсказки транскриптов
- gene prediction с evidence (обучаемые модели)
- интеграция в финальный GFF/GTF + QC

Зачем нужны базы функций (GO/KEGG) после аннотации

- Название белка ("product") — это удобно, но оно часто:
- неполное
- неоднозначное
- зависит от
- GO/KEGG дают второй уровень: функция как категории/пути, а не как имена.

GO: что это такое (очень кратко)

- Gene Ontology (GO) — словарь функций в 3 осях:
- Molecular Function (что делает белок как активность)
- Biological Process (в каком процессе участвует)
- Cellular Component (где находится)
- GO-термины — это “ярлыки смыслов”, которые можно агрегировать по геному

KEGG: что это такое (очень кратко)

- KEGG — база путей и модулей:
- метаболические пути
- сигнальные/регуляторные модули
- олезно для вопроса: “какие системы есть у организма”:
- дыхательная цепь?
- синтез аминокислот?
- транспортёры?
- Это превращает аннотацию в функциональный профиль

Variant Calling

Лекция 8

Валерий Павлов



Почему variant calling вообще существует

- Мы не “находим истину”, мы строим **гипотезы о вариантах** по шумным наблюдениям
- Ценность результата = доверие + воспроизводимость, а не “список мутаций”
- Поэтому у варианта всегда есть “паспорт”: QUAL/FILTER/поддержка чтениями
- Биоинформатика в целом = культура версионирования, метрик и режимов отказа

Ранняя эпоха: pileup-эвристики и боль от ложных вариантов

- Наивная логика: “в pileup большинство = вариант”
- Реальность: ошибки **структурированы** (маппинг, повторы, GC, цикл, контаминация)
- Появляется идея: “надо учитывать качества и модели ошибок”
- Из этого вырастают вероятностные коллеры и стандарты форматов

Вероятностная эпоха: качества, likelihoods

- Вариант ≠ да/нет, а **насколько вероятно**
- Генотип-лайклихуды → QUAL, GQ, PL как язык уверенности
- “PASS” = политика фильтрации, а не гарантия истины
- Ключевой навык: читать уверенность и понимать, что именно она означает

Почему indel — другая лига

- SNV часто “локален”: вопрос про одну позицию
- Indel меняет выравнивание: одна и та же область может объясняться **разными alignments**
- Ошибки чаще приходят из “геометрии”: soft-clips, повторы, низкий MQ
- Поэтому появляются haplotype-aware подходы (локальные гаплотипы вместо одной позиции)

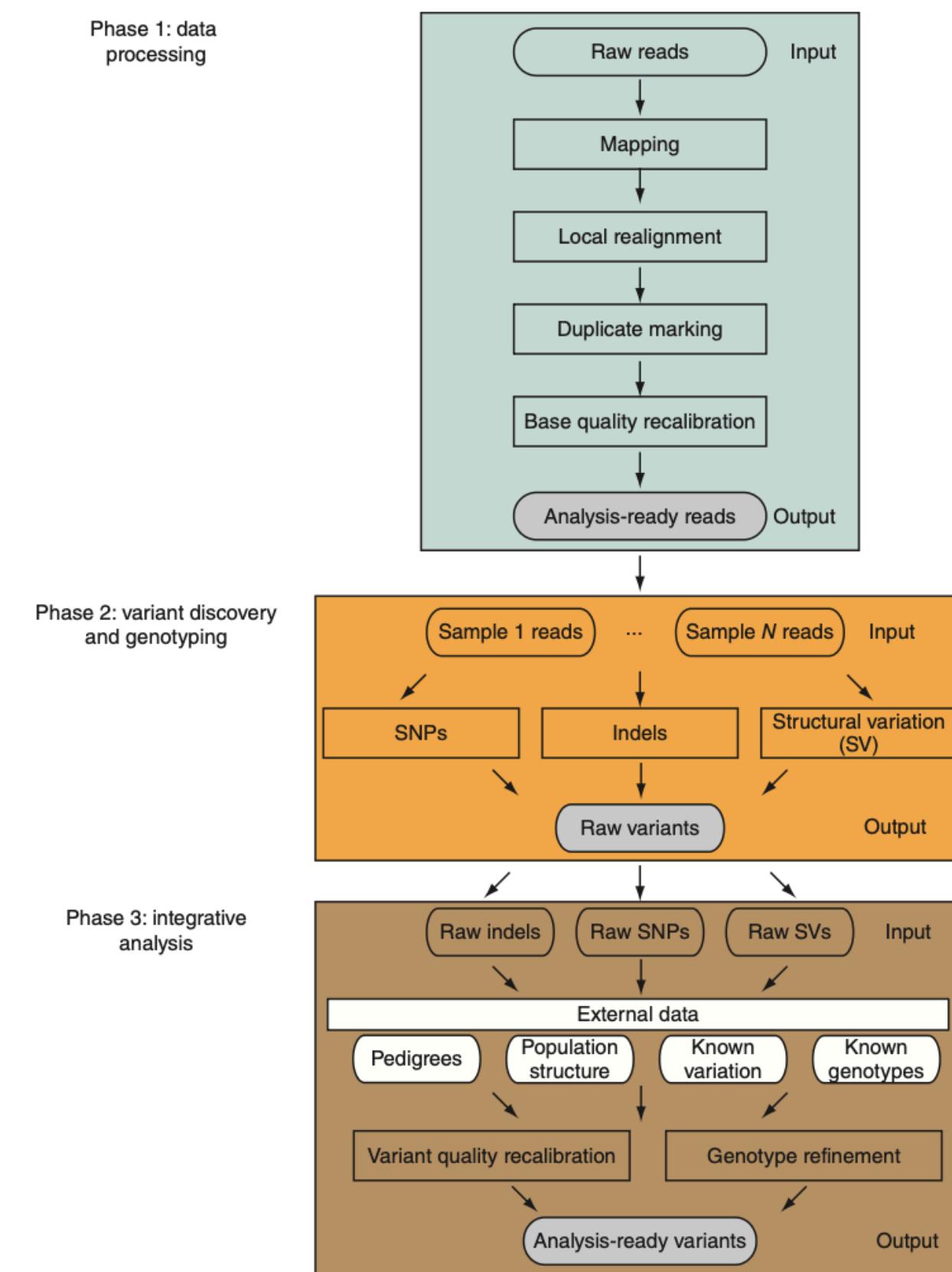
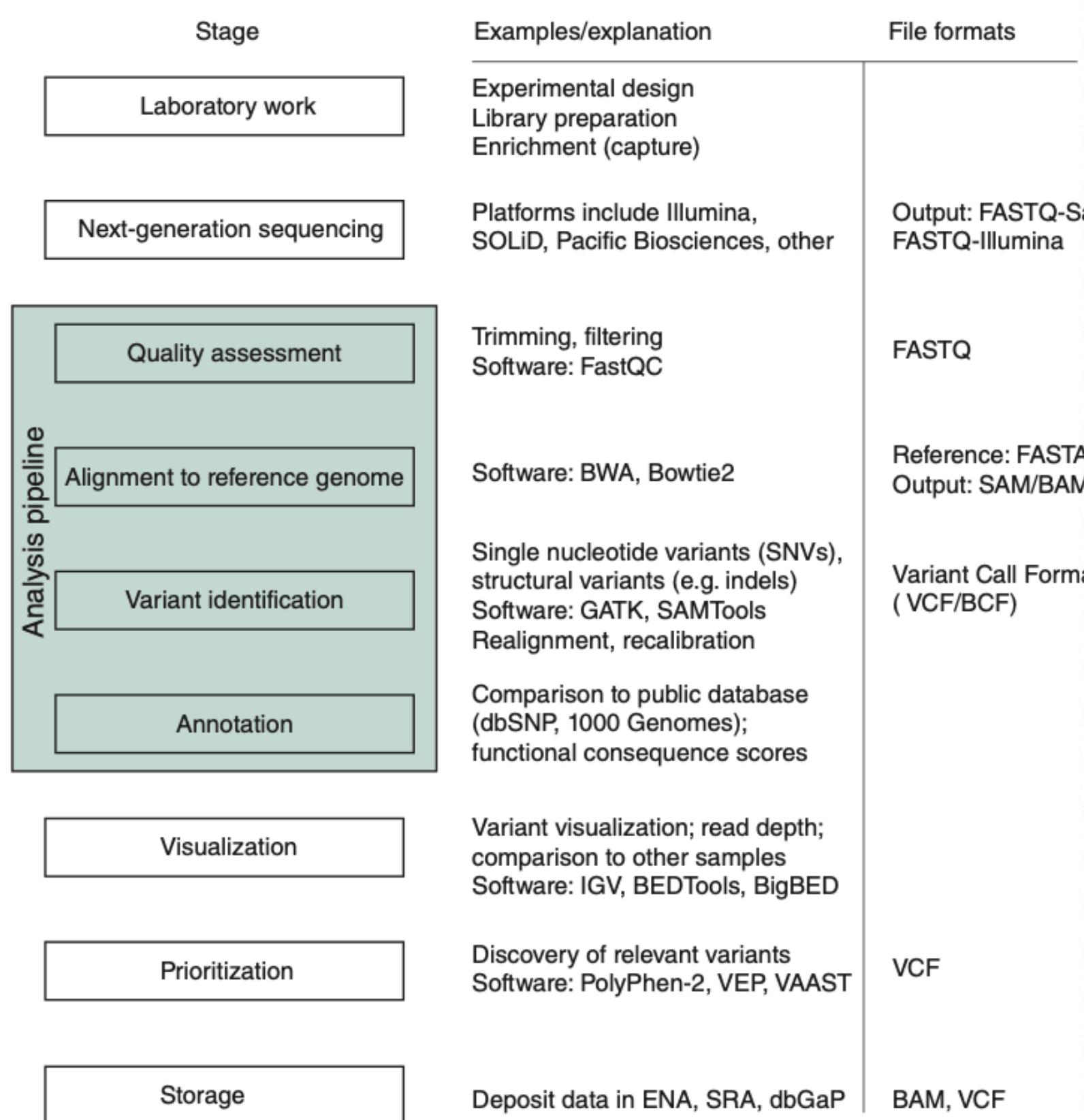
Production: почему выигрывает инфраструктура

- Победили не только “самые точные коллеры”, а **экосистема форматов**
- BAM/CRAM + индекс = быстрый доступ к локусу
- VCF/BCF + bgzip/tabix = быстрый доступ к вариантам
- Это превращает анализ в “API по геному”: быстро смотреть локусы и делать QC

Ограничения: VCF — это модель, а не реальность

- Callset зависит от: референса, aligner, параметров, фильтров, версии софта
- PASS ≠ правда; FAIL ≠ ложь (часто это “не уверен”)
- Разные callers честно не совпадают в сложных регионах
- Цель курса: научиться **защищать варианты** и понимать источники ошибок

Общая картина



Blastim.

Что такое “вариант” в NGS-практике

- Вариант = отличие от референса, поддержанное чтениями
- Вариант = гипотеза о генотипе/аллеле + уровень уверенности
- Главные наблюдения: base quality, mapping quality, strand balance, depth, allele balance
- Цель курса: научиться **защищать варианты** и понимать источники ошибок

Ключевые форматы

- FASTQ: чтения + качества
- BAM/CRAM: выравнивания (CRAM = сжатие относительно референса)
- BAI/CSI: индексы BAM/CRAM
- VCF/BCF: варианты (BCF = бинарный VCF)
- bgzip + tabix: индексируемый VCF.gz

VCF как “контракт данных”

- Каждая строка = один locus + аллеи + метаданные качества
- Header описывает, что значат поля (INFO/FORMAT)
- FILTER = решение/политика (PASS или причина)

Обязательные поля VCF (анатомия строки)

- CHROM, POS, ID, REF, ALT
- QUAL (общая уверенность), FILTER (правило), INFO (атрибуты)
- FORMAT + sample columns (генотип и метрики по образцу)

Ключевые FORMAT-поля, которые реально читать

- GT: генотип (0/0, 0/1, 1/1, 1/2)
- DP: глубина
- AD: поддержка аллелей (REF, ALT)
- GQ: уверенность генотипа
- PL: likelihoods (иногда полезно для спорных кейсов)

Multi-allelic и нормализация (почему варианты “ломают мозг”)

- ALT может быть несколько: 1/2, разные альтернативы
- Разные callers по-разному представляют сложные indel
- Нормализация (left-align, split multiallelic) упрощает сравнение

Типовой протокол SNV/indel calling

- QC reads
- alignment к референсу
- sort/index BAM
- calling (FreeBayes/GATK/etc)
- фильтры (hard filters / VQSR в больших проектах)
- QC callset (stats, Ti/Tv для человека, распределения, ручные примеры в IGV)

SNV vs indel: что проверять по read evidence

- SNV: allele balance, strand bias, base quality, MQ
- Indel: локальный realignment, повторные мотивы, soft-clips, “грязные края”
- Красный флаг: вариант сидит на границе чтений/в повторах/при низком MQ

Почему нужен фильтр (и почему “по умолчанию” нельзя верить)

- Любой коллер выдаёт кандидаты
- Фильтры превращают кандидатов в “набор для интерпретации”
- Фильтр — это выбор компромисса: чувствительность vs точность
- Поэтому важно уметь объяснить: *почему этот вариант PASS*

FreeBayes

- Haplotype-based подход: рассматривает локальные гаплотипы
- Умеет multi-allelic, хорошо для небольших проектов/быстрого обучения
- На выходе: VCF, который мы дальше фильтруем bcftools

bcftools

- view/filter/query/stats → минимальный набор для работы с VCF
- bgzip/tabix → чтобы VCF стал индексируемым
- Пайpline: “сгенерировал → сжал → проиндексировал → фильтрую/смотрю локусы”

GATK (почему все его знают)

- Исторический стандарт для small variants в диплоидных геномах
- HaplotypeCaller: локальные гаплотипы + модель ошибок
- Для больших cohorts: gVCF → joint genotyping (сегодня упоминаем, не делаем руками)

Быстрый QC callset (что смотреть за 2 минуты)

- Кол-во SNV/indel (грубая sanity check)
- Распределение QUAL/DP
- Доля PASS после фильтров
- 3 ручных проверки в IGV: “хороший SNV”, “indel”, “сомнительный”

5 красных флагков (которые ломают доверие)

- Низкий mapping quality / повторы
- Сильный strand bias
- ALT держится на 1–2 чтениях (особенно при низком DP)
- Вариант на краю чтений/в гомополимере
- Несоответствие AD/DP (странный allele balance)