Hands-on Tutorial on SNP Calling

Georg Haberer

Manuel Spannagl

Plant Genome and Systems Biology Group/PGSB

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

Types of Genomic Variation

WGD

polyploidy

aneuploidy

chromosomal mutations

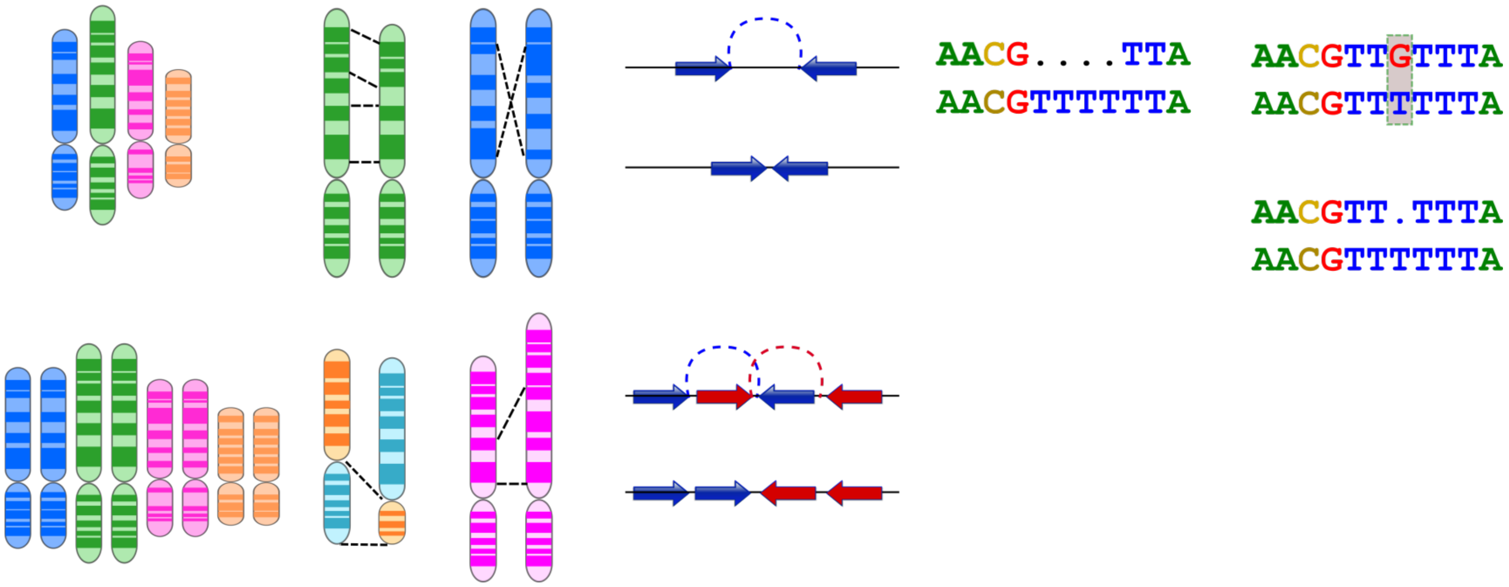
segmental
deletion
insertion
inversion
translocation

SV

~50 bp / ~Mb
deletion
insertion
inversion
translocation

indels

SNP

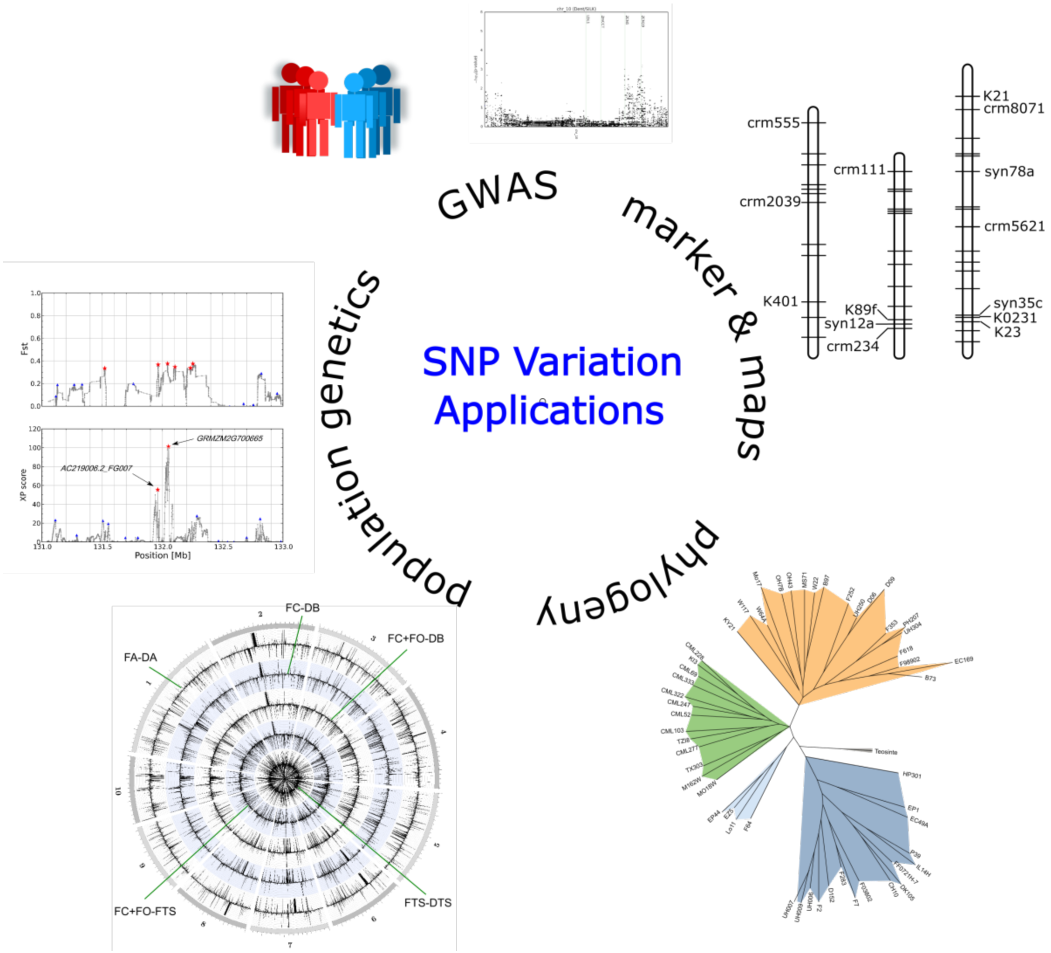


HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

(Some) Applications of Genomic Variation

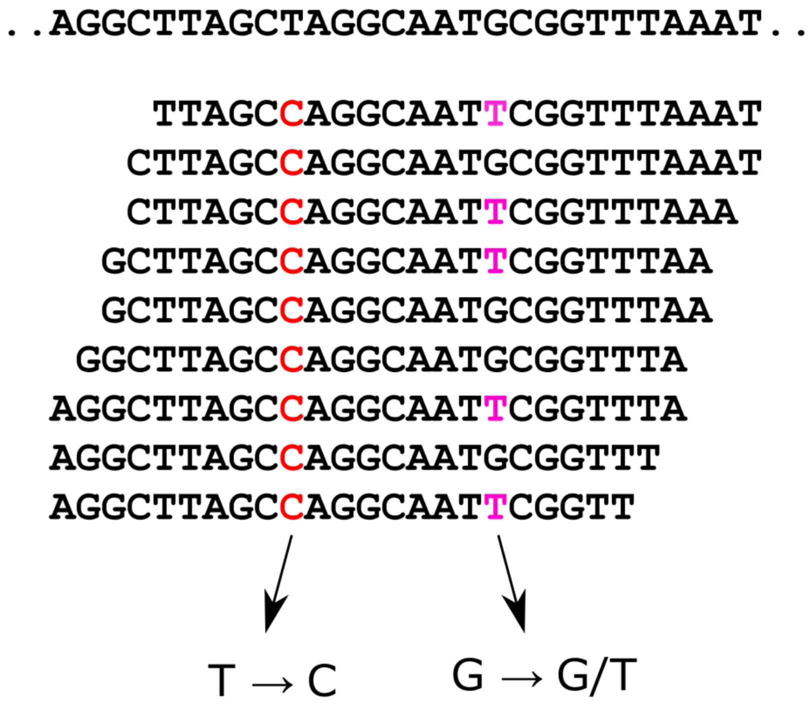
- SNPs have broad applications
- High frequency
- Advanced substitution models
  - Jukes-Cantor
  - Generalized times reversible …
- NGS: dramatic impact on SNP studies



HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

NGS Snp Calling: A Simple Task?



```
..AGGCTTAGCTAGGCAATGCGGTTTAAAT..

      TTAGCCAGGCAATTCGGTTTAAAT
     CTTAGCCAGGCAATGCGGTTTAAAT
     CTTAGCCAGGCAATTCGGTTTAAA
    GCTTAGCCAGGCAATTCGGTTTAA
    GCTTAGCCAGGCAATGCGGTTTAA
   GGCTTAGCCAGGCAATGCGGTTTA
  AGGCTTAGCCAGGCAATTCGGTTTA
  AGGCTTAGCCAGGCAATGCGGTTT
  AGGCTTAGCCAGGCAATTCGGTT
```

T → C

G → G/T

NGS Snp calling

- align the reads to reference
- read out differences

- reads are short
- genomes are complex

-> map position unique ?

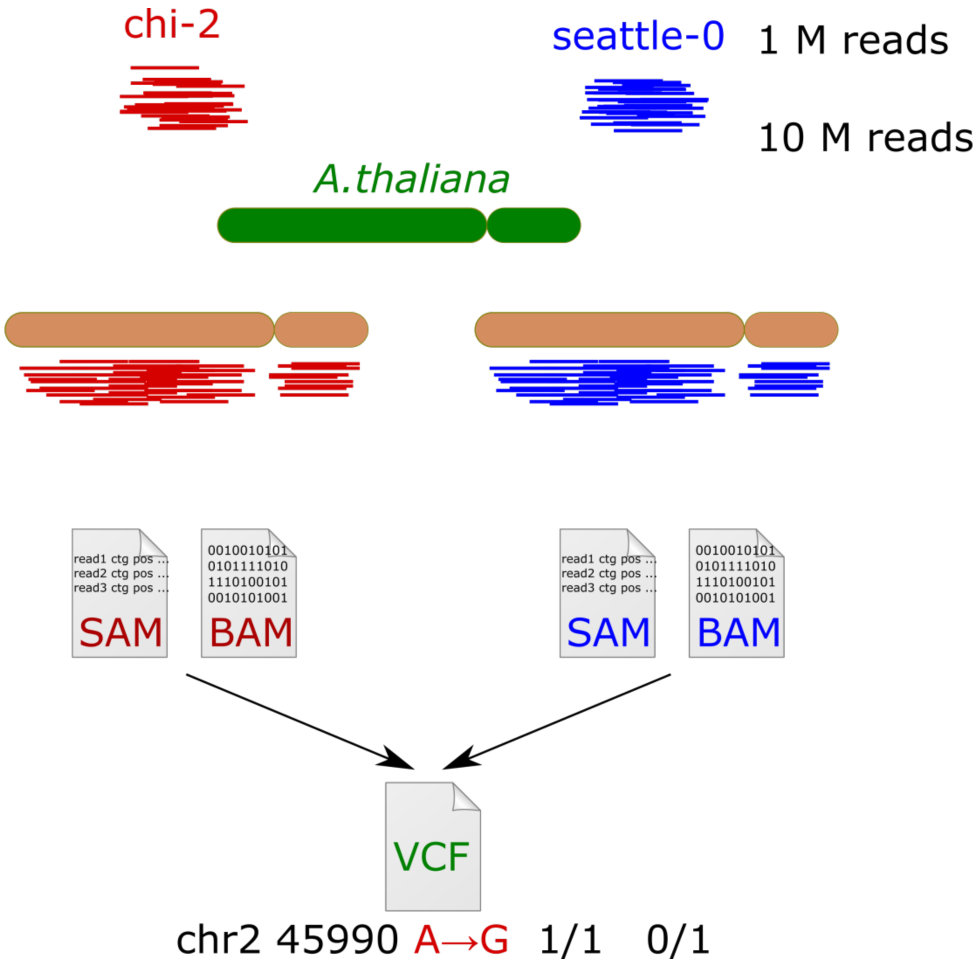
- reads are erroneous
- errors are NOT random

-> base confidence ?

Our Little Project in the Course

FASTQ reads

Program	subprogram
BWA	index align
samtools	compress sort index view
bcftools	call snps postprocess



Aims of the practical course

You will learn …

- run programs via cmd line
- sketchy understanding of underlying algorithms
- Elements of a basic SNP pipeline
- Interpret, understand and read important file formats
- Foundation to develop your own SNP pipeline

You will NOT

- Complete overview of SNP calling methods and software tools
- In-depth discussion of algorithms
- The all-in-one Swiss army knife for all possible applications, datasets and species

The FASTQ File Format

```
@FCC1DVRACXX:8:1102:12782:55474#TCTTATAT/#2
TAGTGAGATCCATGAGCCGCTGTGATTTCGCCGTATACGACATTCTCC
+FCC1DVRACXX:8:1102:12782:55474#TCTTATAT/#2
iijjfhffffeeeeeeca__^BA_[YBRRRRRRRT\][][_ACGHHHD
```

1.line: header with sequence ID

2.line: sequence

3.line: +(optional) sequence ID

4.line: base qualities, ASCII encoded phred scores

ASCII

Computers encode symbols and letters as numbers

keyboard layouts are specific to countries

universal definition:
ASCII (*American Standard Code for Information Interchange*)

ASCII table provides conversion number <-> symbol

encoding includes control characters (eg. carriage return, delete)

33	!	65	A	97	a
34	"	66	B	98	b
35	#	67	C	99	c
...	...	...	...	...	...

Phred Scores

Likelihoods p are frequently very small, eg. 10^{-190}

commonly shown as $\boxed{\log_{10} p}$ $\log_{10} 10^{-190} \rightarrow -190$

phred-scaling is an integer mapping

$$\boxed{\log_{10}(0.00253) = -2.5968\ldots \rightarrow -3}$$

$$\boxed{-\log_{10}(0.00253) = -2.5968\ldots \rightarrow 3}$$

Base Qualities in FASTQ

Base qualities are ASCII encoded phred scores according to

Sanger, Illumina>1.8	$Q = -10 \log_{10} p$	+ 33
Illumina >1.3 & <1.8	$Q = -10 \log_{10} p$	+ 64

phred	p_{error}
3	**~50%**
10	**10%**
15	**3.16%**
20	**1%**
30	**0.1%**

```
@FCC1DVRACXX:8:1102:12782:55474#TCTTATAT/#2
TAGTGAGATCCATGAGCCGCTGTGATTTCGCCGTATACGACATTCTCC
+
iijjfhffffeeeeeeca__^BA_[YBRRRRRRRT\][][_ACGHHHD
```

ASCII(f) → 102

Q(G) = 102 - 64 = 38

→ p_{error} **~ 0.016%**

http://en.wikipedia.org/wiki/FASTQ_format

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

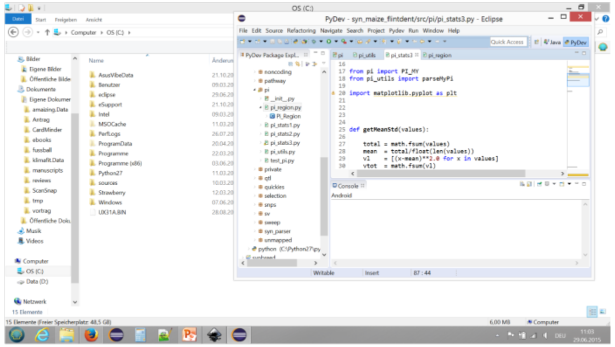
EMBL-EBI

Diversion: The LINUX command line

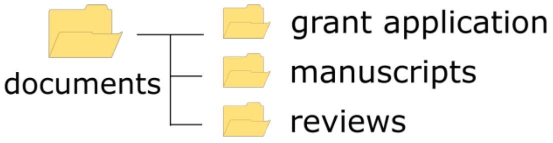
In Linux, navigation and programme executions are performed in a terminal/shell by typing commands

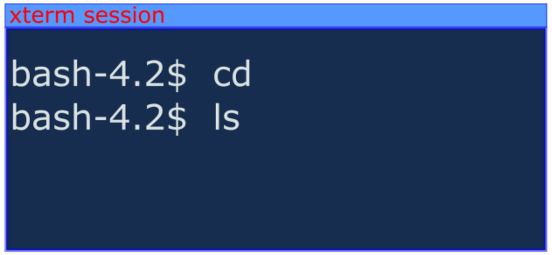
```
command options input ENTER
```

`pwd`	print working dir
`cd`	change to HOME dir
`cd <dir>`	change to `<dir>`
`ls`	ls files and dirs of current directory
`less <file>`	print file content
`<cmd> > <file>`	pipe output of cmd to new file
`<cmd1> | <cmd2>`	pipe output of cmd1 as input to cmd2



```
from pi import PI_MY
from pi_utils import parseMyPi

import matplotlib.pyplot as plt



def getMeanStd(values):

    total = math.fsum(values)
    mean  = total/float(len(values))
    vl    = [(x-mean)**2.0 for x in values]
    vtot  = math.fsum(vl)
```



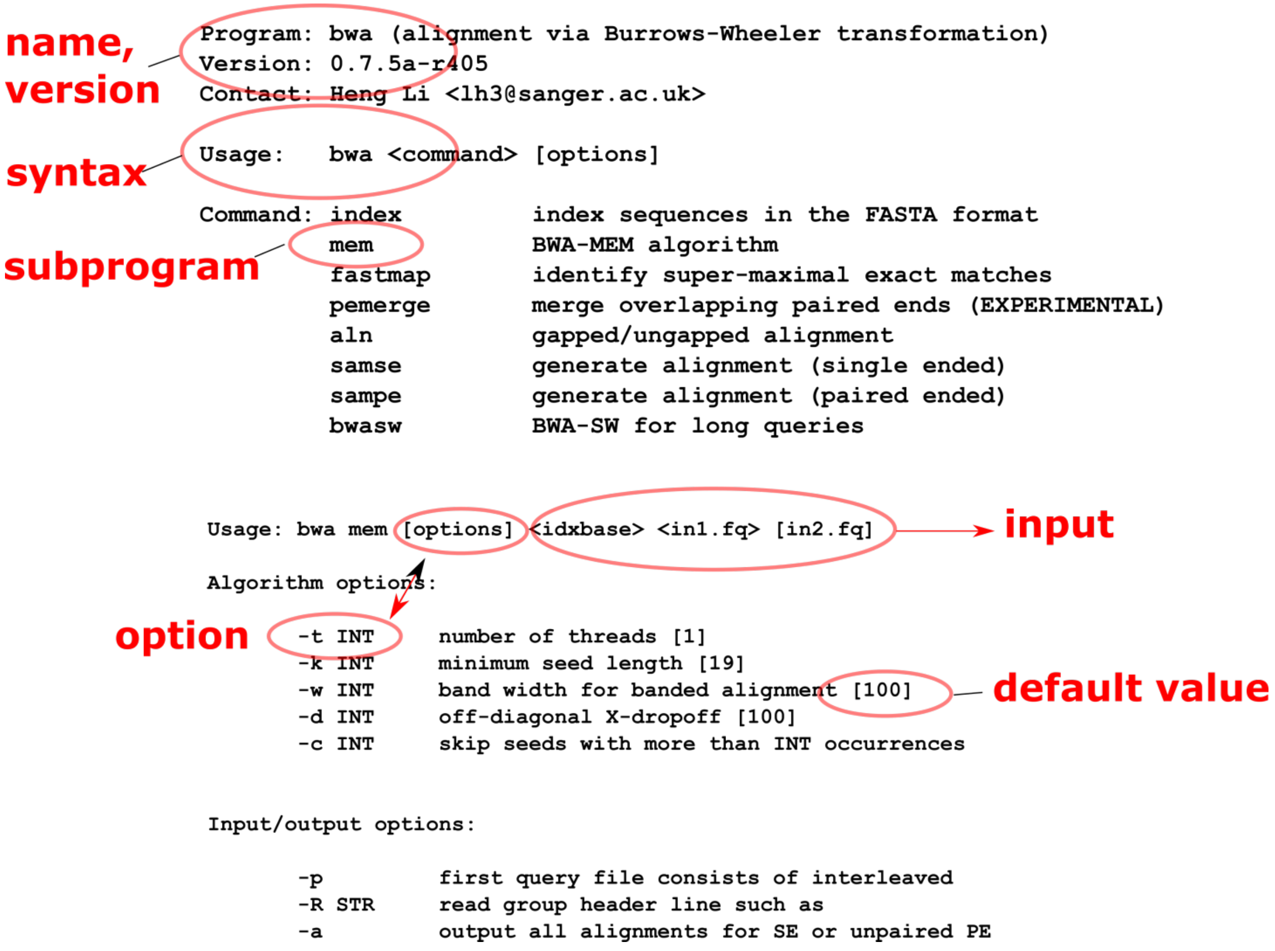


```
bash-4.2$  cd
bash-4.2$  ls
```

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

Principle Cmd-Structure of our Programms



```
Program: bwa (alignment via Burrows-Wheeler transformation)
Version: 0.7.5a-r405
Contact: Heng Li <lh3@sanger.ac.uk>

Usage:   bwa <command> [options]

Command: index         index sequences in the FASTA format
         mem           BWA-MEM algorithm
         fastmap       identify super-maximal exact matches
         pemerge       merge overlapping paired ends (EXPERIMENTAL)
         aln           gapped/ungapped alignment
         samse         generate alignment (single ended)
         sampe         generate alignment (paired ended)
         bwasw         BWA-SW for long queries


Usage: bwa mem [options] <idxbase> <in1.fq> [in2.fq]

Algorithm options:

       -t INT     number of threads [1]
       -k INT     minimum seed length [19]
       -w INT     band width for banded alignment [100]
       -d INT     off-diagonal X-dropoff [100]
       -c INT     skip seeds with more than INT occurrences


Input/output options:

       -p         first query file consists of interleaved
       -R STR     read group header line such as
       -a         output all alignments for SE or unpaired PE
```

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

Practical Part I

- Part A: The LINUX command line
- Part B: Read mapping

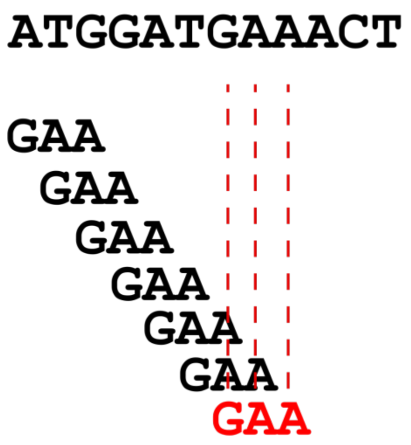
- Please finish after you have typed both commands of B.2, they will run in the background while we will proceed with the presentation

BWA and samtools

- BWA: Burrow-Wheeler Alignment
  - Short read mapper based on suffix arrays
  - Modules to map long reads
  - Generates SAM (**S**equence **A**lignment/**M**ap) format
- Samtools is a collection of programs to manipulate SAM formatted files
  - Sorting, Merging, Indexing, Viewing
- Alternative to samtools: java-based Picard toolkit
  - http://sourceforge.net/projects/picard/

Why do have to index the genome?
The Alignment Problem for NGS Data

Naive



ATGGATGAAACT

GAA
GAA
GAA
GAA
GAA
GAA
GAA

Optimal Alignments

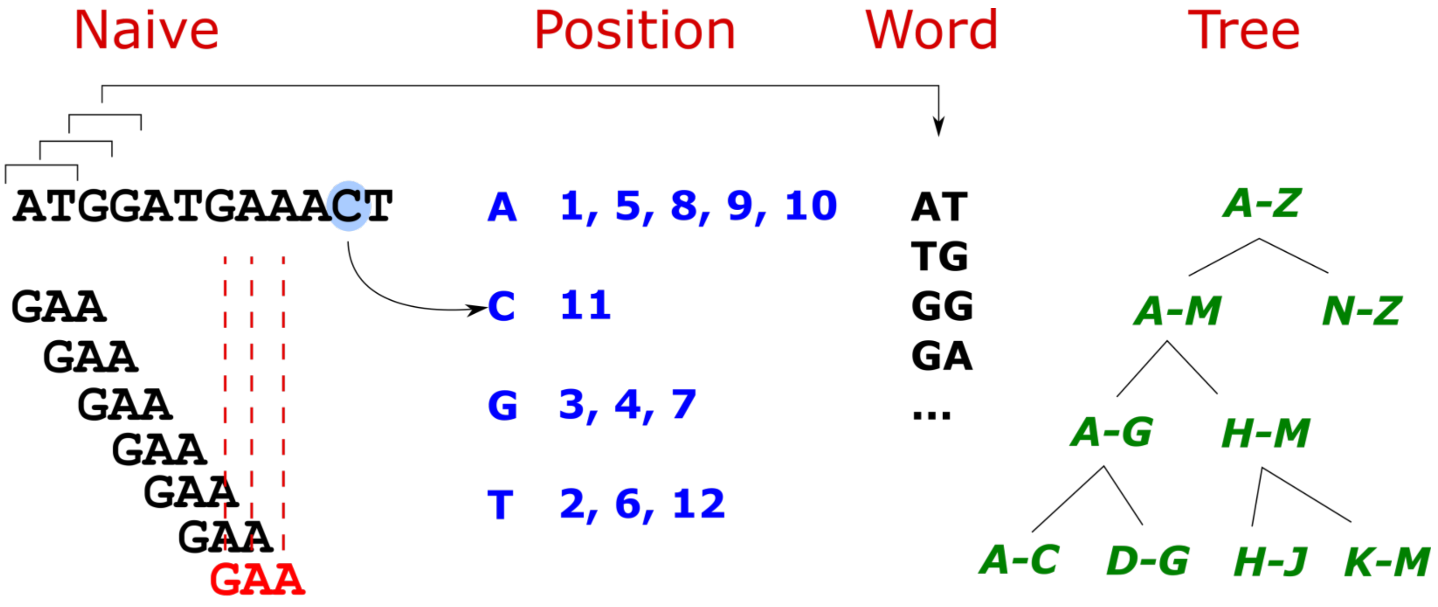
local (SW)
global (NW)

O(n*m) time & memory

For NGS experiments:

genome size	n	Mb-Gb
read length	m	100 bp
read number	N	$1x10^{8-12}$
operations	10^9x 10^2x 10^{10}	

Genome Indexing: Fast (nearly) Exact Searches



Naive

ATGGATGAAACT

GAA
GAA
GAA
GAA
GAA
GAA
GAA

Position

A	1, 5, 8, 9, 10
C	11
G	3, 4, 7
T	2, 6, 12

Word

AT
TG
GG
GA
...

Tree

Optimal Alignments

local (SW)
global (NW)

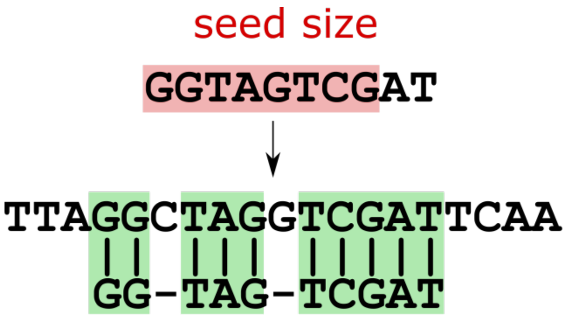
O(n*m) time & memory

Binary Trees
O(log(n))

Suffix Arrays
O(m)

NGS Aligners/Mappers

- NGS aligner are rather mappers **NOT** aligners!
- Considerations for selecting an aligner
  - Maintenance/updates?
  - PE and single reads
  - Long/short reads (miSeq, Illumina …)
  - Platform (SOLID, Illumina, 454)
  - Gapped/ungapped alignments
  - Handling of unmapped reads and multiple hits



seed size

```
      GGTAGTCGAT
            ↓
TTAGGCTAGGTCGATTCAA
   ||  |||  |||||
   GG-TAG-TCGAT
```

Bowtie

Bfast

BWA

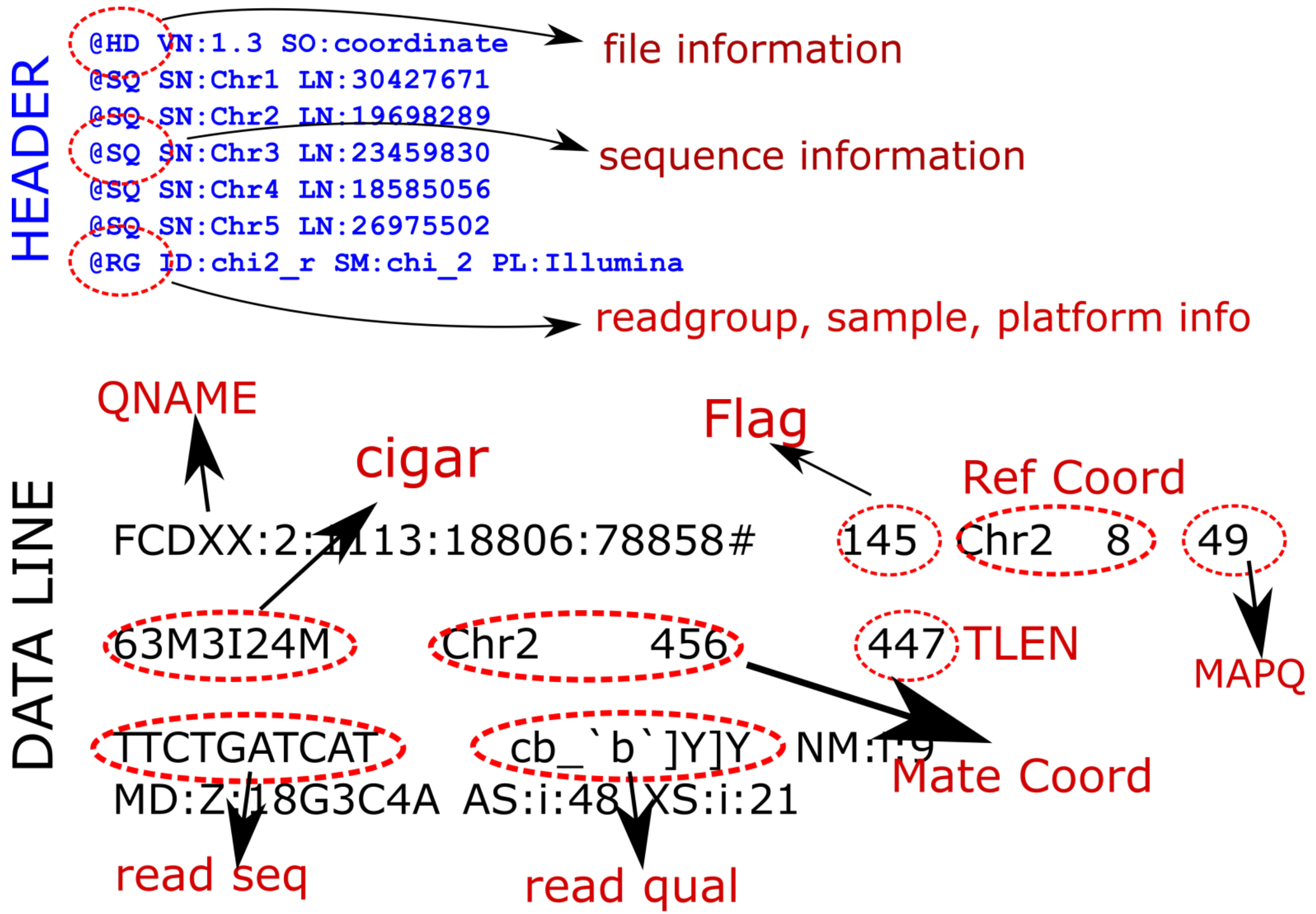
Mosaik

Stampy

Novoalign

http://omictools.com/read-alignment-c83-p1.html

SAM/BAM: The NGS Alignment Format



HEADER

```
@HD VN:1.3 SO:coordinate
@SQ SN:Chr1 LN:30427671
@SQ SN:Chr2 LN:19698289
@SQ SN:Chr3 LN:23459830
@SQ SN:Chr4 LN:18585056
@SQ SN:Chr5 LN:26975502
@RG ID:chi2_r SM:chi_2 PL:Illumina
```

file information

sequence information

readgroup, sample, platform info

DATA LINE

FCDXX:2:1113:18806:78858# 145 Chr2 8 49
63M3I24M Chr2 456 447
TTCTGATCAT cb_`b`]Y]Y NM:i:9
MD:Z:18G3C4A AS:i:48 XS:i:21

QNAME

cigar

Flag

Ref Coord

MAPQ

TLEN

Mate Coord

read seq

read qual

https://samtools.github.io/hts-specs/SAMv1.pdf

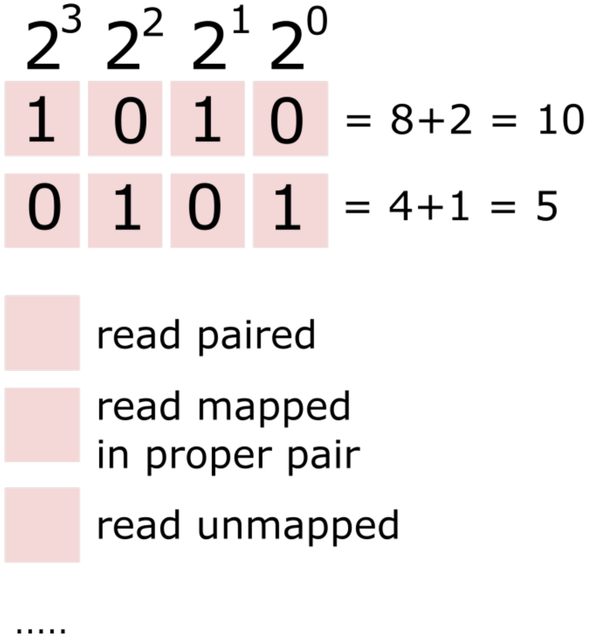
HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

SAM/BAM Format: Flags and Cigar Notation

- https://broadinstitute.github.io/picard/explain-flags.html
- Cigar notation: comprehensive notation of pairwise alignment

Flags are perfect to represent a series of independent yes/no features



2^3	2^2	2^1	2^0	
1	0	1	0	= 8+2 = 10
0	1	0	1	= 4+1 = 5

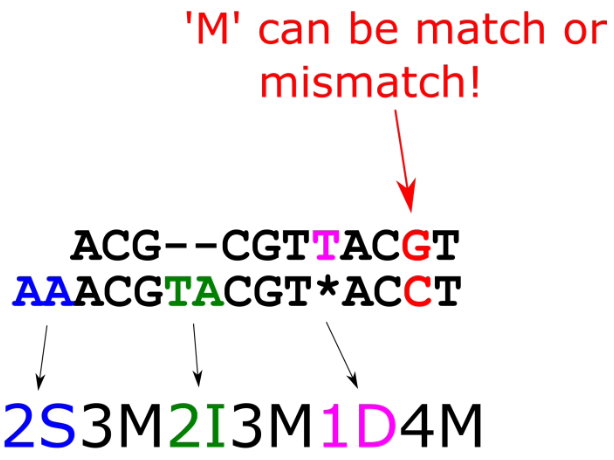
▨ read paired

▨ read mapped in proper pair

▨ read unmapped

.....

CIGAR: Reconstruction of pairwise alignments



'M' can be match or mismatch!

```
  ACG--CGTTACGT
AAACGTACGT*ACCT
```

2S3M2I3M1D4M

Practical Part II

- Please complete Part C

SNP Calling

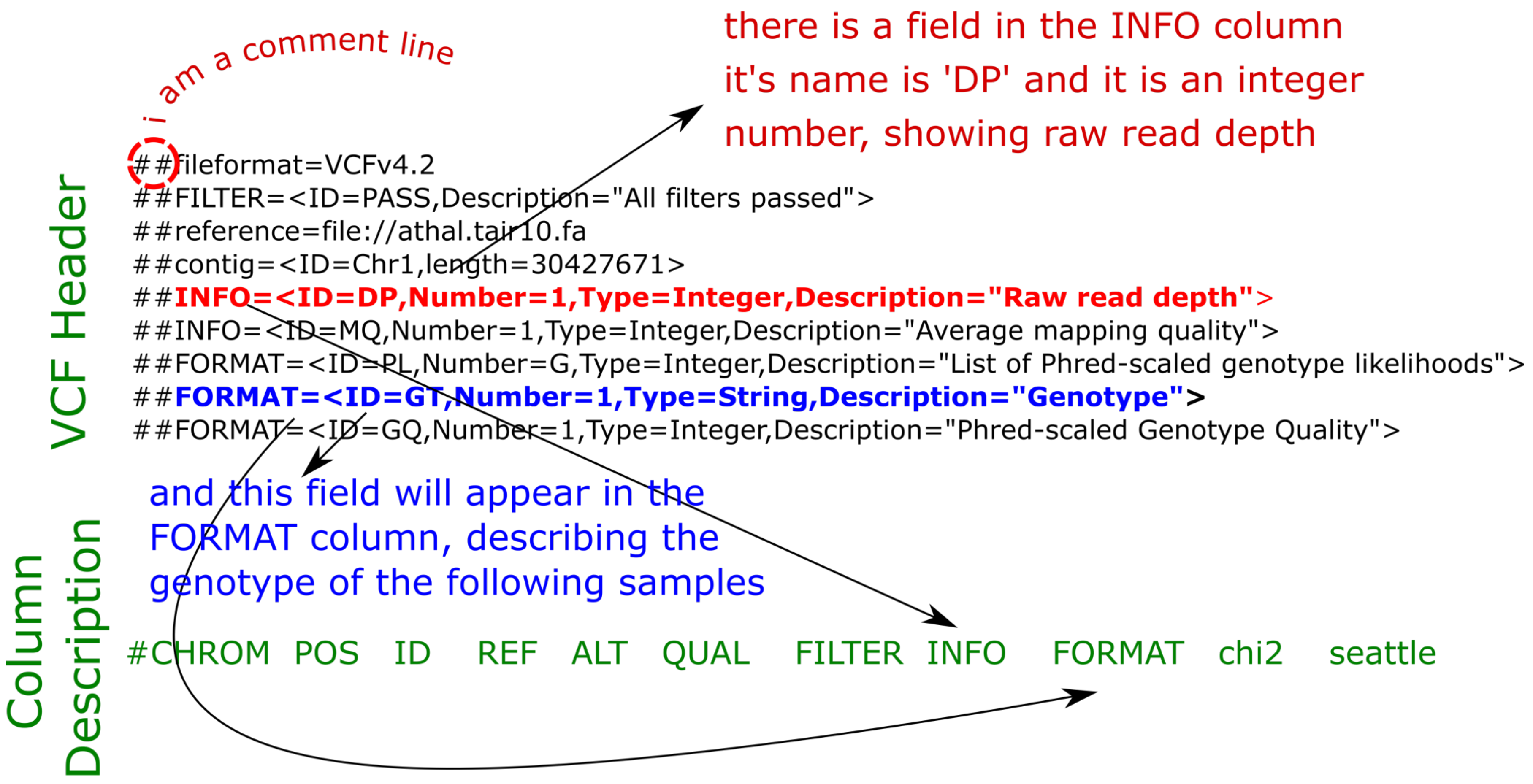
- Hardfilters
  - eg. mpileup as input
  - Use #of observations, mapping and base quality etc etc
- Bayesian/Probabilistic models
  - Use bayesian statistics to derive genotype probabilities under data observation (~read amappings)
  - Use error models
- Postprocessing
  - Hard quality filters
  - Machine learning methods
  - Training and evaluation on known SNPs (eg. 1000 genome project), literature or genotyping arrays

Bcftools and Tabix

- *Bcftools* is a collection of utilities to call SNPs and manipulate VCF (**v**ariant **c**all **f**ormat) files
  - Call SNPs and small indels
  - Annotate and subselect entries from VCF files
  - Query, filter, merge … VCF files
  - https://samtools.github.io/bcftools/bcftools.html

- *Tabix* generates indices for tab-delimited files (eg VCF)
  - http://www.htslib.org/doc/tabix.html

VCF Format (1)



VCF Header

i am a comment line

```
##fileformat=VCFv4.2
##FILTER=<ID=PASS,Description="All filters passed">
##reference=file://athal.tair10.fa
##contig=<ID=Chr1,length=30427671>
##INFO=<ID=DP,Number=1,Type=Integer,Description="Raw read depth">
##INFO=<ID=MQ,Number=1,Type=Integer,Description="Average mapping quality">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="List of Phred-scaled genotype likelihoods">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Phred-scaled Genotype Quality">
```

there is a field in the INFO column it's name is 'DP' and it is an integer number, showing raw read depth

and this field will appear in the FORMAT column, describing the genotype of the following samples

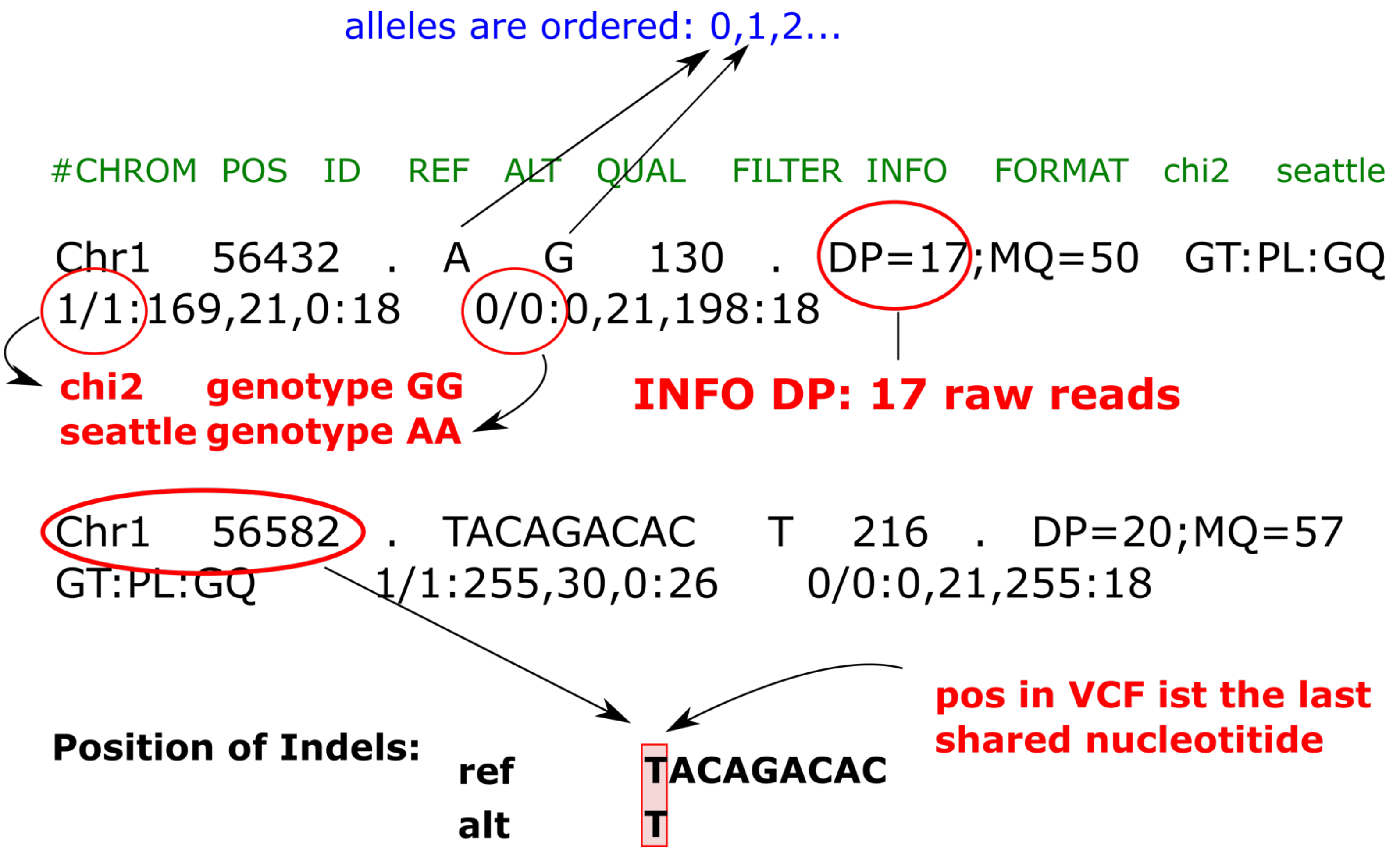
Column Description

```
#CHROM  POS  ID   REF  ALT  QUAL   FILTER  INFO   FORMAT  chi2   seattle
```

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

VCF Format (2)



alleles are ordered: 0,1,2...

```
#CHROM  POS   ID   REF  ALT  QUAL   FILTER  INFO   FORMAT  chi2   seattle

Chr1    56432  .   A     G    130    .   DP=17;MQ=50   GT:PL:GQ
1/1:169,21,0:18    0/0:0,21,198:18
```

chi2 genotype GG

seattle genotype AA

INFO DP: 17 raw reads

```
Chr1    56582  .   TACAGACAC    T    216    .   DP=20;MQ=57
GT:PL:GQ       1/1:255,30,0:26      0/0:0,21,255:18
```

Position of Indels:

ref	**TACAGACAC**
alt	**T**

pos in VCF ist the last shared nucleotitide

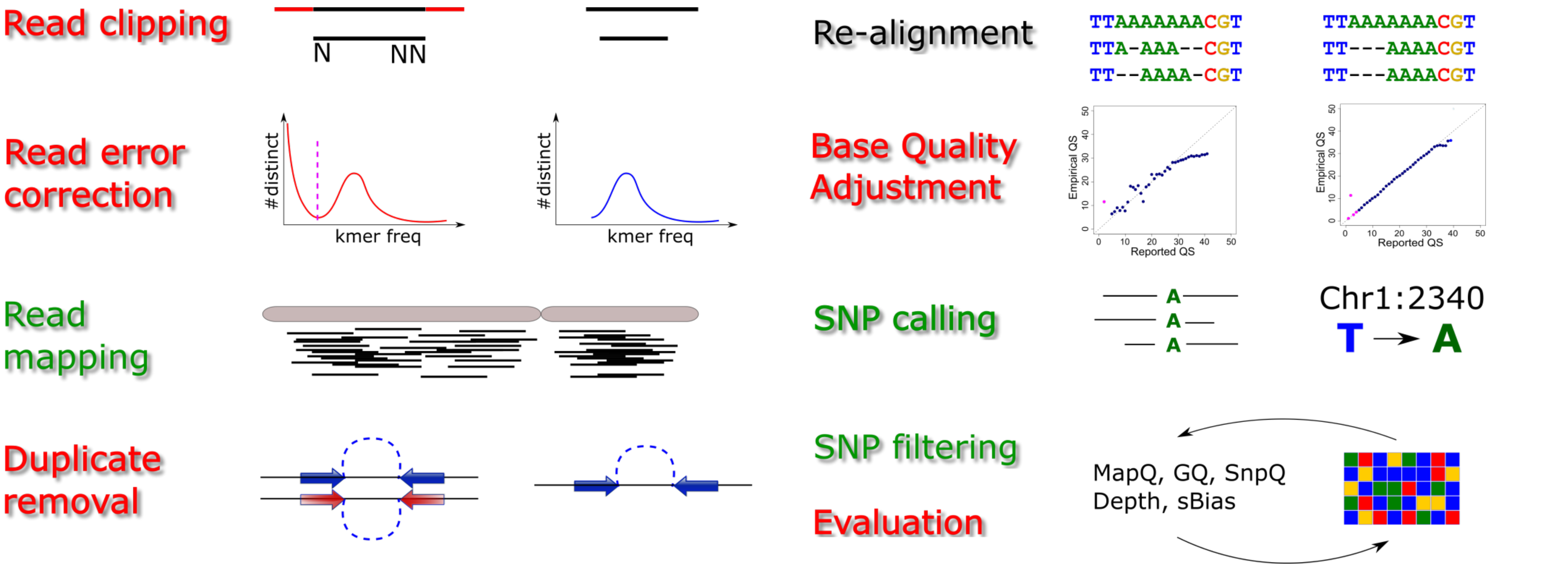
Practical Part 3

- Please finish part D + E
- After this we will have some concluding remarks,
- And you will have just developed your first basic SNP pipeline, **congrats**!

Some directions to go further …

- Look at existing workflows and software
- Bash scripts to chain your commands
- Divide & Conquer: parallelization in a batch queue
- Basic knowledge of a scripting language, eg. python

A (real) Workflow for SNP Calling



Read clipping

Read error correction

Read mapping

Duplicate removal

Re-alignment

Base Quality Adjustment

SNP calling

SNP filtering

Evaluation

HelmholtzZentrum münchen
Deutsches Forschungszentrum für Gesundheit und Umwelt

EMBL-EBI

Additional Popular SNP Callers

- **GATK**: Genome Analysis Toolkit
  - https://www.broadinstitute.org/gatk/index.php
- **soapSNP**
  - http://soap.genomics.org.cn/soapsnp.html
- **freebayes**: calls on pooled data possible
  - https://github.com/ekg/freebayes
- **varscan**
  - http://varscan.sourceforge.net/
- **Galaxy**: web-based & local, workflows
  - https://usegalaxy.org/
- Commercial products like CLS, Golden Helix