Session 3: Beginning manual curation
Curated fasta files and HiC maps production

Day 3

Genome Reference Informatics Team (GRIT)
Wellcome Sanger Institute - Tree of Life

Overview

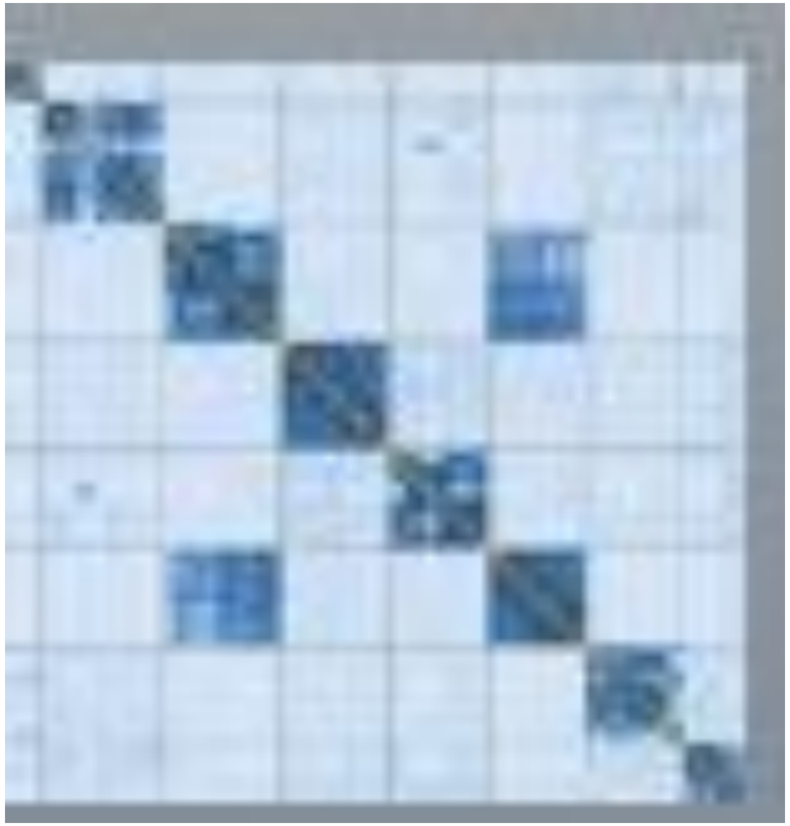
- **Some curation tricks**

- **Curation tools**
- Rapid curation workflow

  How to produce a curated fasta file

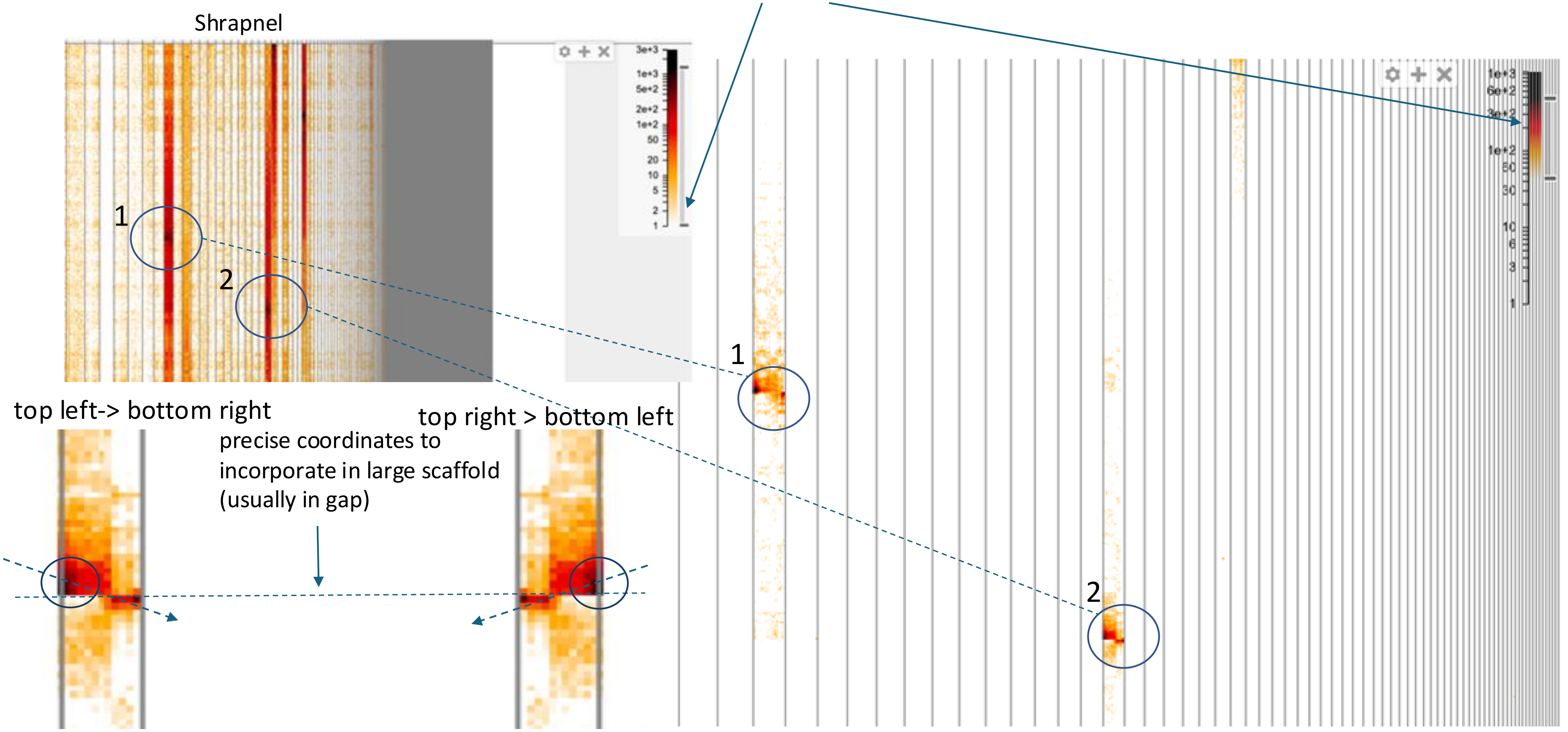
- **Analysis pipelines**

- How to generate your own PretextView Hi-C maps



Shrapnel

Incorporation of smaller scaffolds into larger ones
Usually in gaps



top left-> bottom right

top right > bottom left

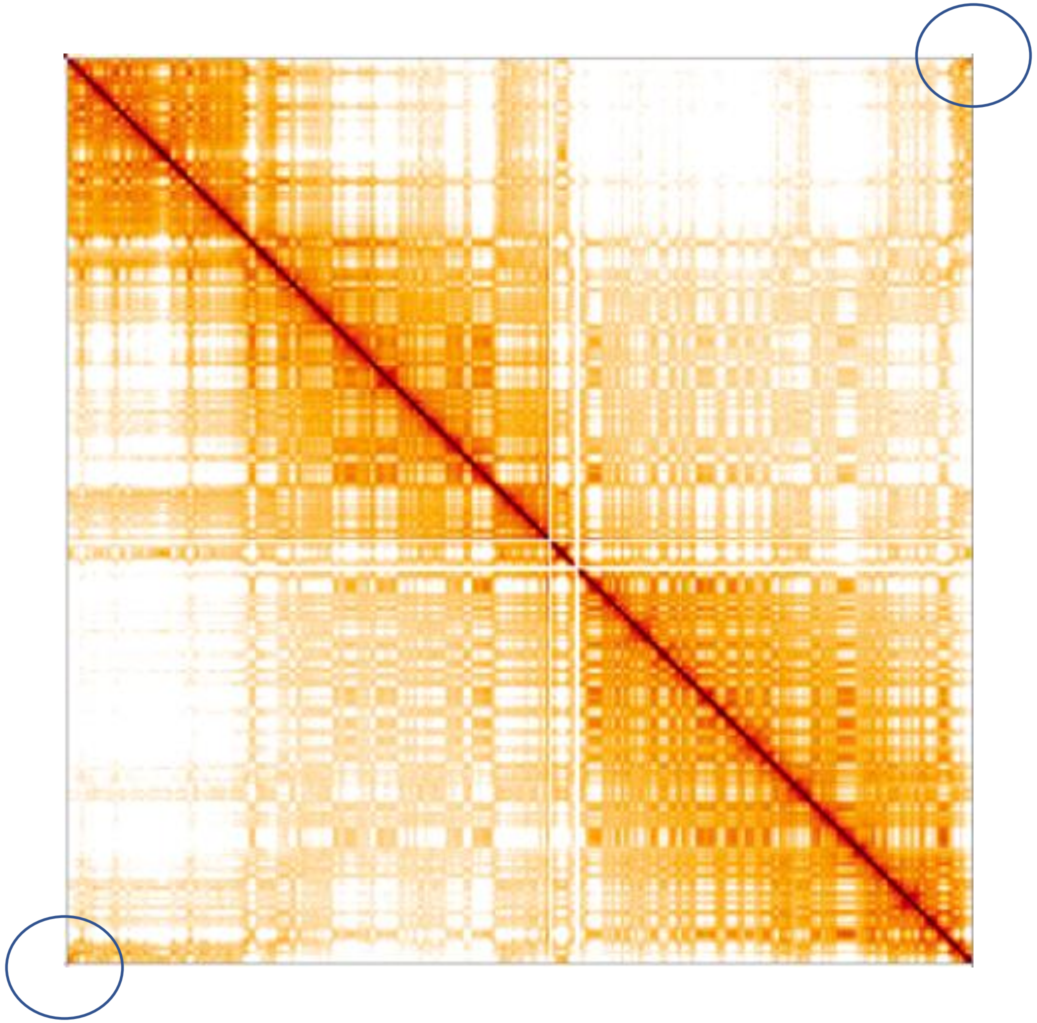
precise coordinates to incorporate in large scaffold (usually in gap)

forward orientation

reverse orientation

(Zoom in on shrapnel. Scaffolds delineated by vertical bars)

Linking between chromosome ends



mZalCal1 – scaffold4

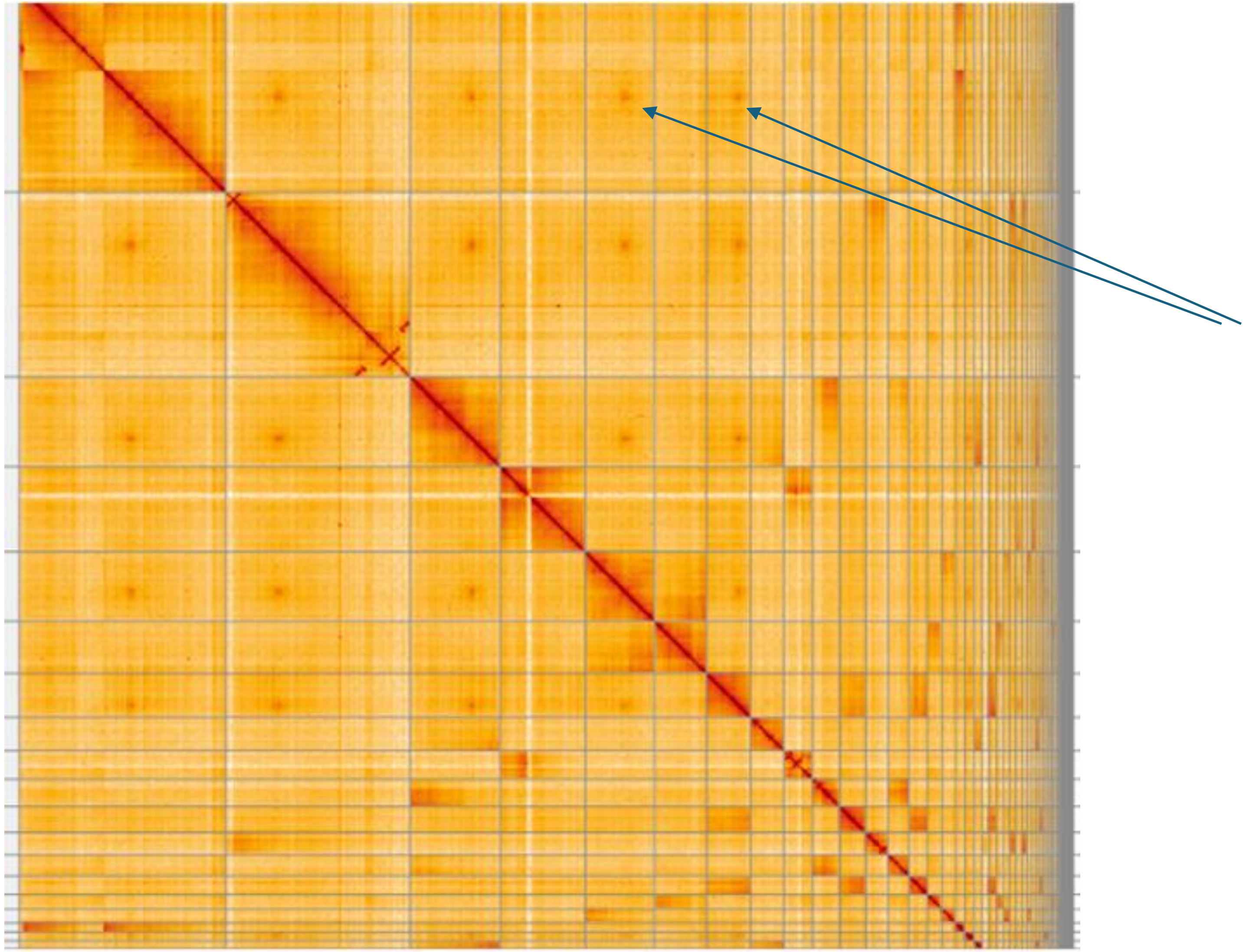
We often see affinity (ie off-diagonal signal at a level higher than we'd expect) between chromosome ends on the same chromosome. All evidence suggests that when we see this the chromosome is assembled correctly.

Telomeres are lighting each other up

↓

Usually chromosome is well assembled

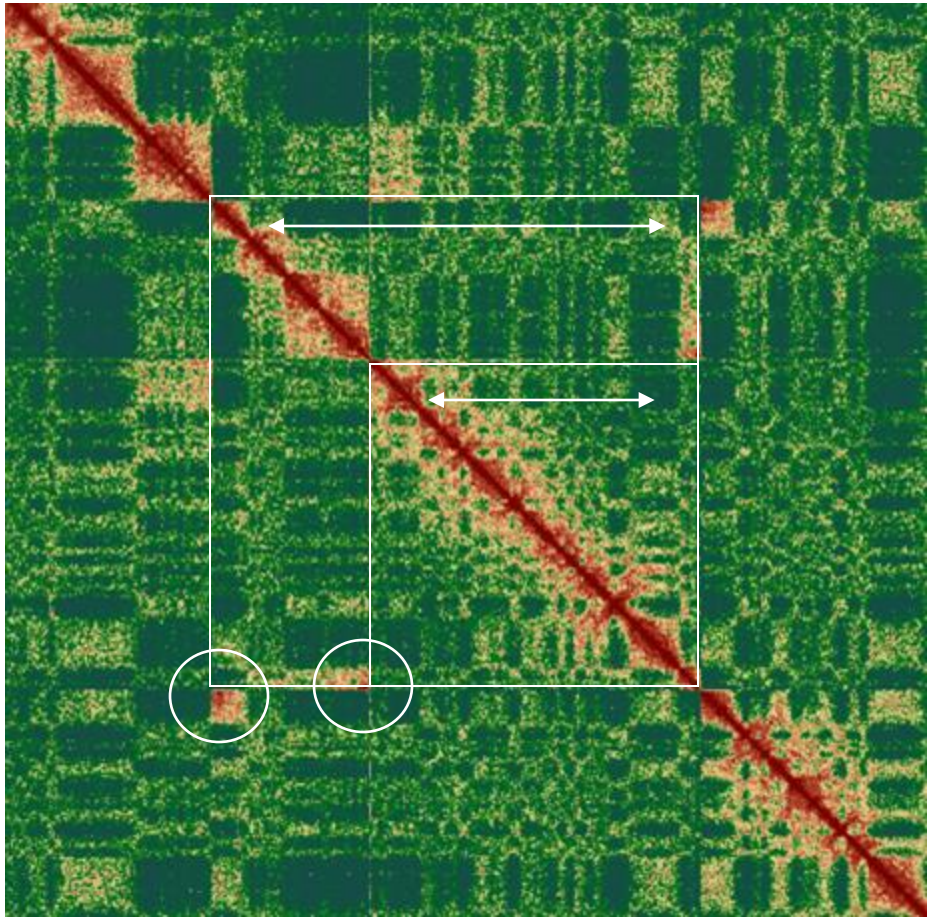
Centromeres also light each other up along the map



iHerIll2

Centromeres have been observed to be highlighted by "hot-spotting" as in these (and all the other) cases in this image.

Colour schemes



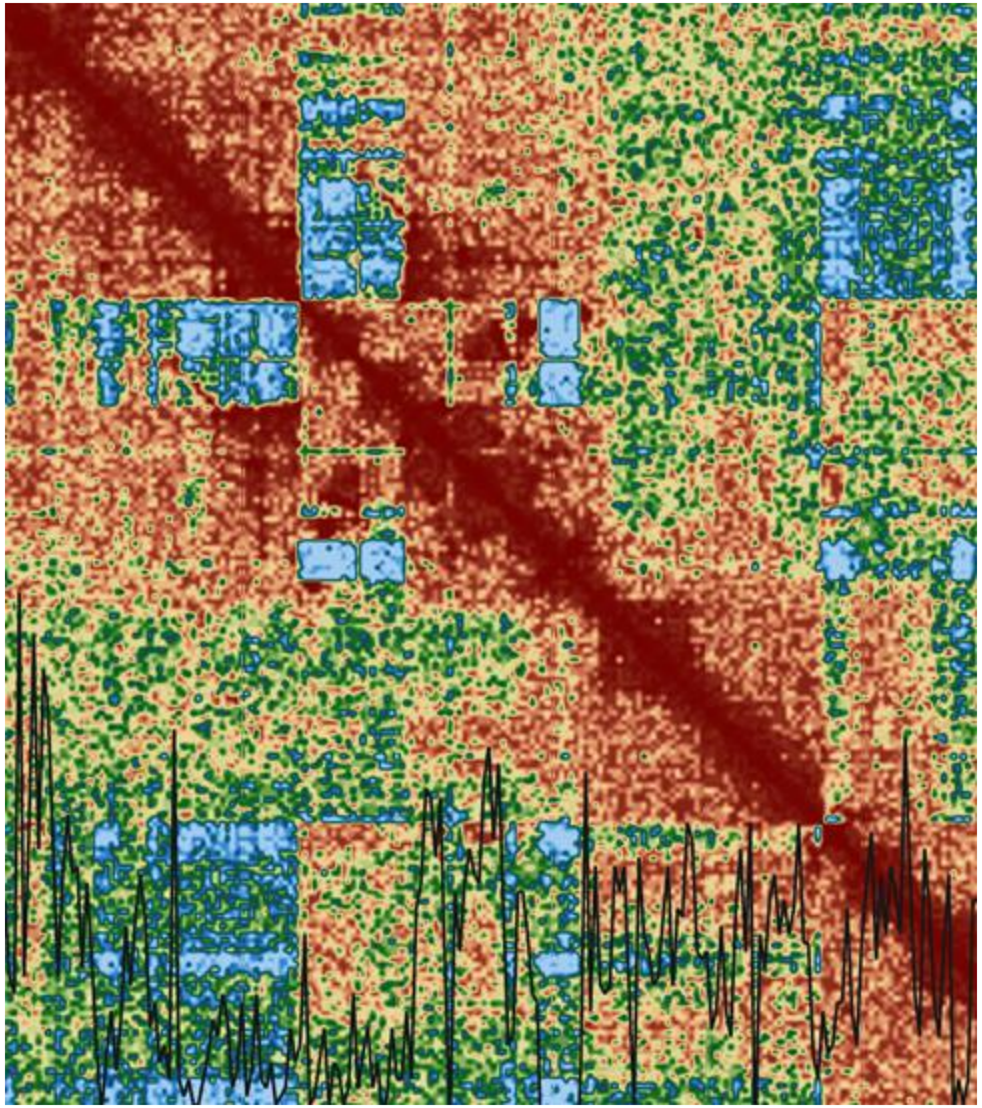
bPteGut1 superscaffold6

Choice of colour schemes is important

2 misassemblies are strongly highlighted in Pretext

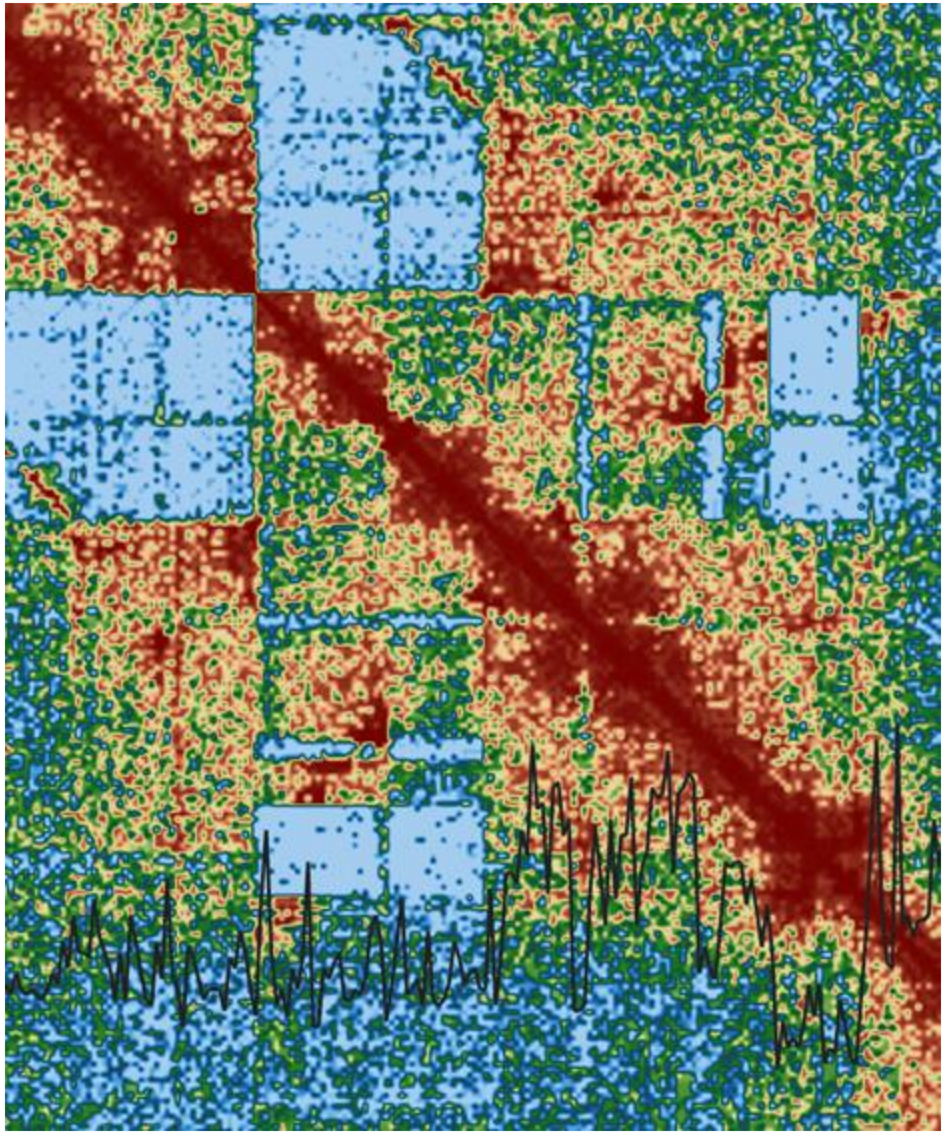
3-way colour scheme called “three wave blue-green-yellow”.

Pretext normal vs. high resolution maps – resolution issues in Pretext



Same zoom level

Works well for haplotigs



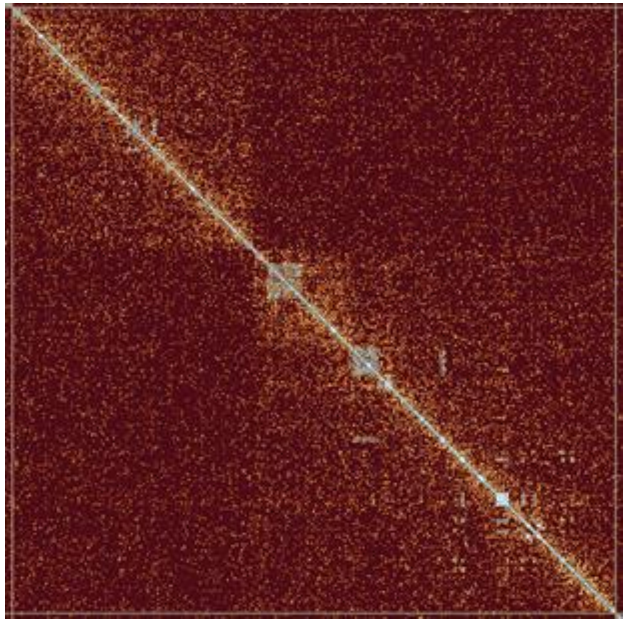
Normal resolution

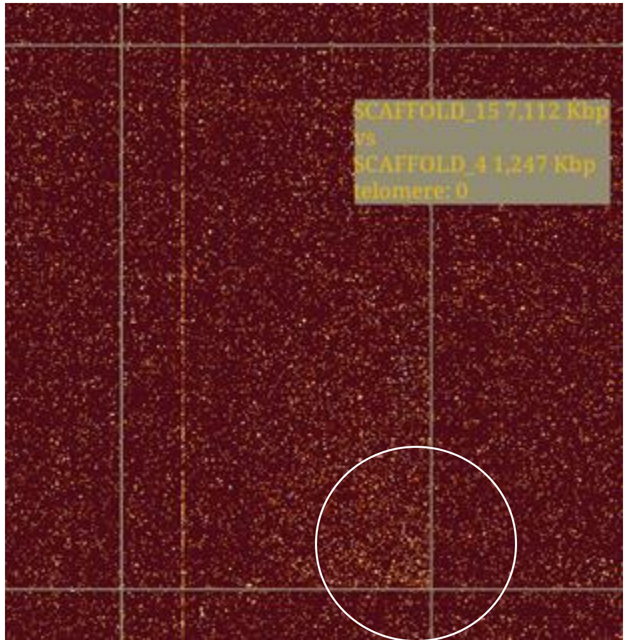
More details when you zoom-in

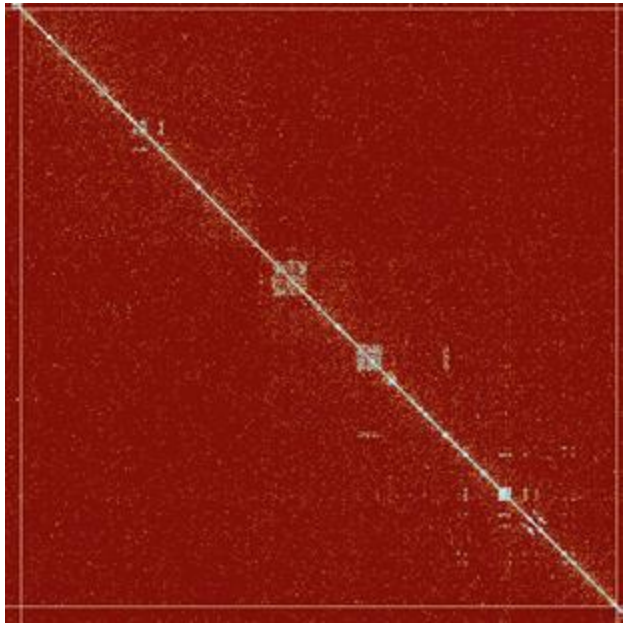
High resolution

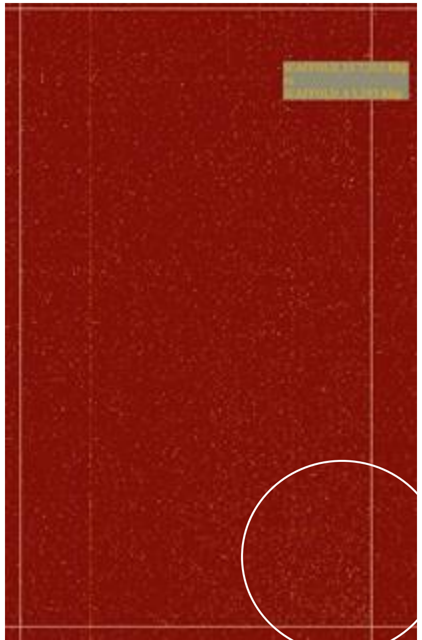
Pretext normal vs. high resolution maps – resolution issues in Pretext

Normal resolution







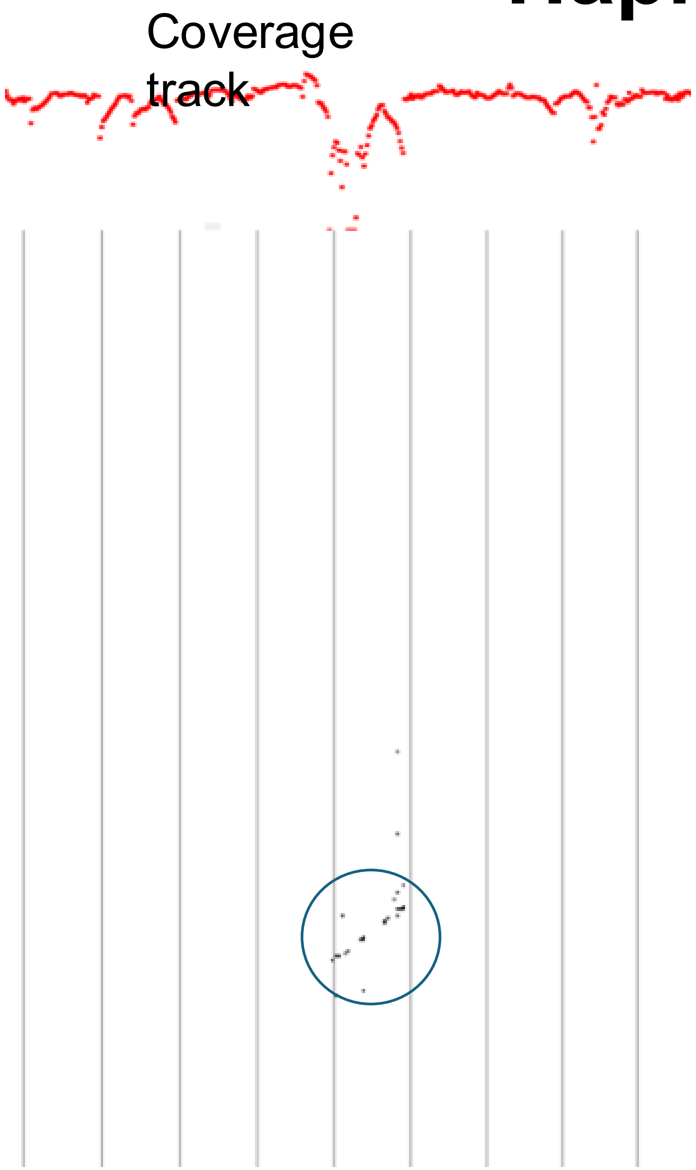


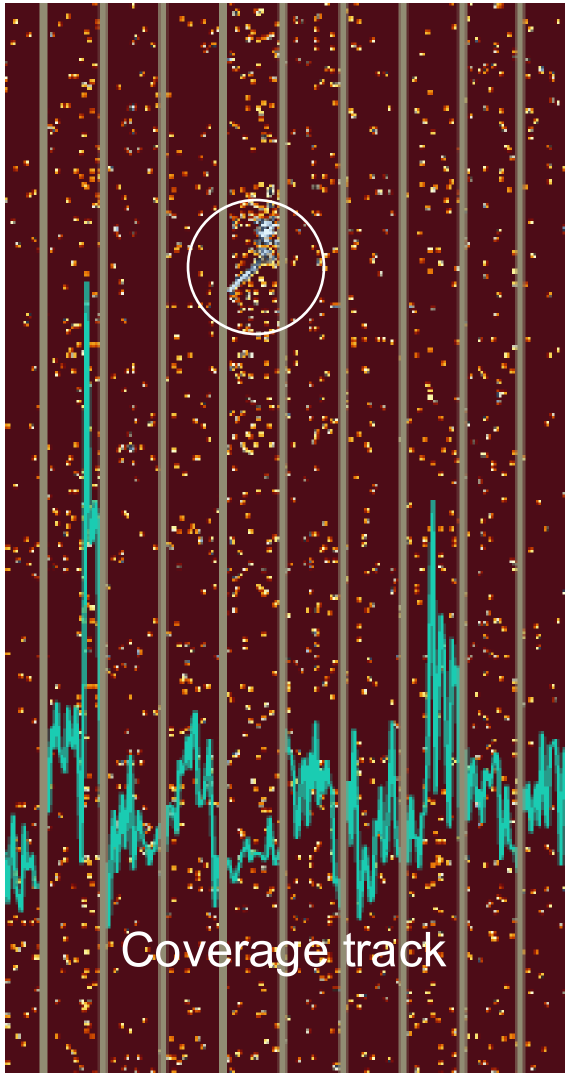
High resolution

Not ideal for joins

Poor HiC

Haplotypic shrapnel contig

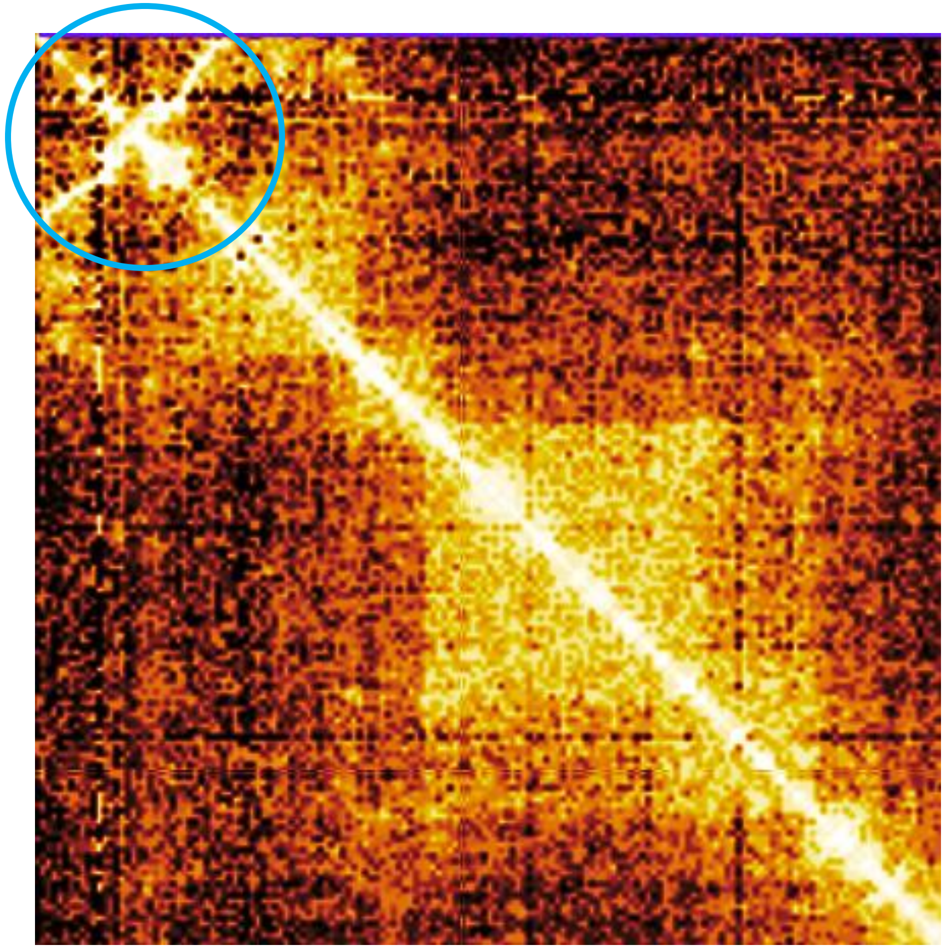




Coverage plot show the contig has half depth and the sporadic contacts are typical of a haplotypic contig. From this plot, you can see that the haplotype is entirely contained in the chromosome in the reverse orientation.

(Remember – top right-> bottom left is always reverse orientation and top left-> bottom right is always forward orientation)

Inverted haplotypes



iAphHyp

Here we have a haplotypic duplication giving rise to an unusual HiC signal suggestive of an inverted repeat. When we inspect the read coverage, it's clear that this is half what it should be for most of this region.

How to produce your curated fasta file?

The finishing process – painting

After curation you should:

Add all relevant metadata tags

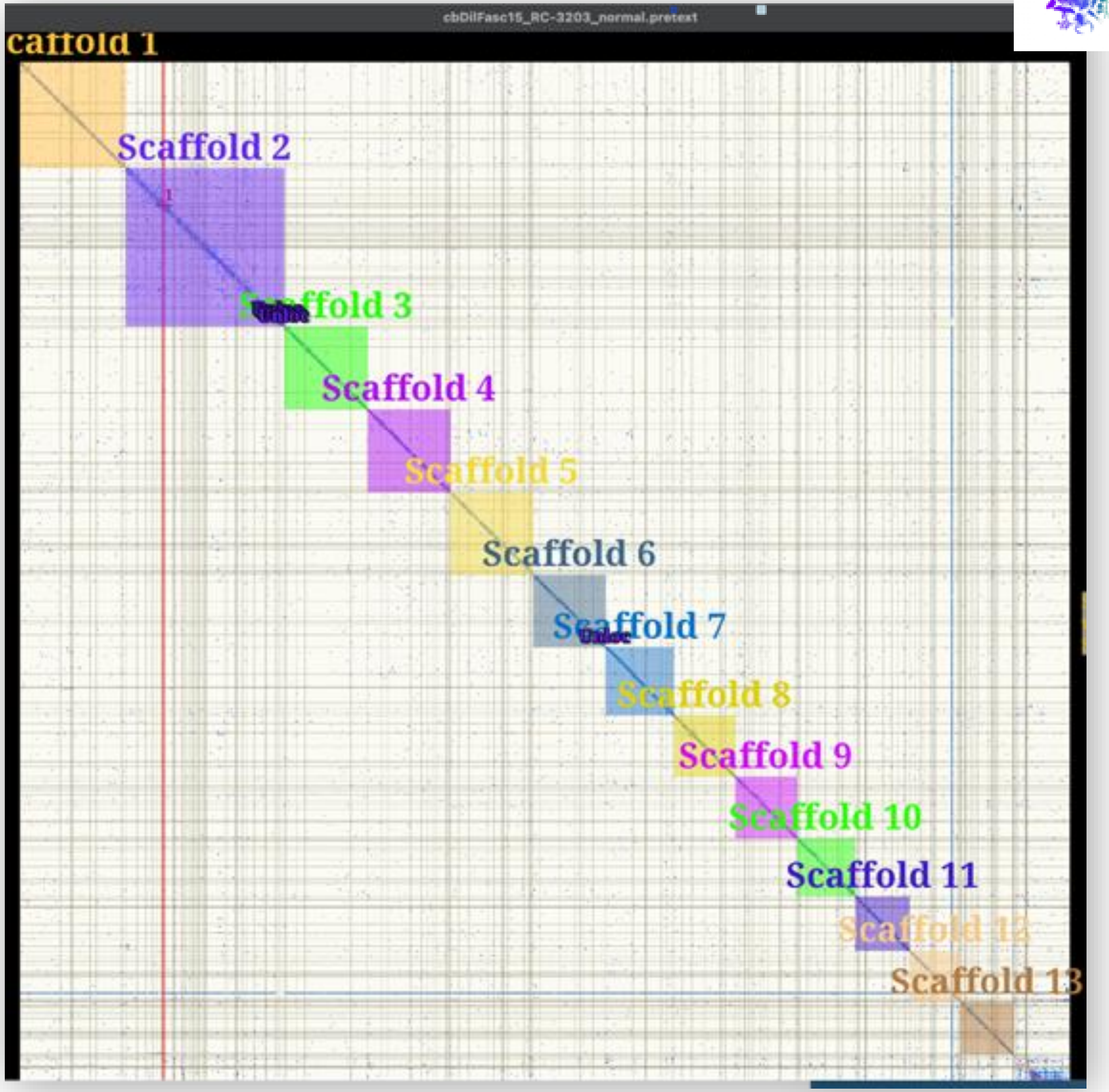
Paint chromosomes

⬇

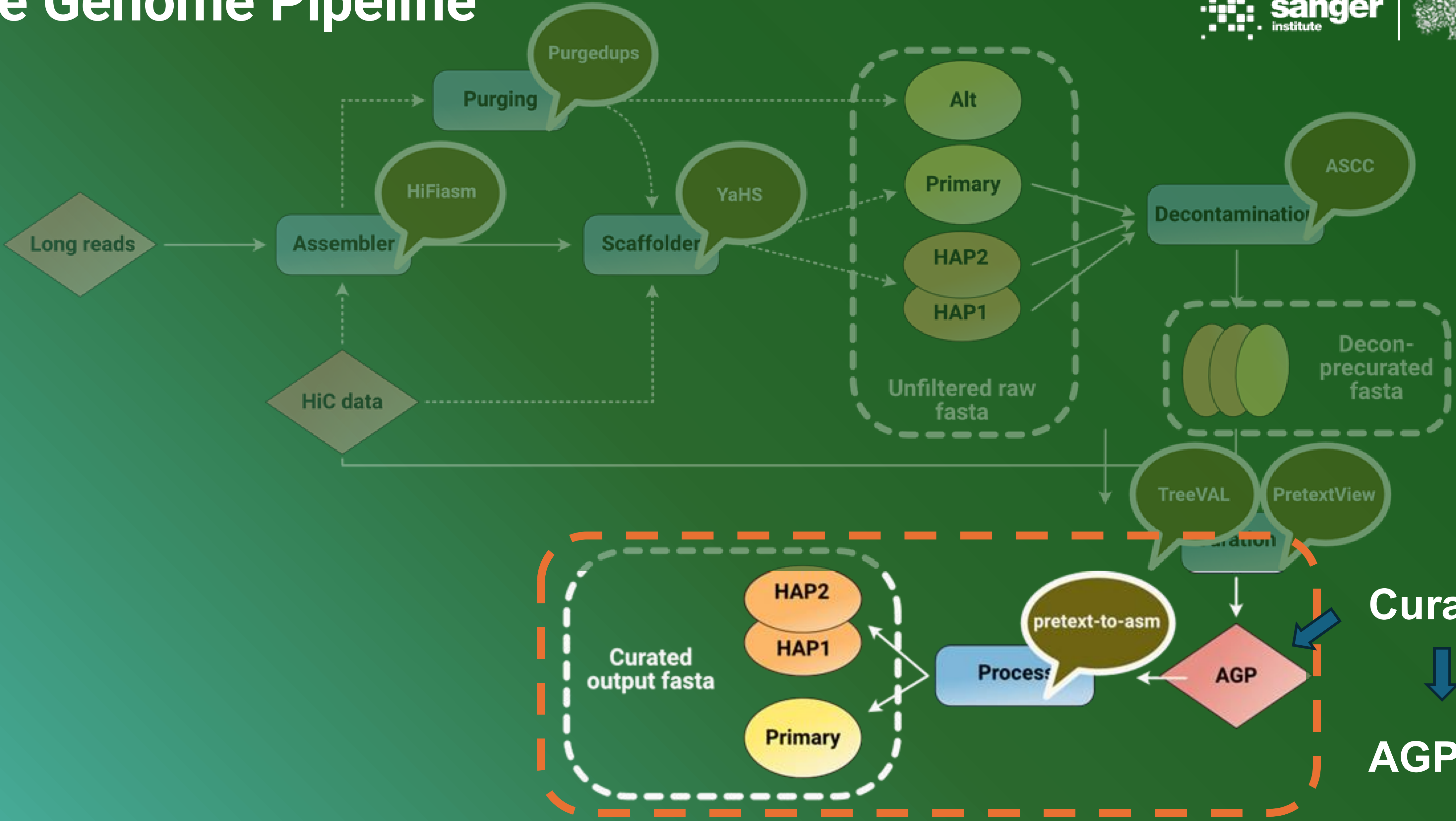
AGP and savestate generation

⬇

Curated fasta file



The Genome Pipeline

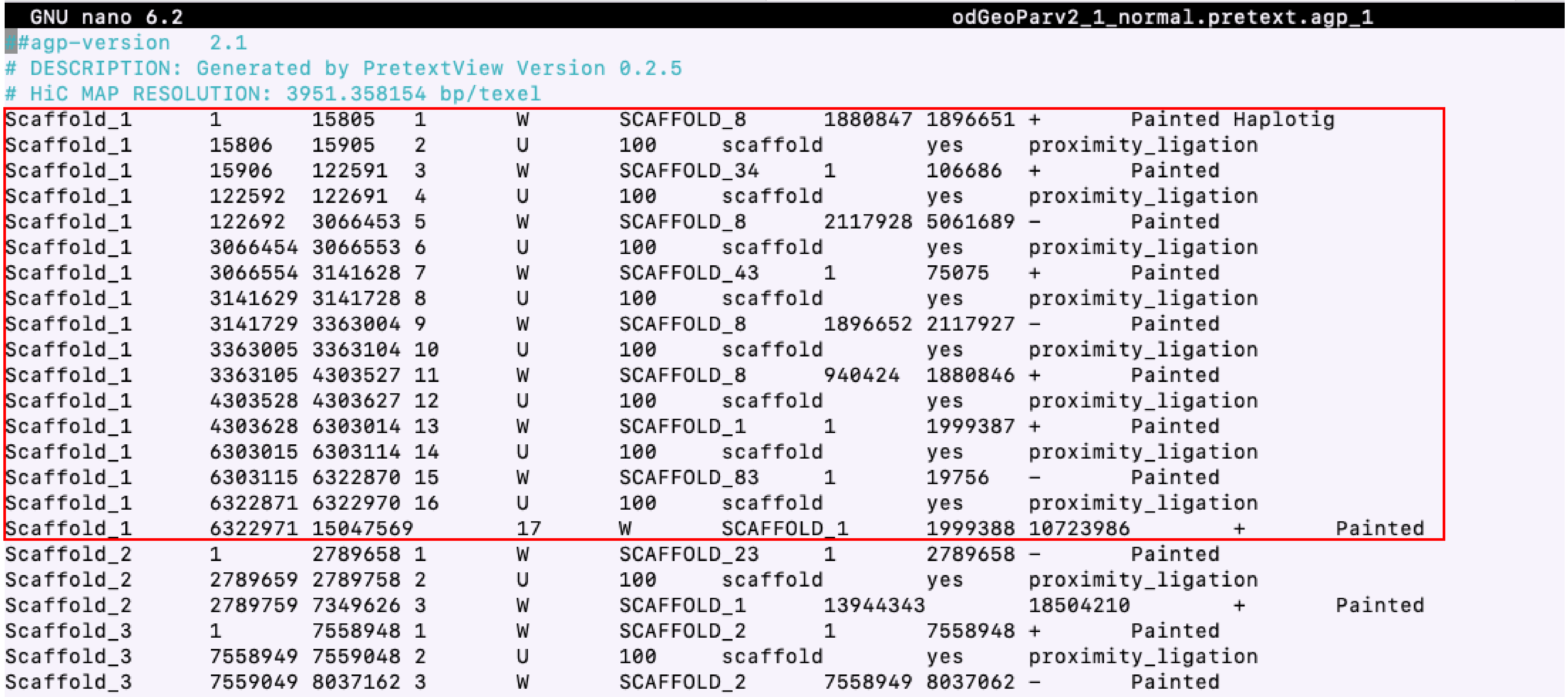


Curation

AGP file

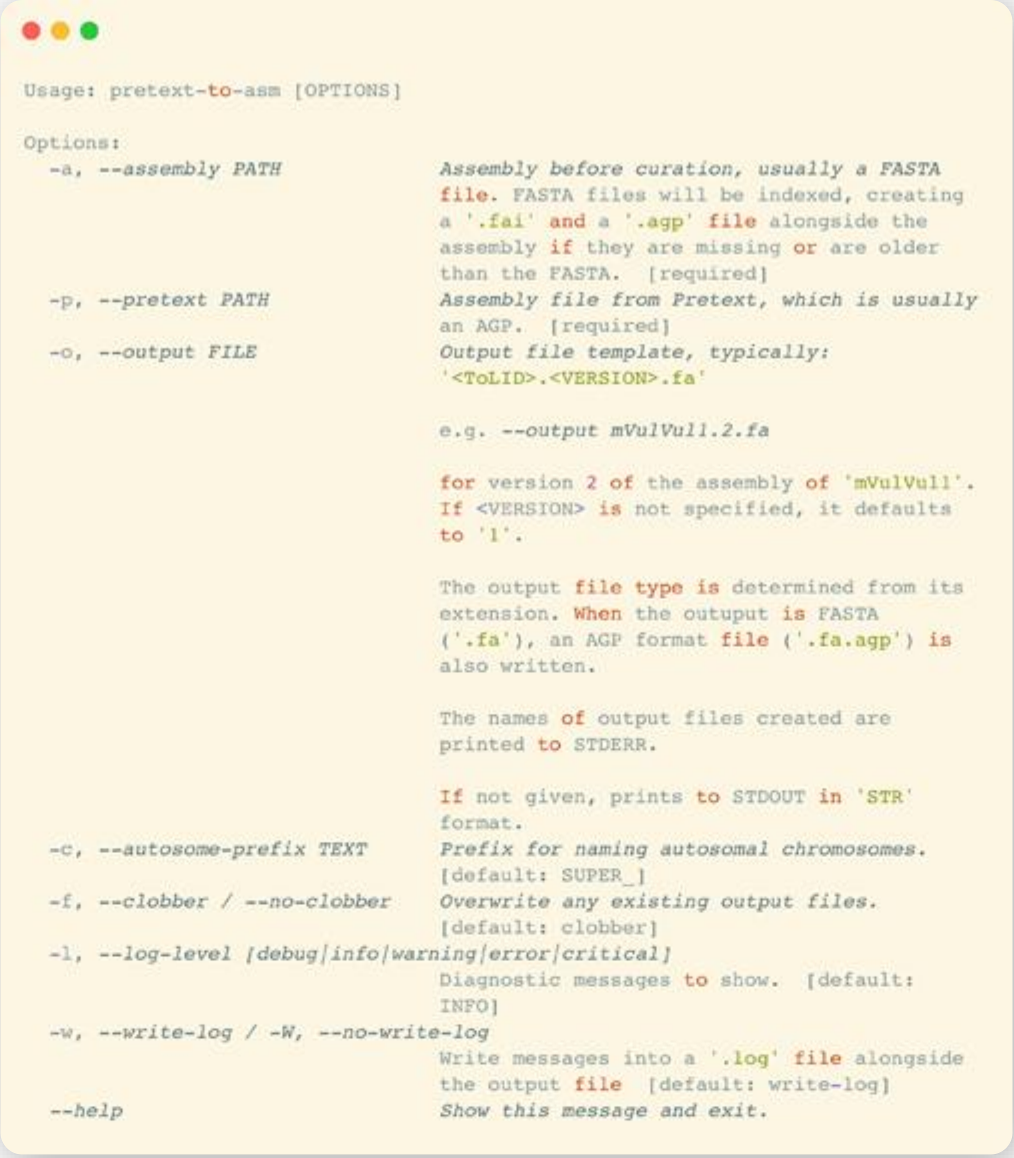
AGP generation

https://www.ncbi.nlm.nih.gov/genbank/genome_agp_specification/



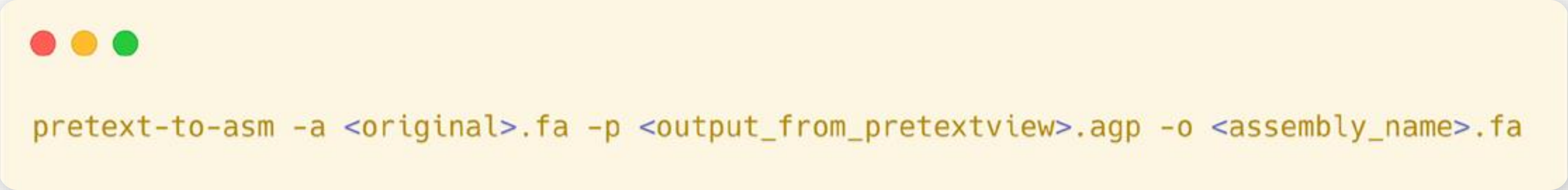
```
  GNU nano 6.2                                                              odGeoParv2_1_normal.pretext.agp_1
##agp-version   2.1
# DESCRIPTION: Generated by PretextView Version 0.2.5
# HiC MAP RESOLUTION: 3951.358154 bp/texel
Scaffold_1      1       15805   1       W       SCAFFOLD_8      1880847 1896651 +       Painted Haplotig
Scaffold_1      15806   15905   2       U       100     scaffold        yes     proximity_ligation
Scaffold_1      15906   122591  3       W       SCAFFOLD_34     1       106686  +       Painted
Scaffold_1      122592  122691  4       U       100     scaffold        yes     proximity_ligation
Scaffold_1      122692  3066453 5       W       SCAFFOLD_8      2117928 5061689 -       Painted
Scaffold_1      3066454 3066553 6       U       100     scaffold        yes     proximity_ligation
Scaffold_1      3066554 3141628 7       W       SCAFFOLD_43     1       75075   +       Painted
Scaffold_1      3141629 3141728 8       U       100     scaffold        yes     proximity_ligation
Scaffold_1      3141729 3363004 9       W       SCAFFOLD_8      1896652 2117927 -       Painted
Scaffold_1      3363005 3363104 10      U       100     scaffold        yes     proximity_ligation
Scaffold_1      3363105 4303527 11      W       SCAFFOLD_8      940424  1880846 +       Painted
Scaffold_1      4303528 4303627 12      U       100     scaffold        yes     proximity_ligation
Scaffold_1      4303628 6303014 13      W       SCAFFOLD_1      1       1999387 +       Painted
Scaffold_1      6303015 6303114 14      U       100     scaffold        yes     proximity_ligation
Scaffold_1      6303115 6322870 15      W       SCAFFOLD_83     1       19756   -       Painted
Scaffold_1      6322871 6322970 16      U       100     scaffold        yes     proximity_ligation
Scaffold_1      6322971 15047569        17      W       SCAFFOLD_1      1999388 10723986        +       Painted
Scaffold_2      1       2789658 1       W       SCAFFOLD_23     1       2789658 -       Painted
Scaffold_2      2789659 2789758 2       U       100     scaffold        yes     proximity_ligation
Scaffold_2      2789759 7349626 3       W       SCAFFOLD_1      13944343        18504210        +       Painted
Scaffold_3      1       7558948 1       W       SCAFFOLD_2      1       7558948 +       Painted
Scaffold_3      7558949 7559048 2       U       100     scaffold        yes     proximity_ligation
Scaffold_3      7559049 8037162 3       W       SCAFFOLD_2      7558949 8037062 -       Painted
```

Generating the curated fasta file
pretext-to-asm

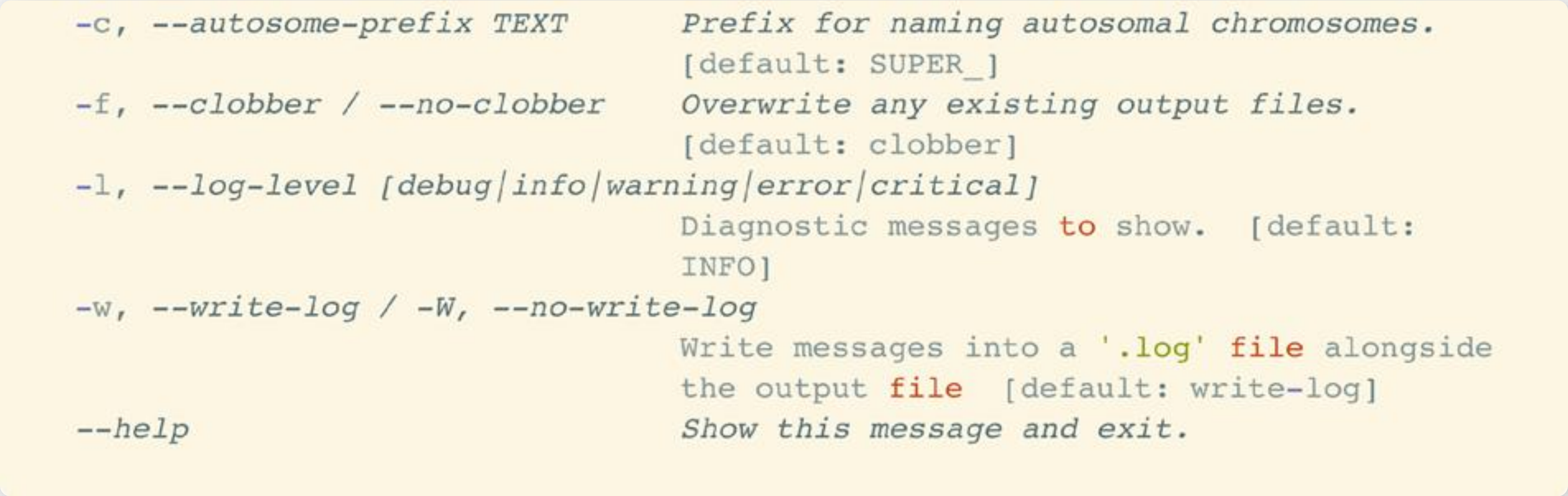


```
Usage: pretext-to-asm [OPTIONS]

Options:
  -a, --assembly PATH             Assembly before curation, usually a FASTA
                                  file. FASTA files will be indexed, creating
                                  a '.fai' and a '.agp' file alongside the
                                  assembly if they are missing or are older
                                  than the FASTA.  [required]
  -p, --pretext PATH              Assembly file from Pretext, which is usually
                                  an AGP.  [required]
  -o, --output FILE               Output file template, typically:
                                  '<ToLID>.<VERSION>.fa'

                                  e.g. --output mVulVul1.2.fa

                                  for version 2 of the assembly of 'mVulVul1'.
                                  If <VERSION> is not specified, it defaults
                                  to '1'.

                                  The output file type is determined from its
                                  extension. When the outupt is FASTA
                                  ('.fa'), an AGP format file ('.fa.agp') is
                                  also written.

                                  The names of output files created are
                                  printed to STDERR.

                                  If not given, prints to STDOUT in 'STR'
                                  format.
  -c, --autosome-prefix TEXT      Prefix for naming autosomal chromosomes.
                                  [default: SUPER_]
  -f, --clobber / --no-clobber    Overwrite any existing output files.
                                  [default: clobber]
  -l, --log-level [debug|info|warning|error|critical]
                                  Diagnostic messages to show.  [default:
                                  INFO]
  -w, --write-log / -W, --no-write-log
                                  Write messages into a '.log' file alongside
                                  the output file  [default: write-log]
  --help                          Show this message and exit.
```

Generating the curated fasta file
pretext-to-asm

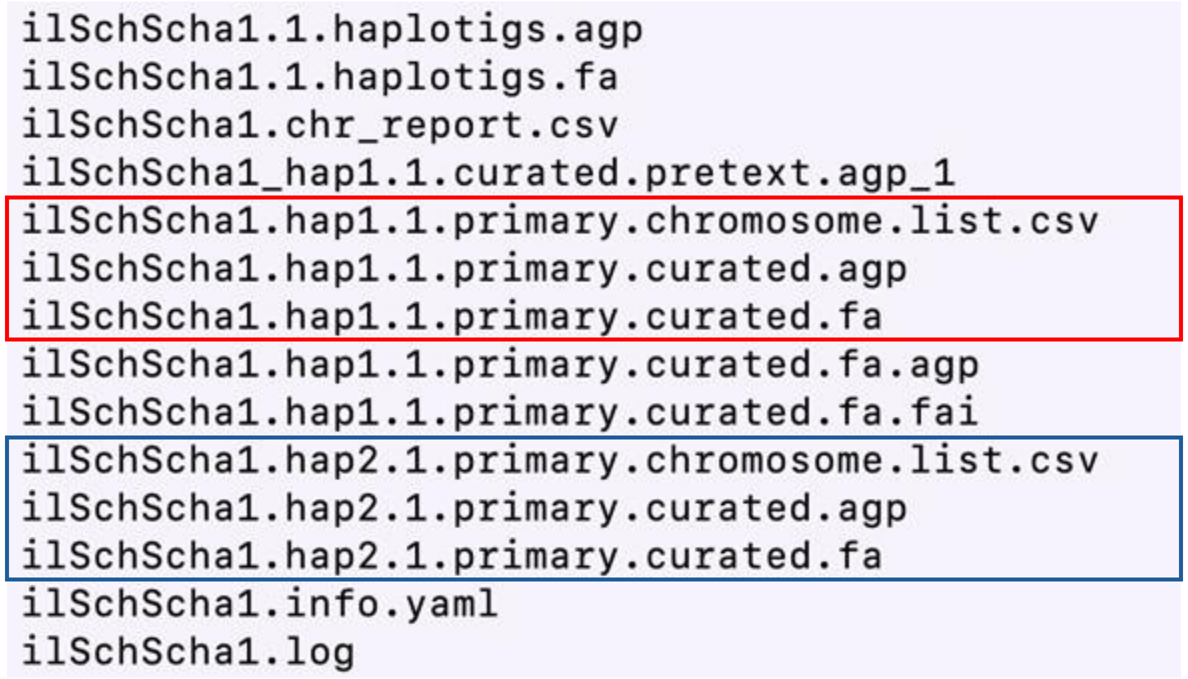


```
pretext-to-asm -a <original>.fa -p <output_from_pretextview>.agp -o <assembly_name>.fa
```



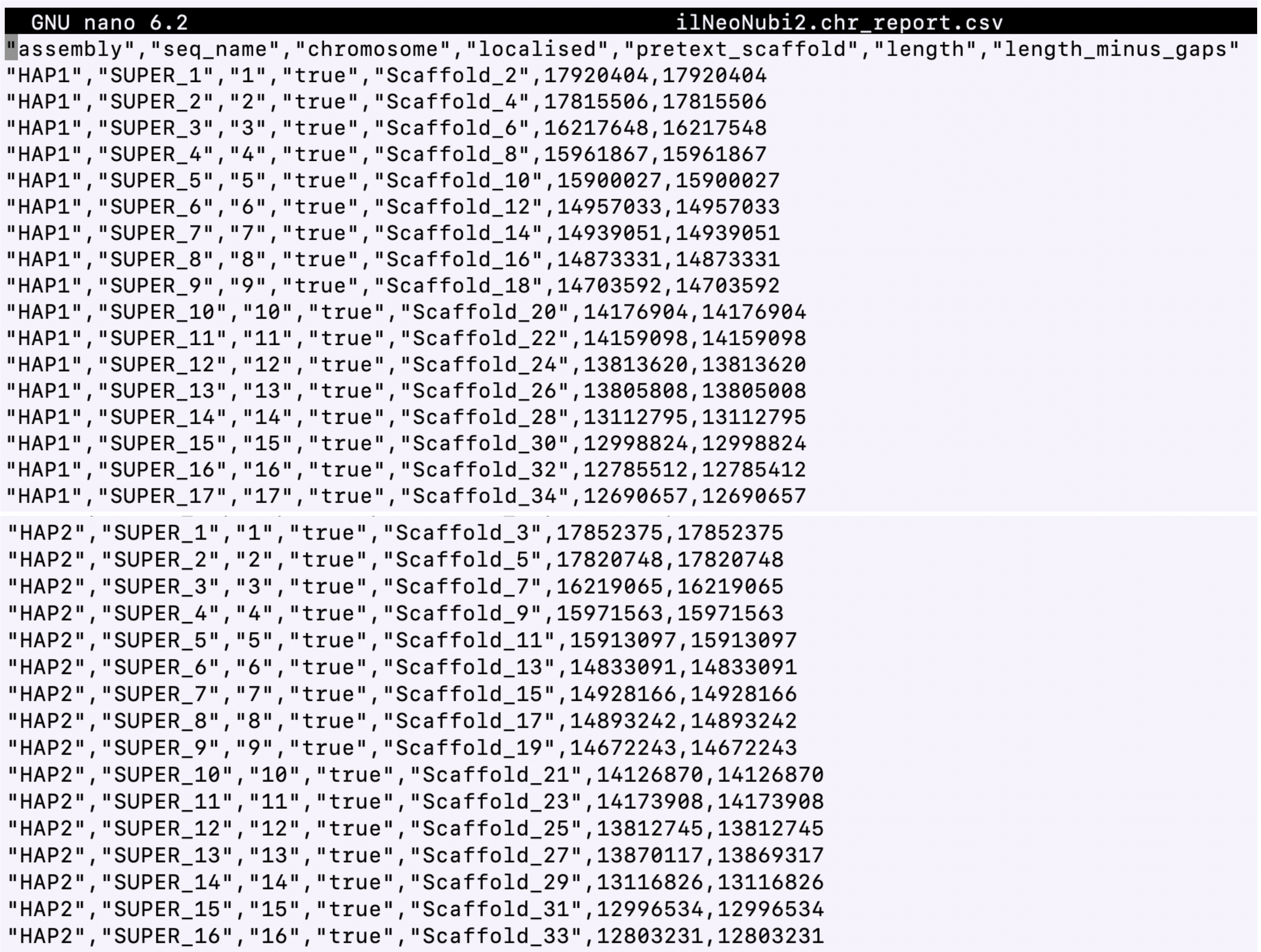
```
  -c, --autosome-prefix TEXT      Prefix for naming autosomal chromosomes.
                                  [default: SUPER_]
  -f, --clobber / --no-clobber    Overwrite any existing output files.
                                  [default: clobber]
  -l, --log-level [debug|info|warning|error|critical]
                                  Diagnostic messages to show.  [default:
                                  INFO]
  -w, --write-log / -W, --no-write-log
                                  Write messages into a '.log' file alongside
                                  the output file  [default: write-log]
  --help                          Show this message and exit.
```

Pretext-to-asm output files



```
ilSchScha1.1.haplotigs.agp
ilSchScha1.1.haplotigs.fa
ilSchScha1.chr_report.csv
ilSchScha1_hap1.1.curated.pretext.agp_1
ilSchScha1.hap1.1.primary.chromosome.list.csv
ilSchScha1.hap1.1.primary.curated.agp
ilSchScha1.hap1.1.primary.curated.fa
ilSchScha1.hap1.1.primary.curated.fa.agp
ilSchScha1.hap1.1.primary.curated.fa.fai
ilSchScha1.hap2.1.primary.chromosome.list.csv
ilSchScha1.hap2.1.primary.curated.agp
ilSchScha1.hap2.1.primary.curated.fa
ilSchScha1.info.yaml
ilSchScha1.log
```

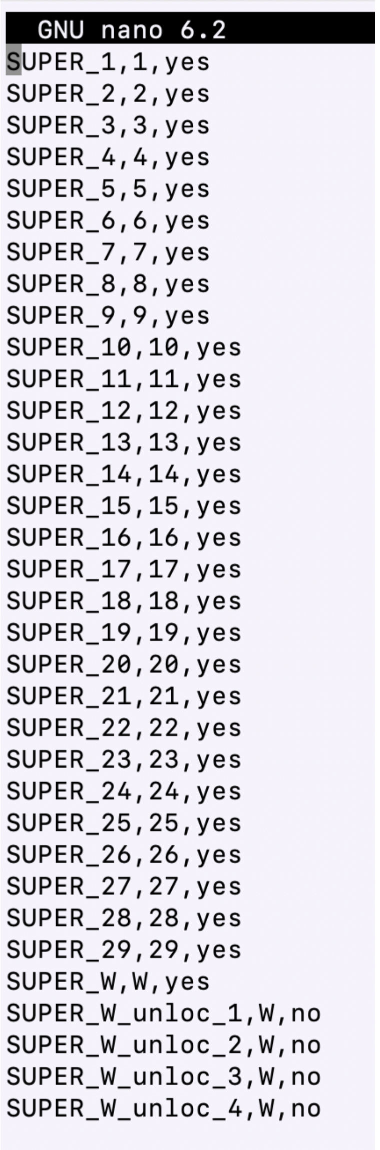
Pretext-to-asm output files



```
  GNU nano 6.2                                          ilNeoNubi2.chr_report.csv
"assembly","seq_name","chromosome","localised","pretext_scaffold","length","length_minus_gaps"
"HAP1","SUPER_1","1","true","Scaffold_2",17920404,17920404
"HAP1","SUPER_2","2","true","Scaffold_4",17815506,17815506
"HAP1","SUPER_3","3","true","Scaffold_6",16217648,16217548
"HAP1","SUPER_4","4","true","Scaffold_8",15961867,15961867
"HAP1","SUPER_5","5","true","Scaffold_10",15900027,15900027
"HAP1","SUPER_6","6","true","Scaffold_12",14957033,14957033
"HAP1","SUPER_7","7","true","Scaffold_14",14939051,14939051
"HAP1","SUPER_8","8","true","Scaffold_16",14873331,14873331
"HAP1","SUPER_9","9","true","Scaffold_18",14703592,14703592
"HAP1","SUPER_10","10","true","Scaffold_20",14176904,14176904
"HAP1","SUPER_11","11","true","Scaffold_22",14159098,14159098
"HAP1","SUPER_12","12","true","Scaffold_24",13813620,13813620
"HAP1","SUPER_13","13","true","Scaffold_26",13805808,13805008
"HAP1","SUPER_14","14","true","Scaffold_28",13112795,13112795
"HAP1","SUPER_15","15","true","Scaffold_30",12998824,12998824
"HAP1","SUPER_16","16","true","Scaffold_32",12785512,12785412
"HAP1","SUPER_17","17","true","Scaffold_34",12690657,12690657
```

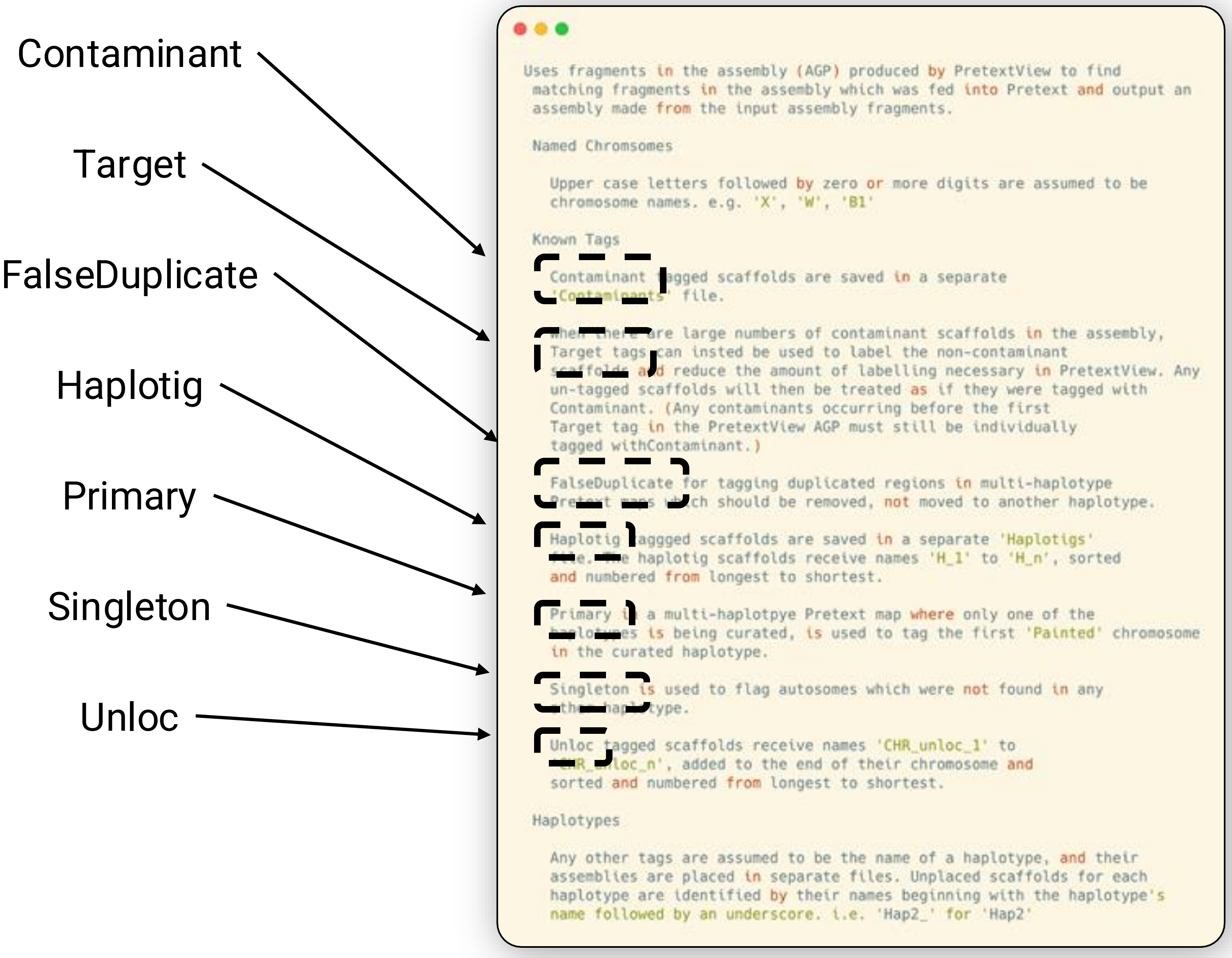
```
"HAP2","SUPER_1","1","true","Scaffold_3",17852375,17852375
"HAP2","SUPER_2","2","true","Scaffold_5",17820748,17820748
"HAP2","SUPER_3","3","true","Scaffold_7",16219065,16219065
"HAP2","SUPER_4","4","true","Scaffold_9",15971563,15971563
"HAP2","SUPER_5","5","true","Scaffold_11",15913097,15913097
"HAP2","SUPER_6","6","true","Scaffold_13",14833091,14833091
"HAP2","SUPER_7","7","true","Scaffold_15",14928166,14928166
"HAP2","SUPER_8","8","true","Scaffold_17",14893242,14893242
"HAP2","SUPER_9","9","true","Scaffold_19",14672243,14672243
"HAP2","SUPER_10","10","true","Scaffold_21",14126870,14126870
"HAP2","SUPER_11","11","true","Scaffold_23",14173908,14173908
"HAP2","SUPER_12","12","true","Scaffold_25",13812745,13812745
"HAP2","SUPER_13","13","true","Scaffold_27",13870117,13869317
"HAP2","SUPER_14","14","true","Scaffold_29",13116826,13116826
"HAP2","SUPER_15","15","true","Scaffold_31",12996534,12996534
"HAP2","SUPER_16","16","true","Scaffold_33",12803231,12803231
```

Chromosome list file



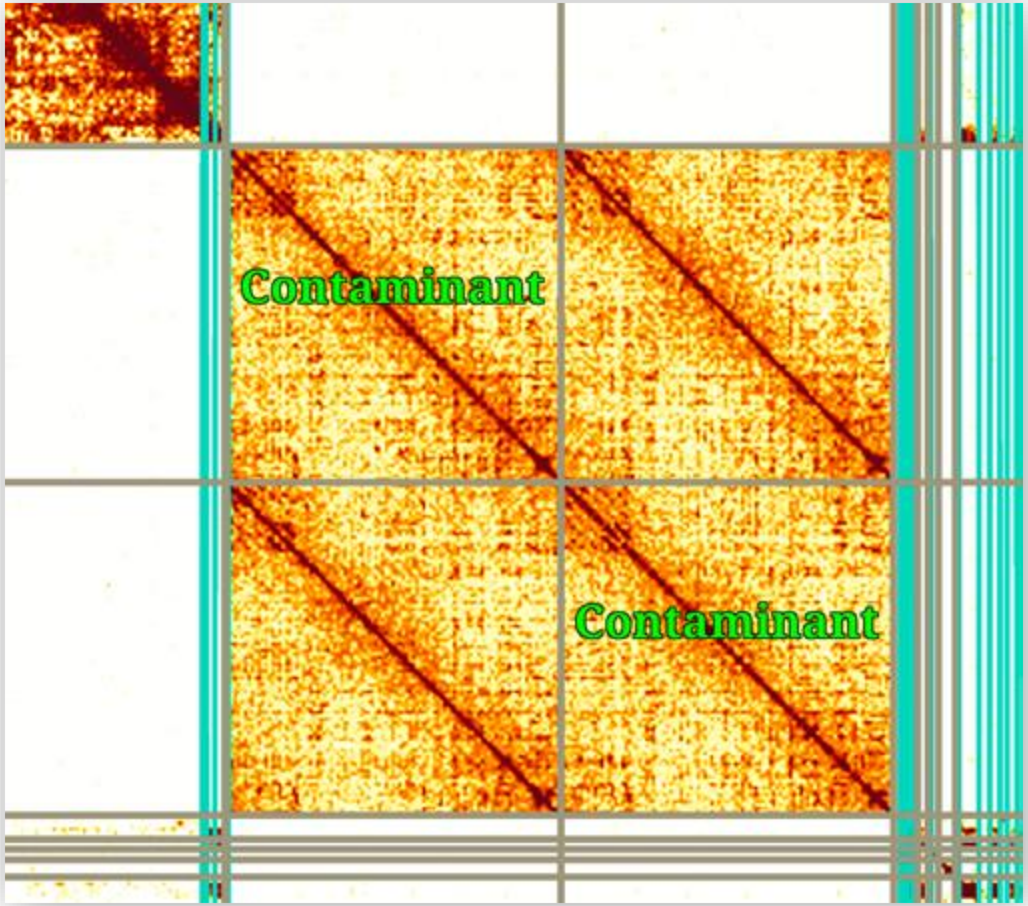
```
  GNU nano 6.2
SUPER_1,1,yes
SUPER_2,2,yes
SUPER_3,3,yes
SUPER_4,4,yes
SUPER_5,5,yes
SUPER_6,6,yes
SUPER_7,7,yes
SUPER_8,8,yes
SUPER_9,9,yes
SUPER_10,10,yes
SUPER_11,11,yes
SUPER_12,12,yes
SUPER_13,13,yes
SUPER_14,14,yes
SUPER_15,15,yes
SUPER_16,16,yes
SUPER_17,17,yes
SUPER_18,18,yes
SUPER_19,19,yes
SUPER_20,20,yes
SUPER_21,21,yes
SUPER_22,22,yes
SUPER_23,23,yes
SUPER_24,24,yes
SUPER_25,25,yes
SUPER_26,26,yes
SUPER_27,27,yes
SUPER_28,28,yes
SUPER_29,29,yes
SUPER_W,W,yes
SUPER_W_unloc_1,W,no
SUPER_W_unloc_2,W,no
SUPER_W_unloc_3,W,no
SUPER_W_unloc_4,W,no
```

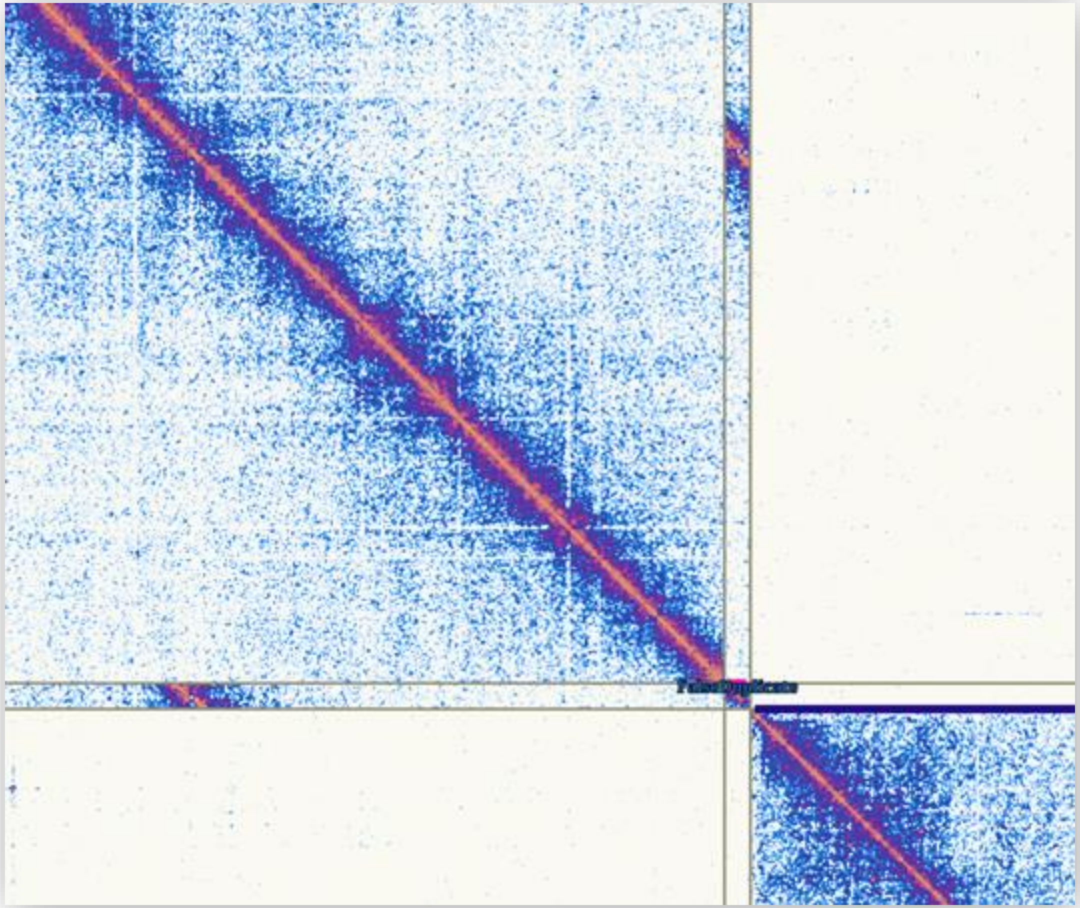
What pretext-to-asm does



```
Uses fragments in the assembly (AGP) produced by PretextView to find
 matching fragments in the assembly which was fed into Pretext and output an
 assembly made from the input assembly fragments.

 Named Chromsomes

   Upper case letters followed by zero or more digits are assumed to be
   chromosome names. e.g. 'X', 'W', 'B1'

 Known Tags

   Contaminant tagged scaffolds are saved in a separate
   'Contaminants' file.

   When there are large numbers of contaminant scaffolds in the assembly,
   Target tags can insted be used to label the non-contaminant
   scaffolds and reduce the amount of labelling necessary in PretextView. Any
   un-tagged scaffolds will then be treated as if they were tagged with
   Contaminant. (Any contaminants occurring before the first
   Target tag in the PretextView AGP must still be individually
   tagged withContaminant.)

   FalseDuplicate for tagging duplicated regions in multi-haplotype
   Pretext maps which should be removed, not moved to another haplotype.

   Haplotig taggged scaffolds are saved in a separate 'Haplotigs'
   file. The haplotig scaffolds receive names 'H_1' to 'H_n', sorted
   and numbered from longest to shortest.

   Primary in a multi-haplotpye Pretext map where only one of the
   haplotypes is being curated, is used to tag the first 'Painted' chromosome
   in the curated haplotype.

   Singleton is used to flag autosomes which were not found in any
   other haplotype.

   Unloc tagged scaffolds receive names 'CHR_unloc_1' to
   'CHR_unloc_n', added to the end of their chromosome and
   sorted and numbered from longest to shortest.

 Haplotypes

   Any other tags are assumed to be the name of a haplotype, and their
   assemblies are placed in separate files. Unplaced scaffolds for each
   haplotype are identified by their names beginning with the haplotype's
   name followed by an underscore. i.e. 'Hap2_' for 'Hap2'
```

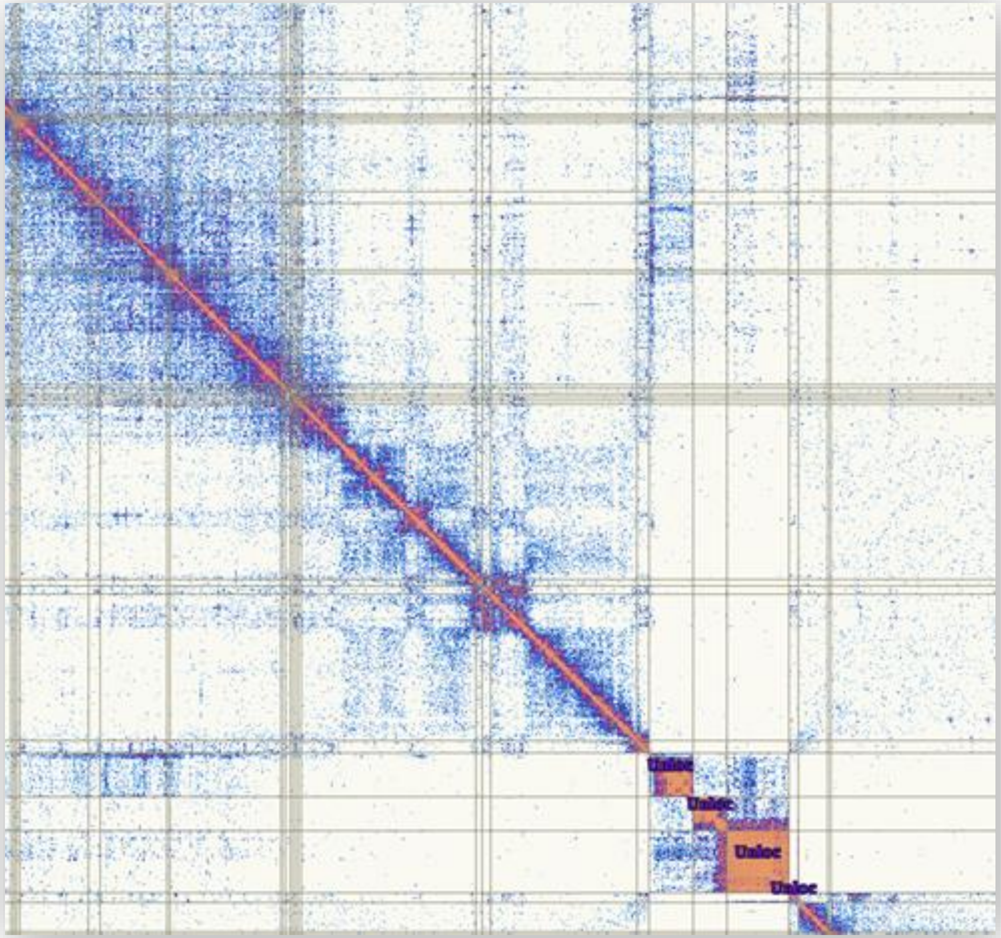
What pretext-to-asm does

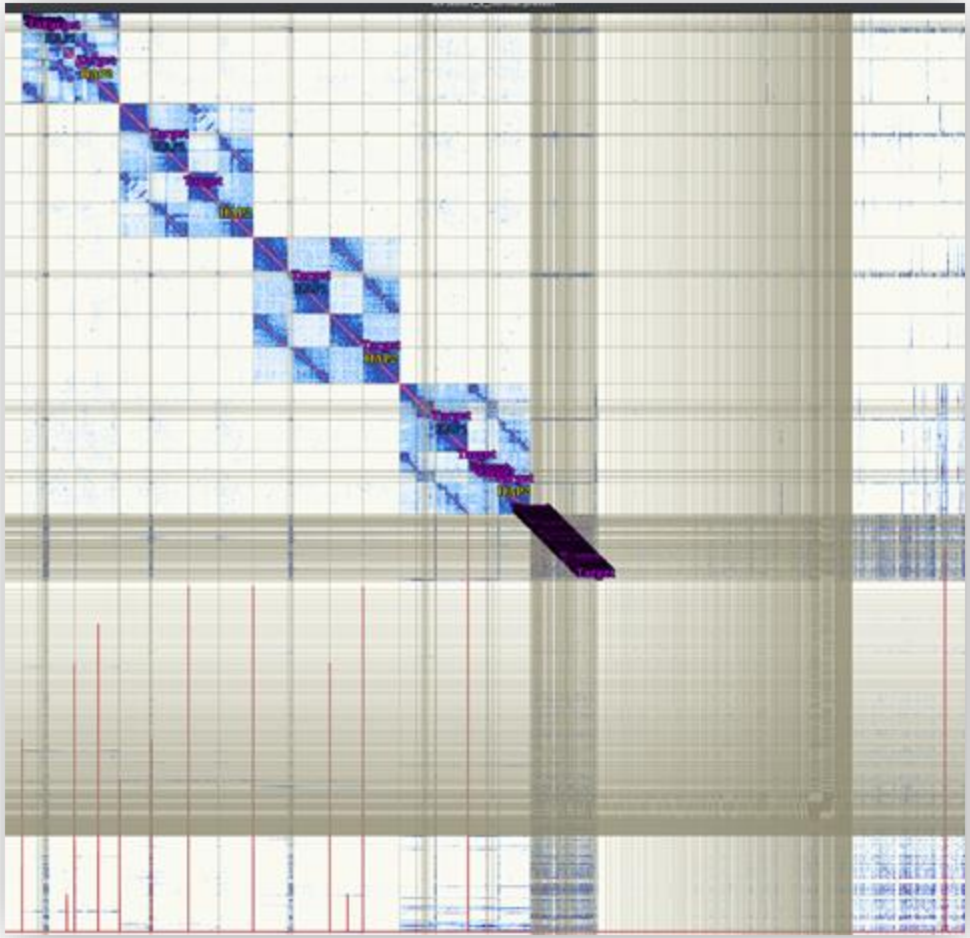




Combined maps
Uneven coverage

What pretext-to-asm does



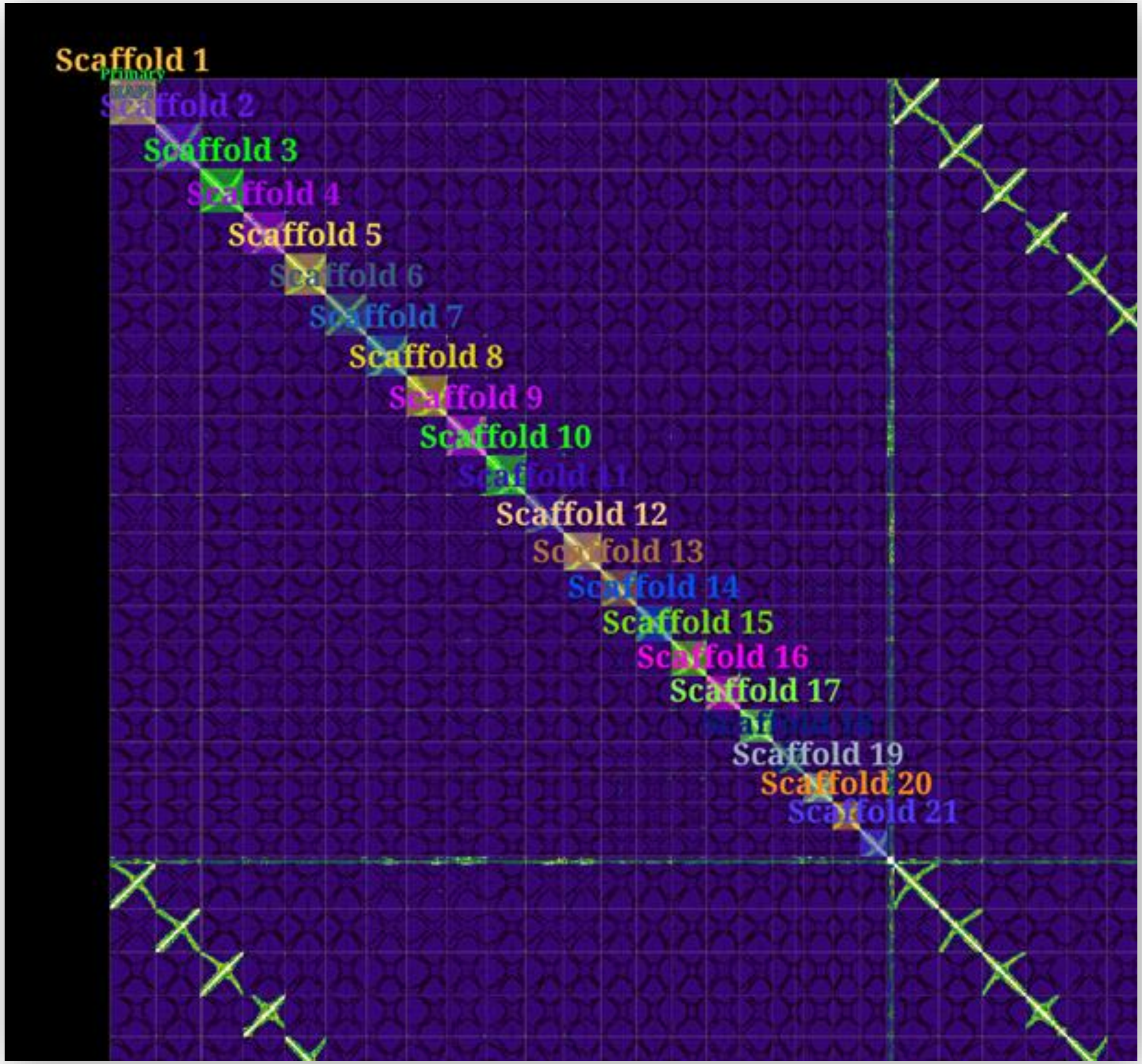


‘Unloc’ tag

‘Target’ tag

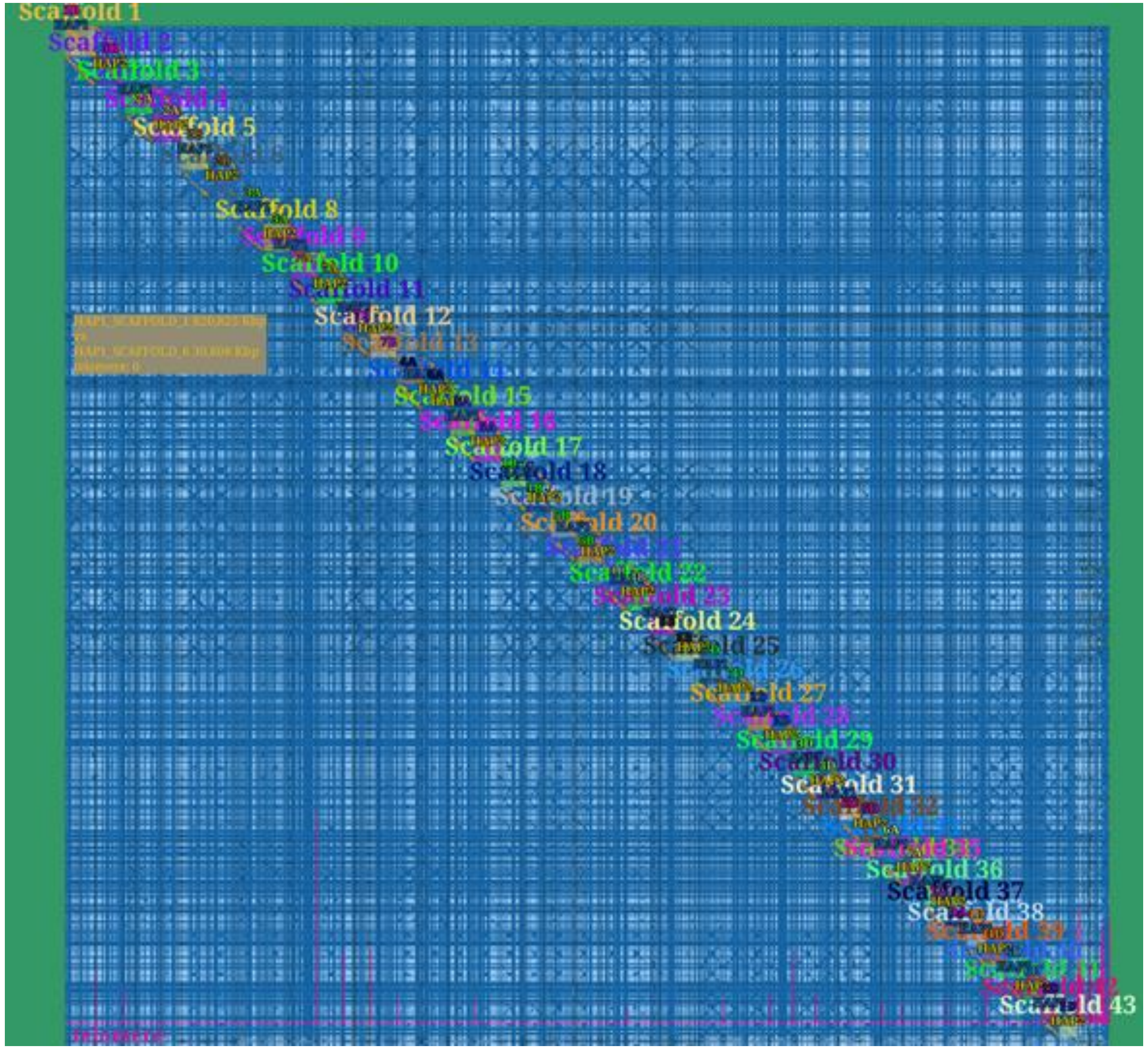
What pretext-to-asm does

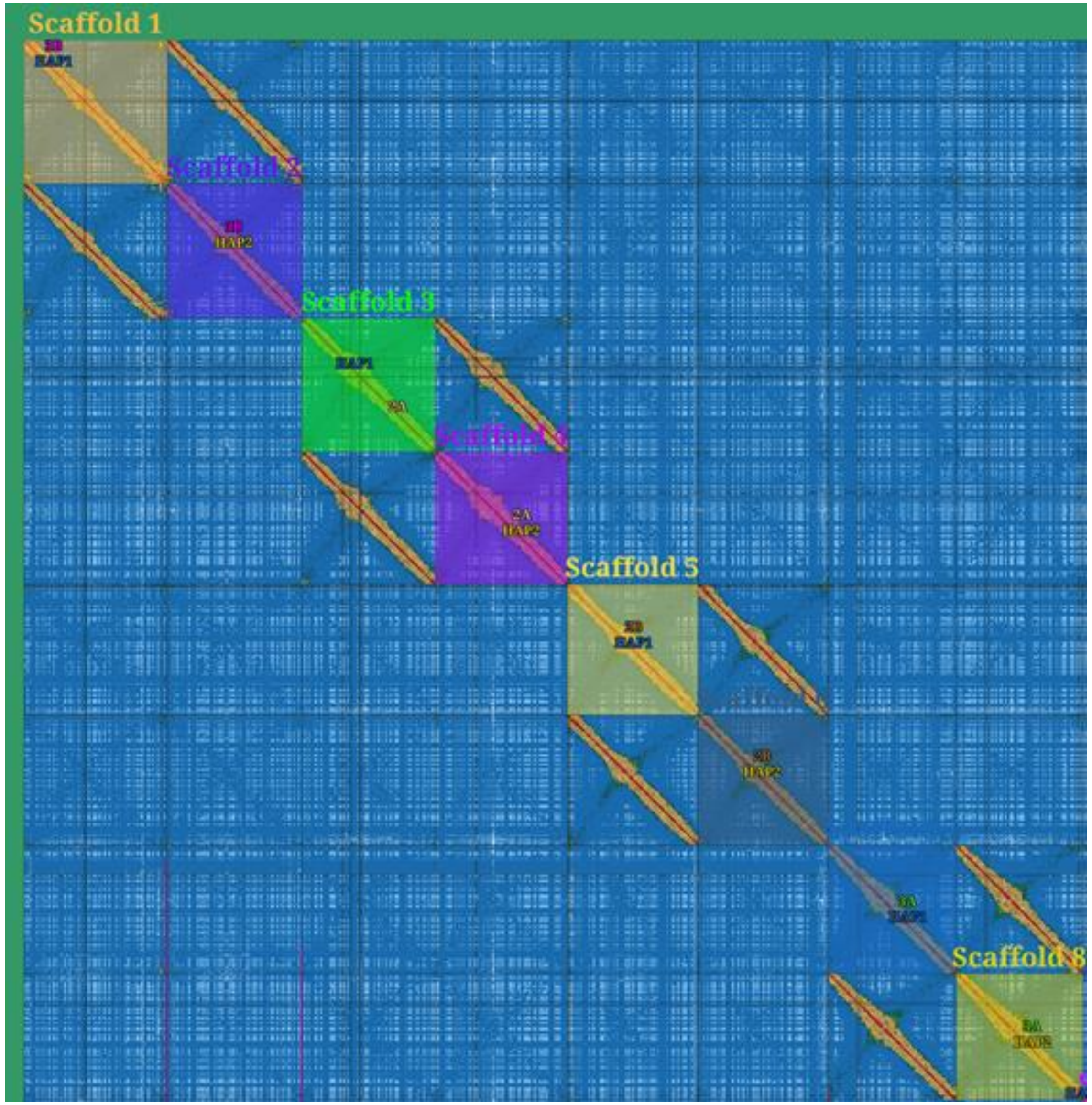
‘Primary’ tag



What pretext-to-asm does

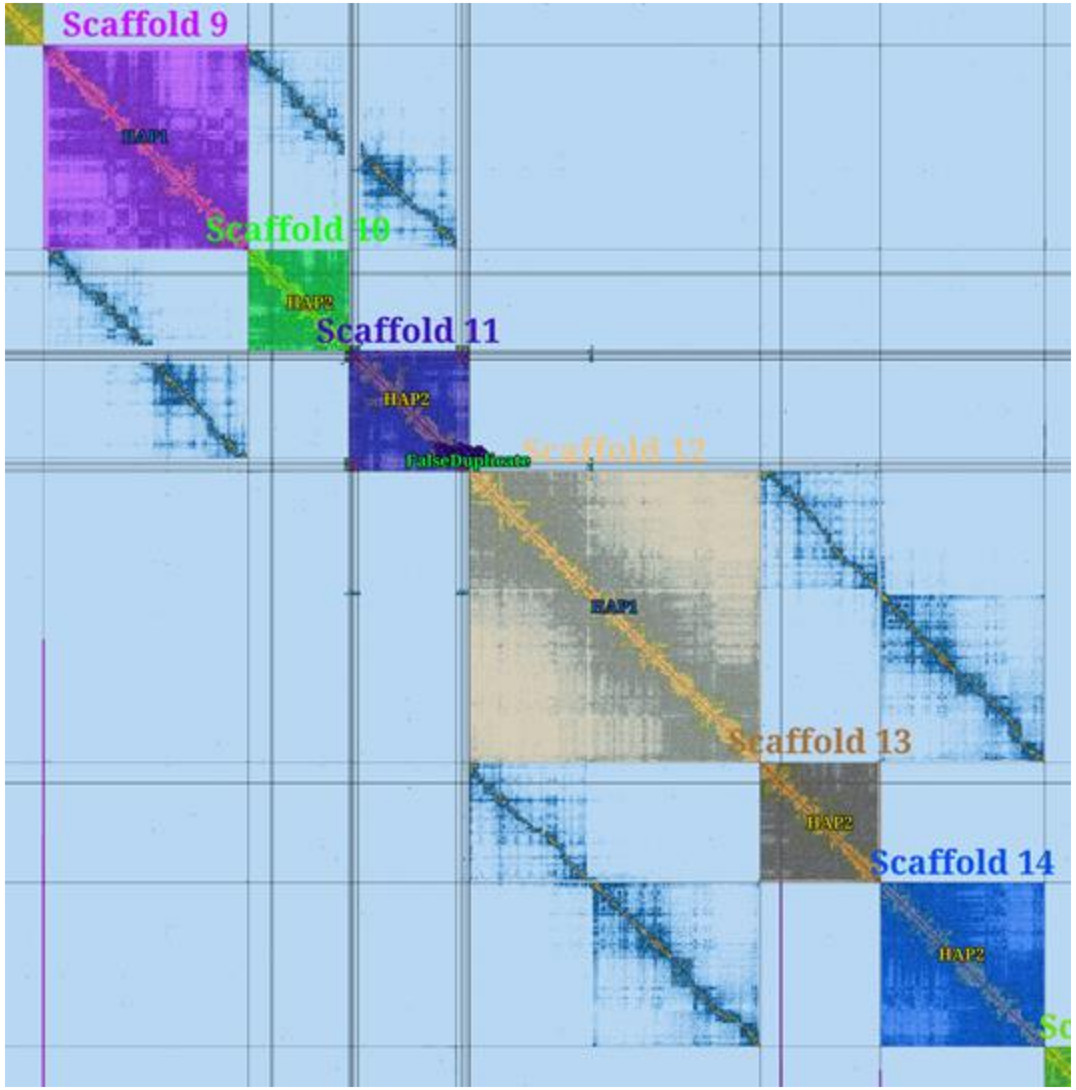
Renaming after a reference



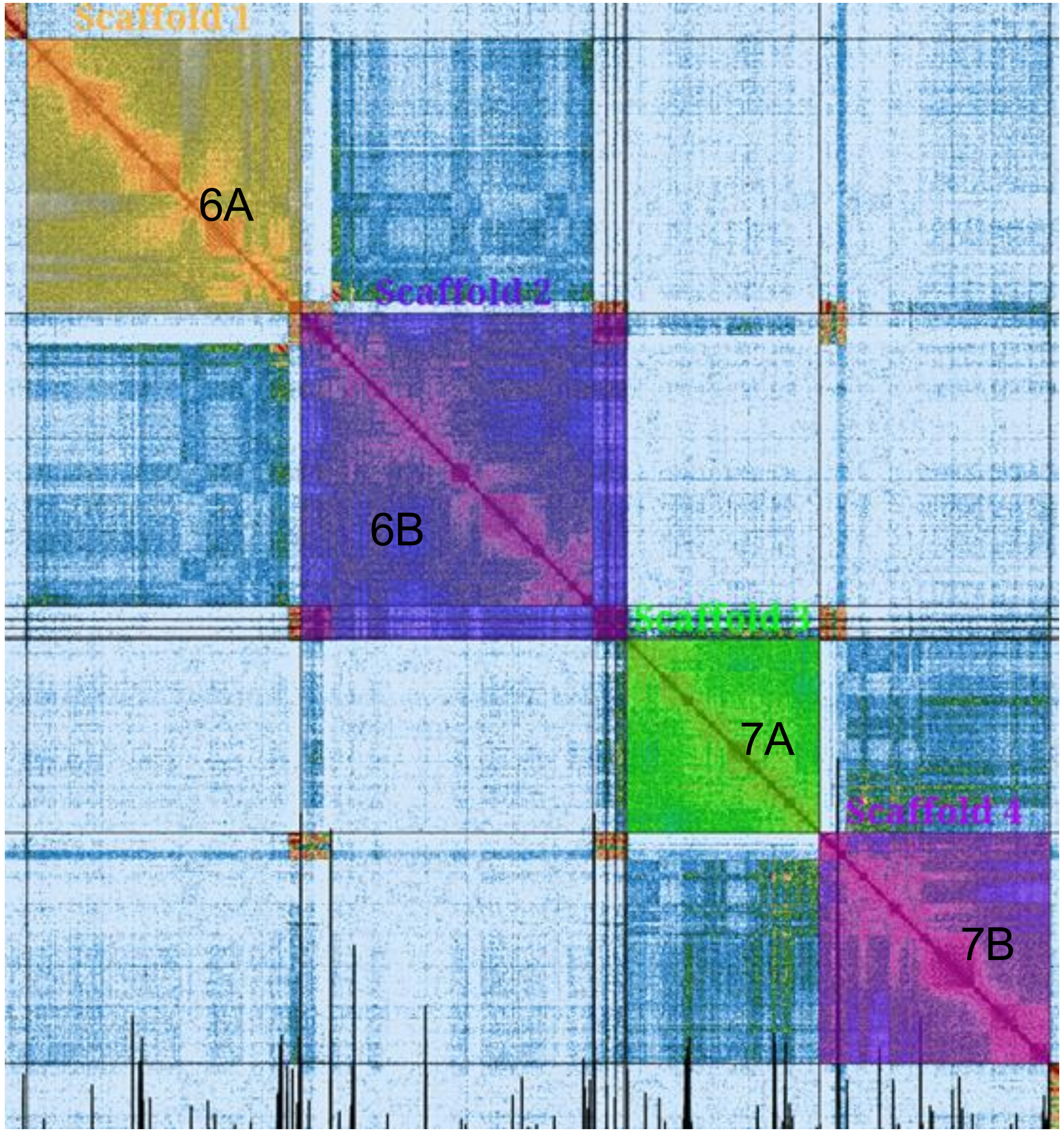


What pretext-to-asm does

Dealing with fusions/fissions



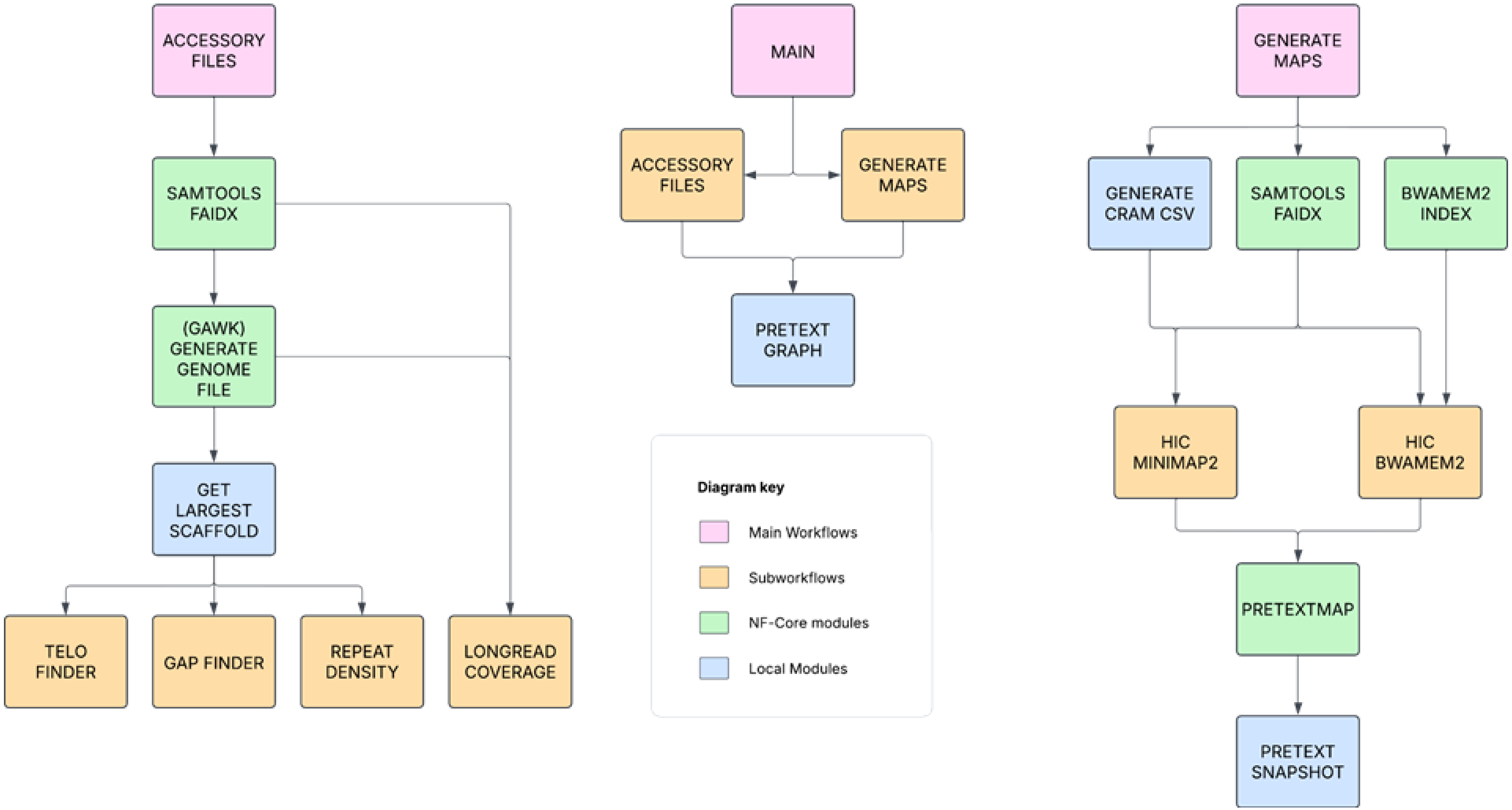
Fissioned chroms in HAP2 file



HAP2 is renamed after HAP1

CurationPretext NextFlow Pipeline

https://pipelines.tol.sanger.ac.uk/curationpretext



ACCESSORY FILES

SAMTOOLS FAIDX

(GAWK) GENERATE GENOME FILE

GET LARGEST SCAFFOLD

TELO FINDER

GAP FINDER

REPEAT DENSITY

LONGREAD COVERAGE

MAIN

ACCESSORY FILES

GENERATE MAPS

PRETEXT GRAPH

Diagram key

- 🟪 Main Workflows
- 🟧 Subworkflows
- 🟩 NF-Core modules
- 🟦 Local Modules

GENERATE MAPS

GENERATE CRAM CSV

SAMTOOLS FAIDX

BWAMEM2 INDEX

HIC MINIMAP2

HIC BWAMEM2

PRETEXTMAP

PRETEXT SNAPSHOT

CurationPretext NextFlow Pipeline

https://pipelines.tol.sanger.ac.uk/curationpretext

```
nextflow run sanger-tol/curationpretext \
  --input { input.fasta } \
  --cram { path/to/hic/cram/ } \
  --reads { path/to/longread/fasta/ } \
  --read_type { default is "hifi" }
  --sample { default is "pretext_rerun" } \
  --teloseq { default is "TTAGGG" } \
  --map_order { default is "unsorted" } \
  --multi_mapping { default is "0" (for no mapping)} \
  --all_output <true/false> \
  --outdir { OUTDIR } \
  -profile <docker/singularity/{institute}>
```

Hands-on

https://github.com/epaule/Physalia-Manual-Genome-Curation/blob/main/Session3.2.md