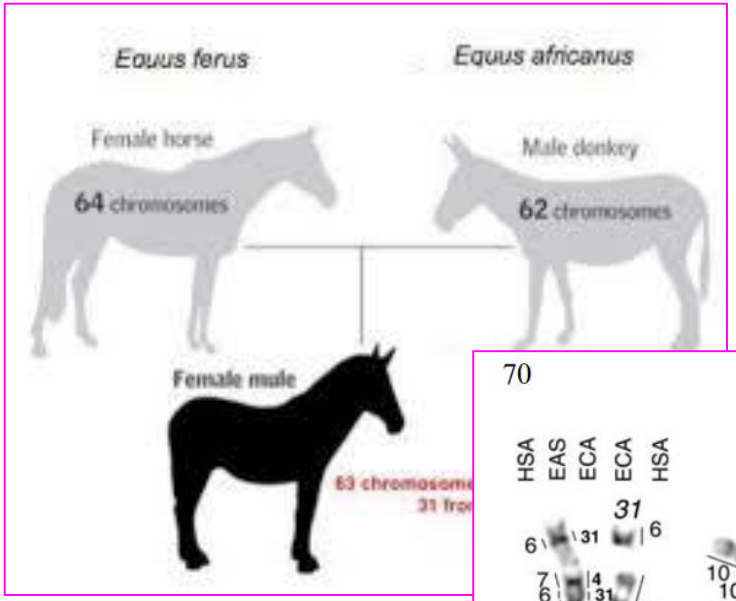
AncesBin – A Pipeline to Bin 10X, HiC, PacBio and ONT reads Based on Ancestry Assemblies

Zemin Ning

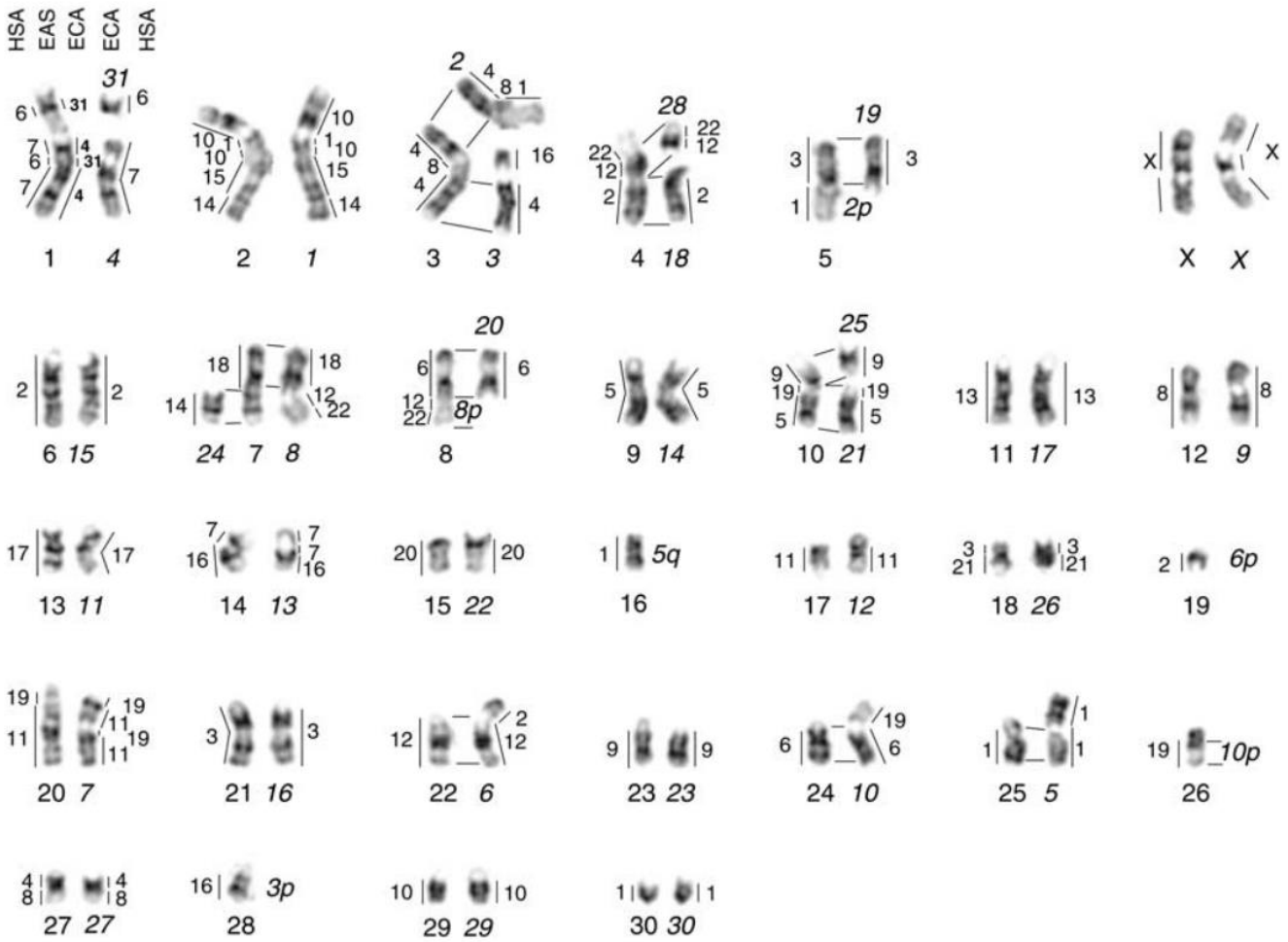
The Wellcome Sanger Institute

UK

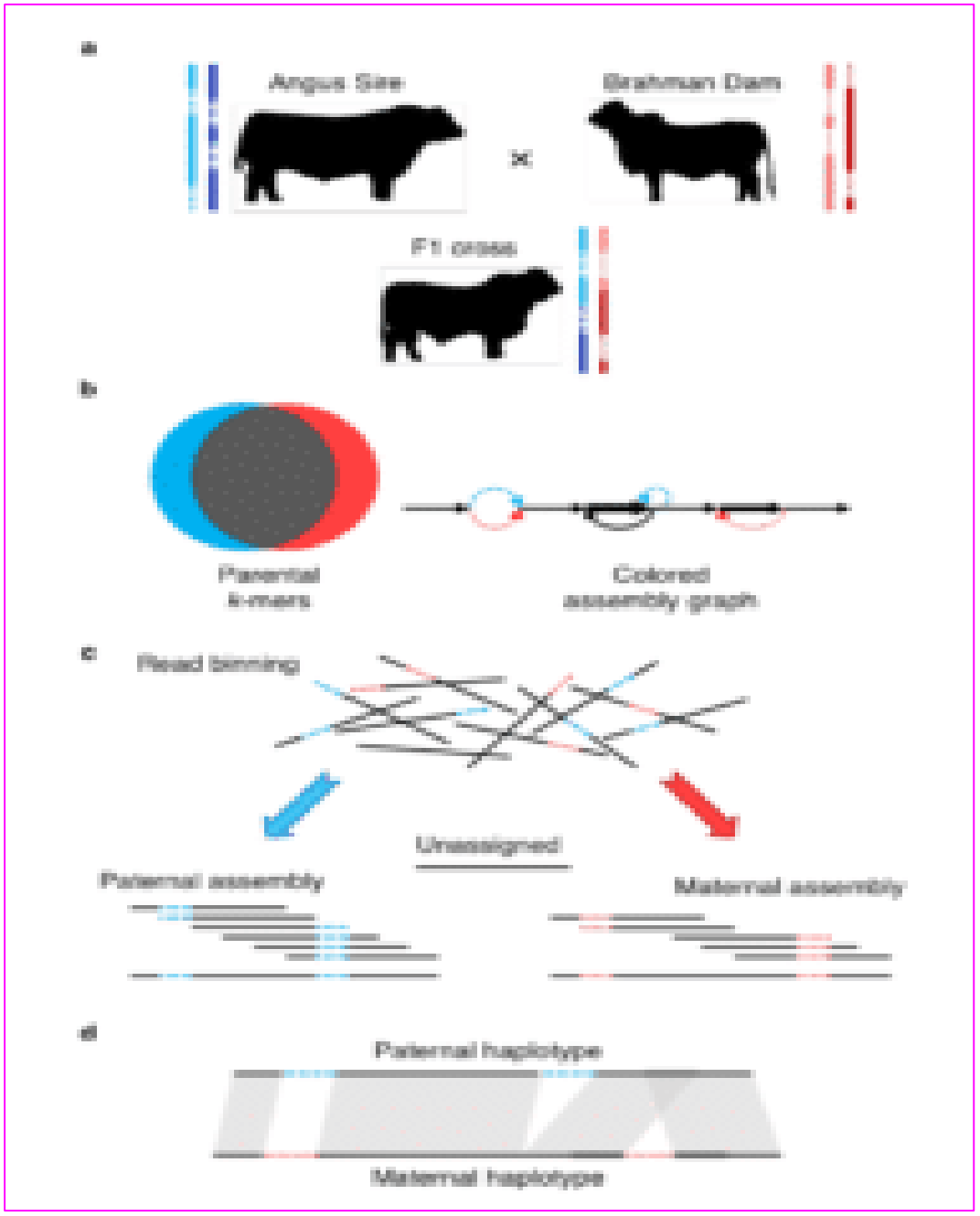
wellcome trust sanger institute



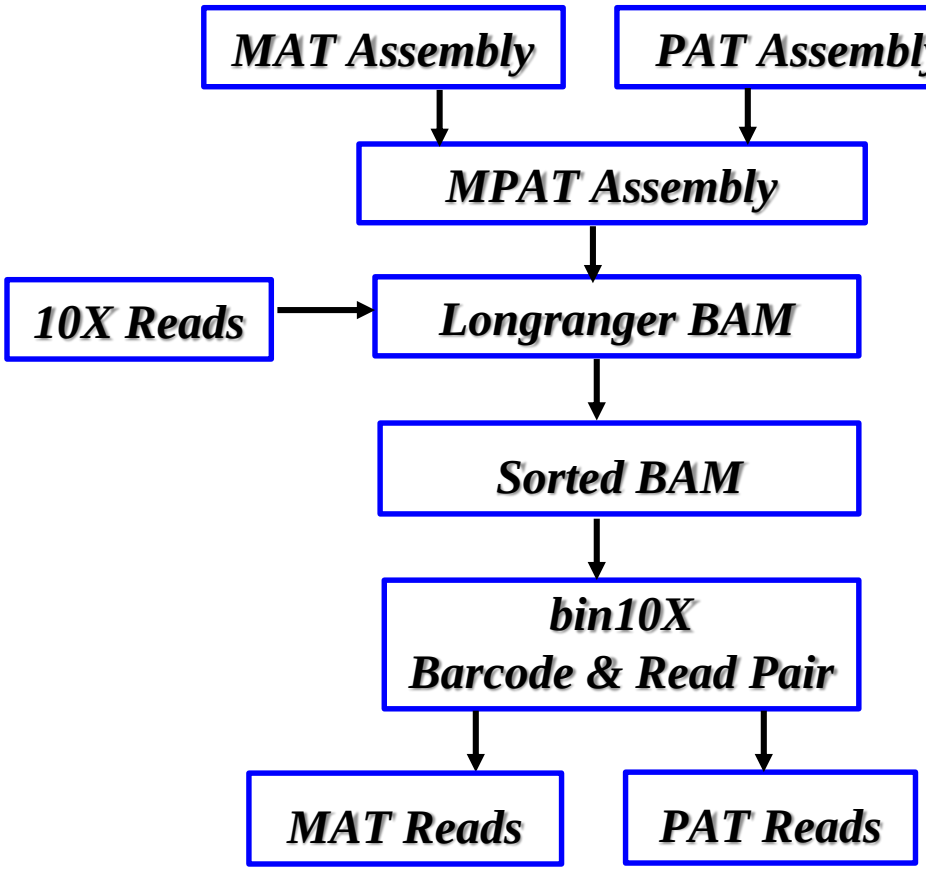
Mule Genome and Chromosome Painting

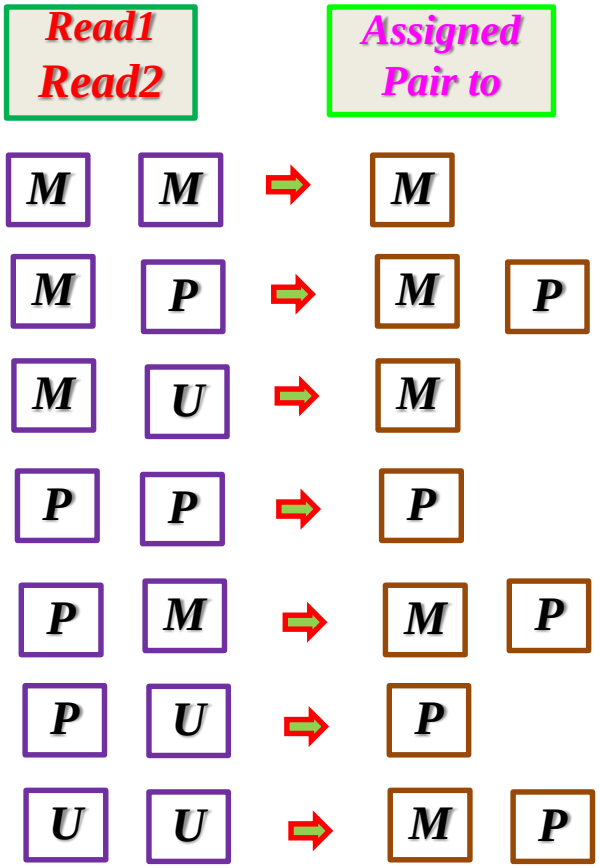


Cattle Genome with triobinning



Flowchart of bin10X

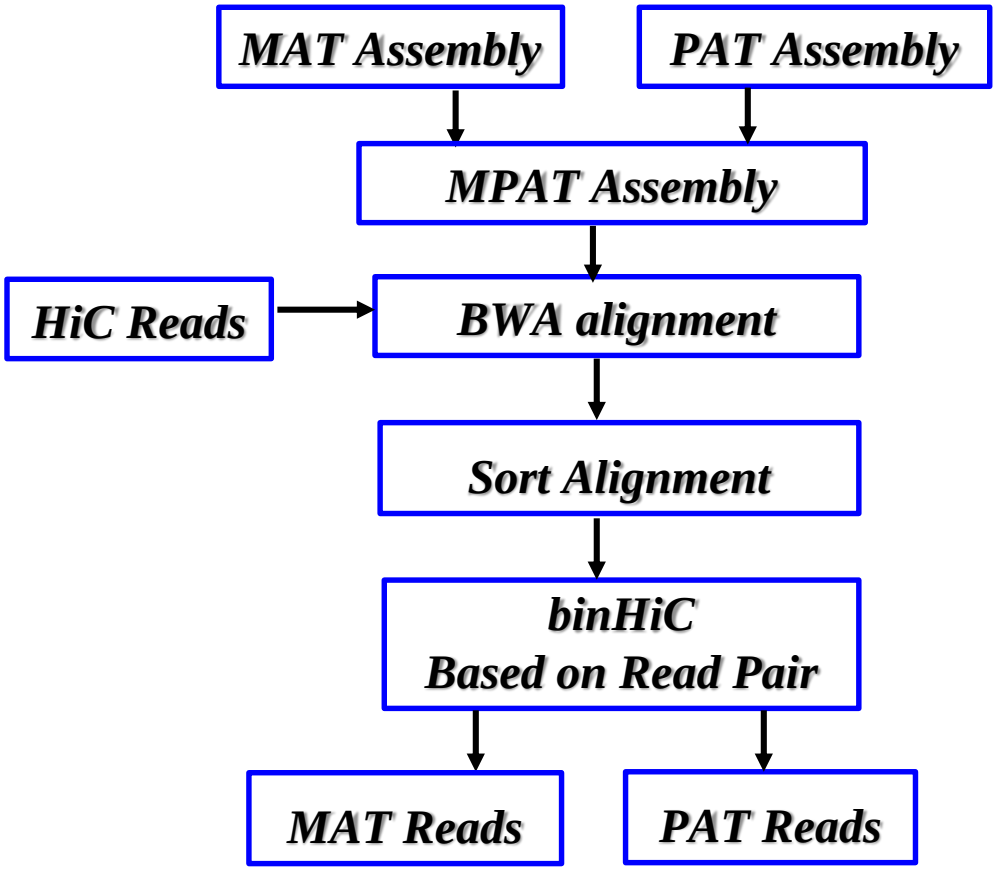




How to run bin10X

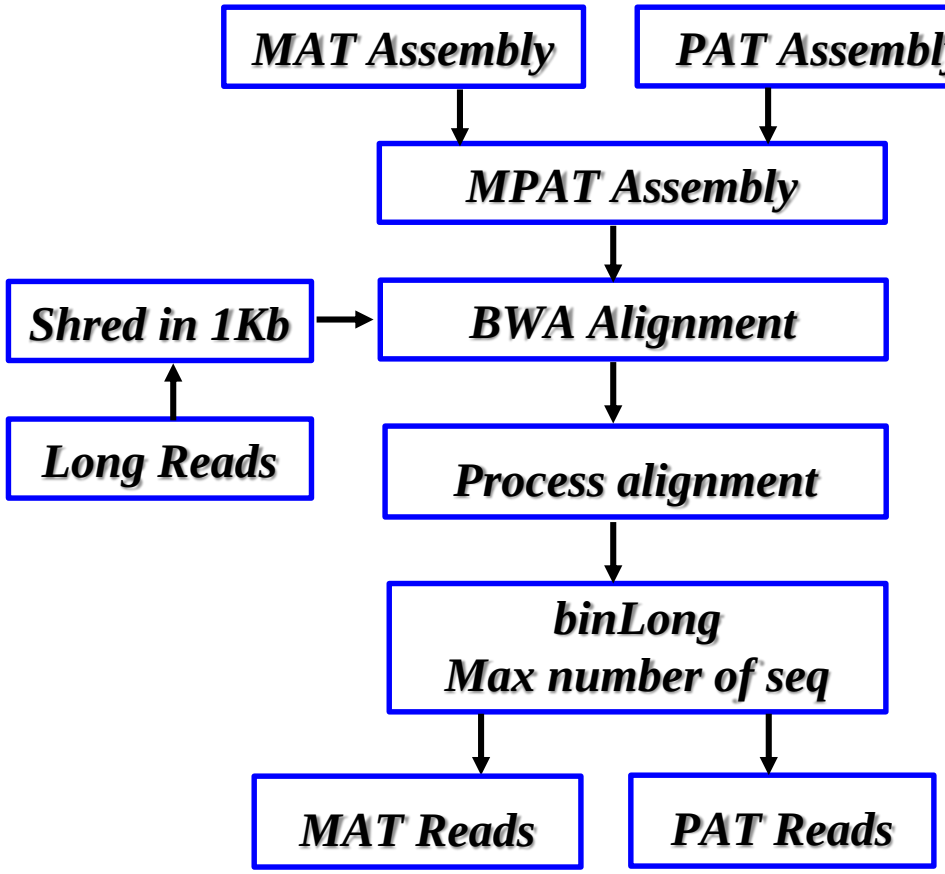
```
#### Run bin10x:
          $ /full/path/to/AncesBin/src/bin10x -nodes <nodes> Input_sorted_bam Input_data_file Output_directory \

          Parameters:
            nodes:            number of CPUs requested  [ default = 30 ]
            Input_sorted_bam: read name sorted longranger BAM file
                              input a bam file which had been produced by using lariat in longranger,
                        (a). rename the assembly file:
                         $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name MAT MAT_assembly.fa MAT_ref.fasta
                         $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name PAT PAT_assembly.fa PAT_ref.fasta

                        (b). cat assemblies
                         $ cat MAT_ref.fasta PAT_ref.fasta > MPAT_ref.fasta

                        (b). generate reference assembly file using longranger
                         $ longranger mkref MPAT_ref.fasta

                        (c). align 10x reads using lariat longranger
                         $ longranger align --fastq="reads_10x" --sample=fTakRub1 --reference="refdata-MPAT_ref" --localcores=50 --id=10x-align
                         Note: for reads_10x please provide full path
                               10x-align is an output directory

                        (d). sort the longranger bam
                         $ samtools sort -n -@ 30 -O BAM -o possorted_sort.bam possorted_bam.bam

            Input_data_file:  a text file to point the locations of the reads in paired files\n");

q1=/lustre/scratch116/vr/projects/Tes1_S1_L008_R1_001.fastq.gz \
q2=/lustre/scratch116/vr/projects/Tes1_S1_L008_R2_001.fastq.gz \
q1=/lustre/scratch116/vr/projects/Tes1_S2_L008_R1_001.fastq.gz \
q2=/lustre/scratch116/vr/projects/Tes1_S2_L008_R2_001.fastq.gz \
q1=/lustre/scratch116/vr/projects/Tes1_S3_L008_R1_001.fastq.gz \
q2=/lustre/scratch116/vr/projects/Tes1_S3_L008_R2_001.fastq.gz \
q1=/lustre/scratch116/vr/projects/Tes1_S4_L008_R1_001.fastq.gz \
q2=/lustre/scratch116/vr/projects/Tes1_S4_L008_R2_001.fastq.gz \

            Output_directory: a director contained all the binned 10X reads\n");
:
```

Flowchart of binHiC



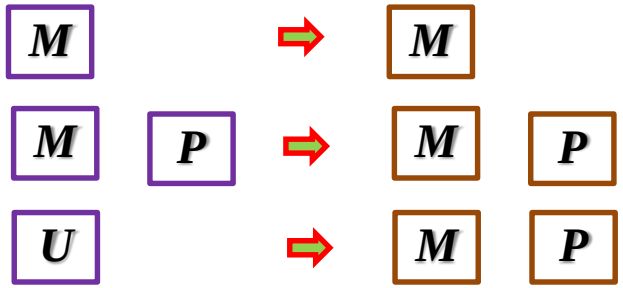
Read1	Read2	Assign the Pair to	
M	M	M	
M	P	M	P
M	U	M	
P	P	P	
P	M	M	P
P	U	P	
U	U	M	P

Flowchart of binLong



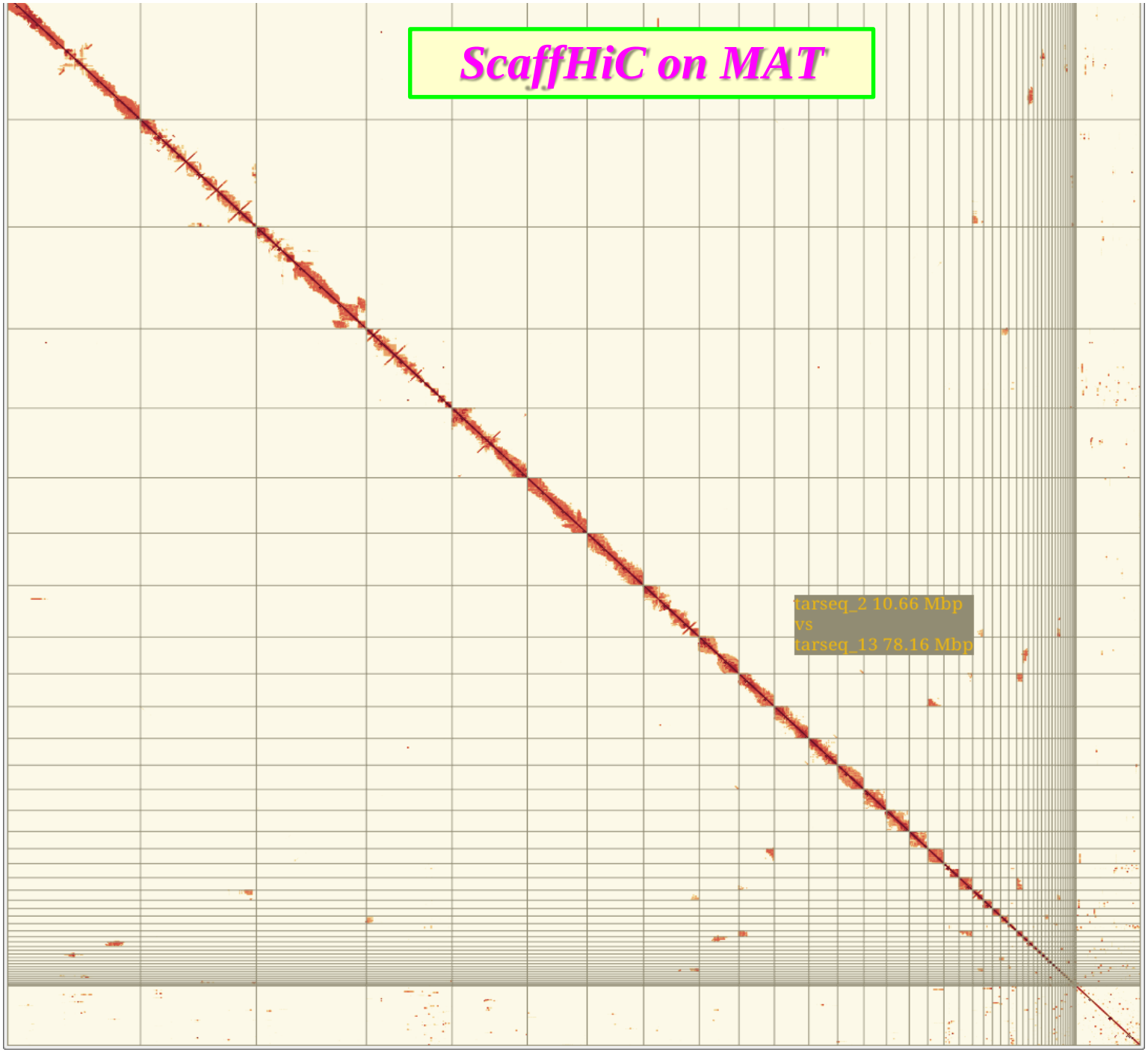
Seq1
Seq2
…
SeqN

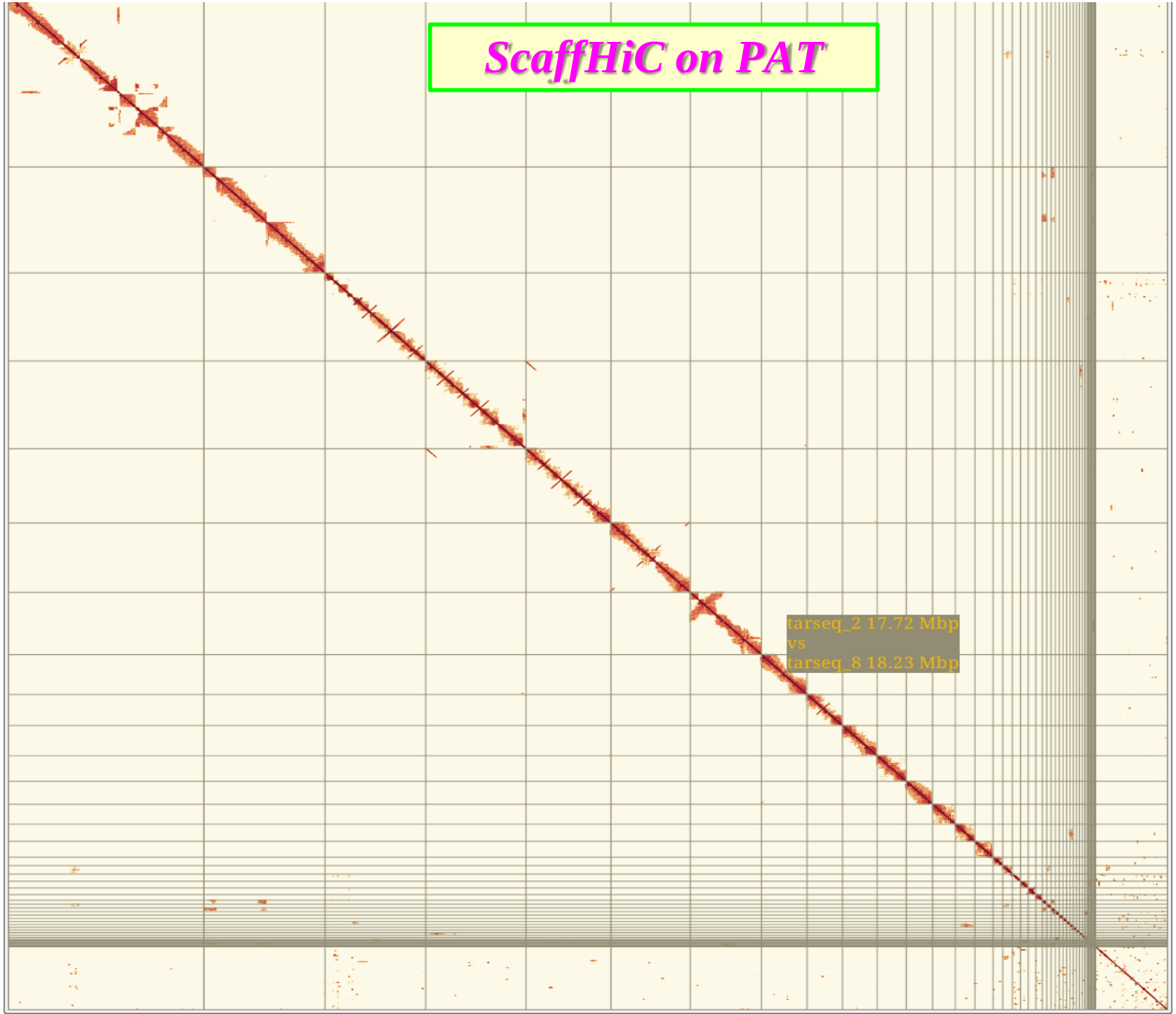
Assign the Read to



How to run binHiC and binLong

```
#### Run binHiC:

          $ /full/path/to/AncesBin/src/binHiC -nodes 30 MAT_ref.fasta PAT_ref.fasta Input_read_1.fq.gz Input_read_2.fq.gz Output_directory

          Parameters:
            nodes:              number of CPUs requested  [ default = 30 ]
            MAT_ref.fasta:      ancestry MAT assembly
            PAT_ref.fasta:      ancestry PAT assembly
               Note:            you need to rename the assembly file:
                                $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name MAT MAT_assembly.fa MAT_ref.fasta
                                $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name PAT PAT_assembly.fa PAT_ref.fasta
            Input_read_1.fq.gz: gzipped HiC read 1
            Input_read_2.fq.gz: gzipped HiC read 2
            Output_directory:   a director contained all the binned HiC reads\n");



#### Run binLong:

          $ /full/path/to/AncesBin/src/binLong -nodes 30 MAT_ref.fasta PAT_ref.fasta Input_data_file Output_directory  \

          Parameters:
            nodes:            number of CPUs requested  [ default = 30 ]
            MAT_ref.fasta:    ancestry MAT assembly
            PAT_ref.fasta:    ancestry PAT assembly
               Note:          you need to rename the assembly file:
                              $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name MAT MAT_assembly.fa MAT_ref.fasta
                              $ /full/path/to/AncesBin/src/scaff-bin/seqbin_rename -name PAT PAT_assembly.fa PAT_ref.fasta
            Input_data_file:  a text file to point the locations of the reads in gzipped files\n");

/lustre/fTakRub1/PacBio/fasta/m54097_180320_123755.subreads.fasta.gz \
/lustre/fTakRub1/PacBio/fasta/m54097_180321_135512.subreads.fasta.gz \
/lustre/fTakRub1/PacBio/fasta/m54097_180322_133901.subreads.fasta.gz \
/lustre/fTakRub1/PacBio/fasta/m54097_180323_154627.subreads.fasta.gz \

            Output_directory: a director contained all the binned PacBio or ONT long reads\n");

:
```


ScaffHiC on MAT
tarseq_2 10.66 Mbp
vs
tarseq_13 78.16 Mbp


ScaffHiC on PAT
tarseq_2 17.72 Mbp
vs
tarseq_8 18.23 Mbp

Download and Compile

```
# AncesBin v1.0
Pipeline to bin 10X, HiC, PacBio and ONT reads based on ancestry assemblies.

### Download and Compile:
Requirements for compiling: gcc:

    $ git clone  https://github.com/wtsi-hpag/AncesBin.git
    $ cd AncesBin
    $ ./install.sh

If everything compiled successfully you must see the final comment:
                "Congrats: installation successful!"
```