

Demo - 10x Genomics Cloud Analysis

1. Download demo dataset

<https://bit.ly/4zkdCub>

2. Create account on 10x Genomics Cloud Analysis
(<https://www.10xgenomics.com/products/cloud-analysis>)

3. Upload dataset

4. Start analysis



Single-cell RNA-Seq Analysis (1)

生物資訊與數據分析

10/06/2026

開課時間：2:00-4:00PM

上課地點：環變大樓6005室

余承欣 (Chen-Hsin Albert Yu)

IMB Bioinformatics Core

Outline

1. 10x scRNA-seq data analysis demo

10x Genomics Cloud Analysis

(<https://www.10xgenomics.com/products/cloud-analysis>)

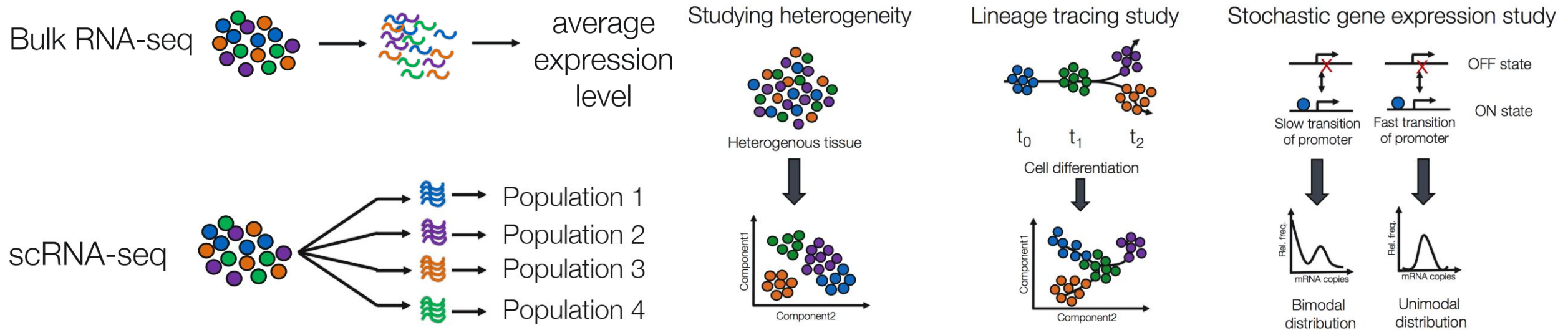
2. 10x scRNA-seq workflow and application

3. 10x scRNA-seq data analysis



Why using single-cell RNA-Seq

- Resolving Cellular Heterogeneity & Rare Cell Discovery (ex: cancer stem cells)
- Tracking Cell Population Dynamics (ex: gene expression shifted after treatment)
- Reconstructing Developmental Trajectories & Pseudotime
- Uncovering Gene Regulatory Networks & Cell Communication



Commercial single cell RNA-Seq platforms

10x Genomics Chromium (70~80% Market)

BD Rhapsody

Takara Smart-seq

Drop-seq

MERFISH

Seq-Well

CITE-seq

...

Demo - 10x Genomics Cloud Analysis

1. Download demo dataset

<https://bit.ly/4zkdCub>

2. Create account on 10x Genomics Cloud Analysis
(<https://www.10xgenomics.com/products/cloud-analysis>)

3. Upload dataset

4. Start analysis



Dataset

10x Genomics Flex (Fixed RNA Profiling) dataset consisting of ~16,000 Human PBMCs multiplexed across 4 samples (4plex) with TotalSeq-C antibody feature barcoding.

https://www.10xgenomics.com/datasets/Flexv2_16k_Human_PBMCs_TotalseqC_4plex

Input Files

1. fastq_gex/
2. fastq_ab/
3. feature_reference.csv

id	name	read	pattern	sequence	feature_type
CD80	Hu.CD80	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	ACGAATCAATCTGTG	Antibody Capture
CD86	Hu.CD86	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	GTCTTTGTCAGTGCA	Antibody Capture
CD274	Hu.CD274	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	GTTGTCCGACAATAC	Antibody Capture
CD273	Hu.CD273	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	TCAACGCTTGGCTAG	Antibody Capture
CD275	Hu.CD275	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	GTGCATTCAACAGTA	Antibody Capture
CD155	Hu.CD155	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	ATCACATCGTTGCCA	Antibody Capture
CD47	Hu.CD47	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	GCATTCTGTACCTA	Antibody Capture
CD70	Hu.CD70	R2	^NNNNNNNNNN(BC)NNNNNNNNNN	CGCGAACATAAGAAG	Antibody Capture

Configuration

Input CSV

```
[gene-expression]
reference,/mnt/deck-code/references/refdata-cellranger-GRCh38-2024-A
chemistry,auto
no-secondary,false
enable-tsne,false
create-bam,false
probe-set,/mnt/deck-code/packages/cellranger-10.1.0/probe_sets/Chromium_Human_Transcriptome_Probe_Set_v2.0.0_GRCh38-2024-A.csv
filter-probes,true
tenx-cloud-token-path,/mnt/deck-io/inputs/b60537e8-37f9-4b18-81e2-bb4edfe81990/analysis_inputs/tenx_cloud_token.txt
cell-annotation-model,auto
skip-cell-annotation,false
filter-high-occupancy-gems,true

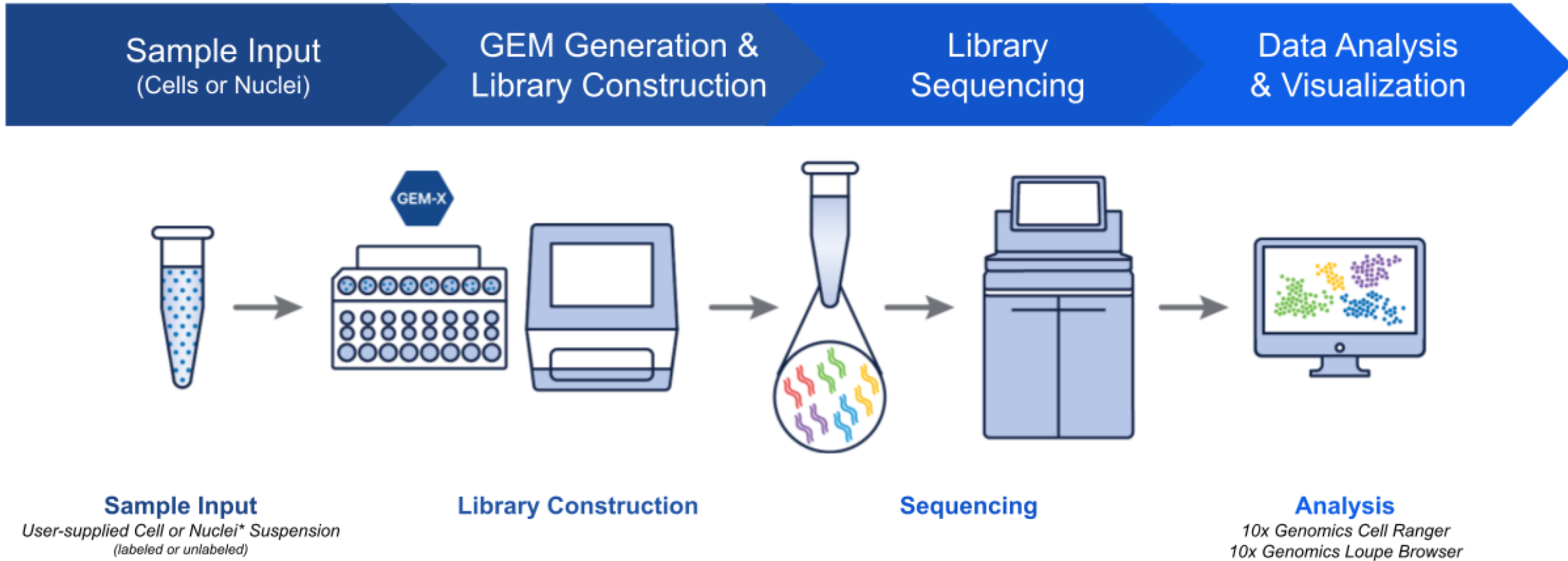
[feature]
reference,/mnt/deck-io/inputs/b60537e8-37f9-4b18-81e2-bb4edfe81990/analysis_inputs/feature_reference.csv
filter-aggregates,false

[libraries]
fastq_id,fastqs,lanes,feature_types,subsample_rate
16k_Human_PBMCs_TotalseqC_4plex_ab,/mnt/deck-io/inputs/b60537e8-37f9-4b18-81e2-bb4edfe81990/ad6e0010-5e61-46ab-b58b-4e0b8db3a6c2,any,Antibody Capture,
16k_Human_PBMCs_TotalseqC_4plex_gex,/mnt/deck-io/inputs/b60537e8-37f9-4b18-81e2-bb4edfe81990/ed0b7001-f183-4c5e-a280-1d7a838be974,any,Gene Expression,

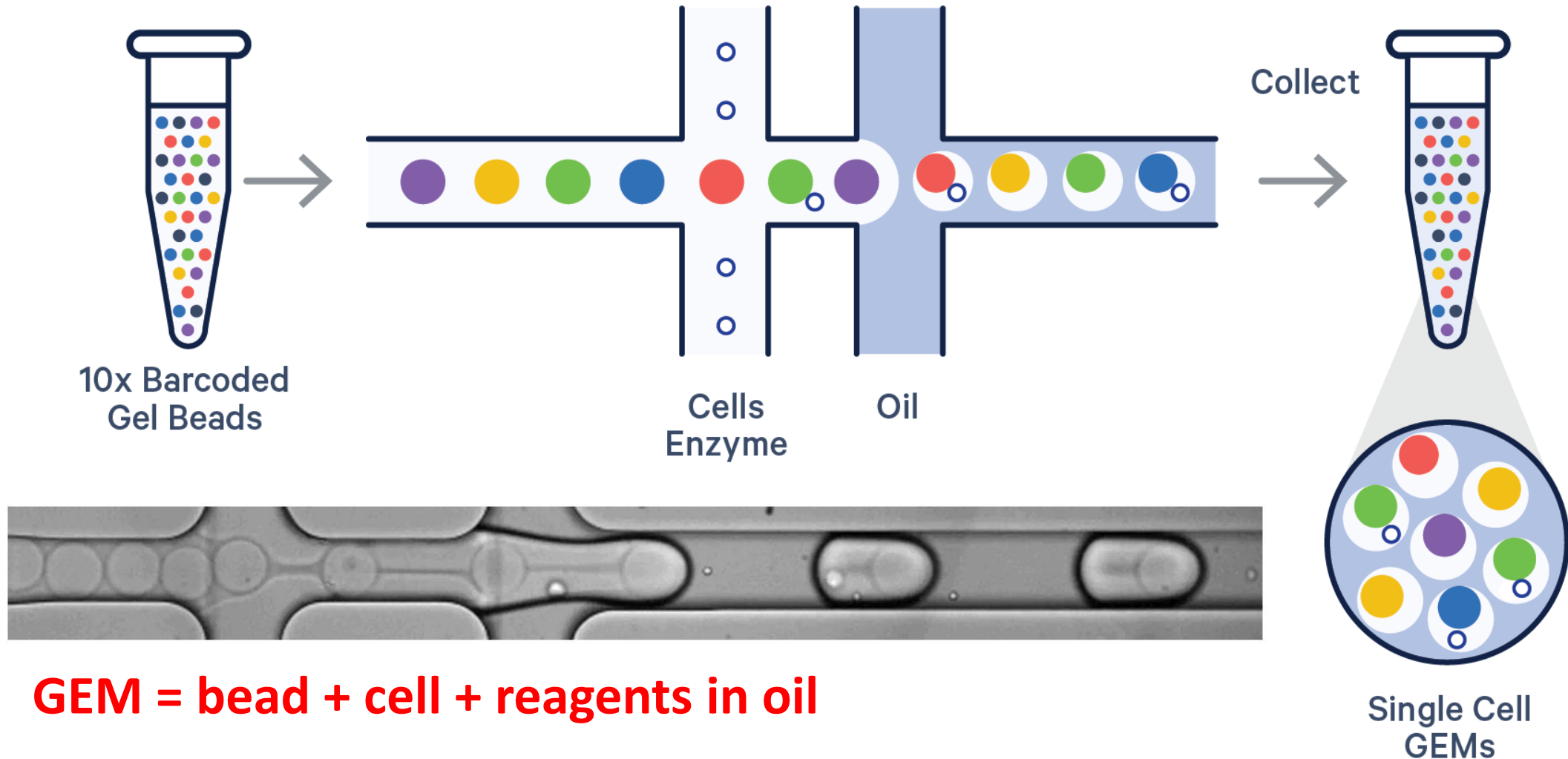
[samples]
sample_id,probe_barcode_ids,description,expect_cells,force_cells
A01,C-A01,,,
B01,C-B01,,,
C01,C-C01,,,
D01,C-D01,,,

```

GEM-X technology workflow



GEM microfluidic chips



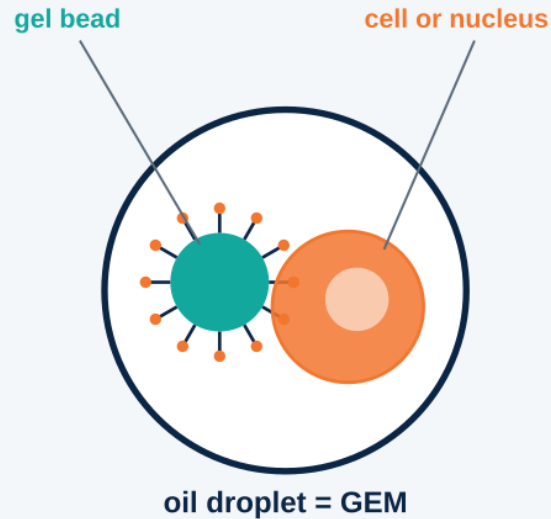
GEM = bead + cell + reagents in oil

Most GEMs are empty (>85%)

Gel Bead-in-Emulsion

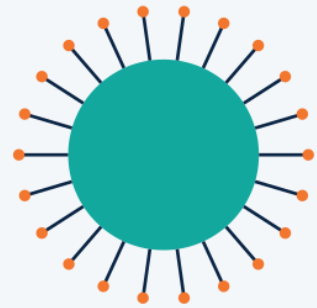
10x GEM (One Droplet, One Cell, One Barcode)

1 Cell isolation & partitioning



Each cell sits in its own oil compartment, so nucleic acids cannot cross-contaminate.

2 Cell-specific barcoding



gel bead: millions of oligos, all sharing ONE barcode



barcode: which cell

(16 nt, one per bead)

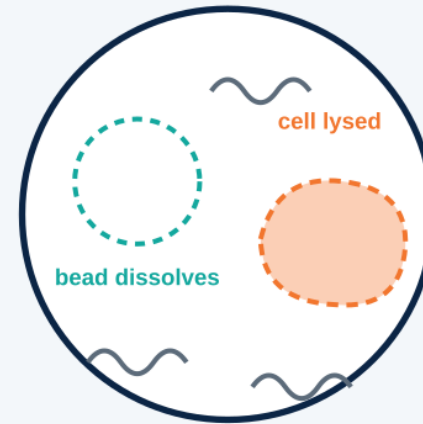
UMI: which molecule

(counts transcripts, removes PCR duplicates)

capture sequence: poly-dT for mRNA

(or a primer/adaptor for other assays)

3 Lysis & enzymatic reaction



mRNA / DNA released

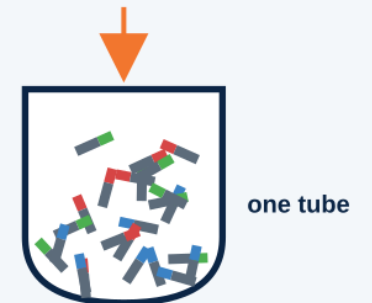


barcoded cDNA (or DNA fragment):
barcode is covalently attached
to the cell's own molecules

4 High-throughput pooling



break the emulsion



library prep
+ sequencing



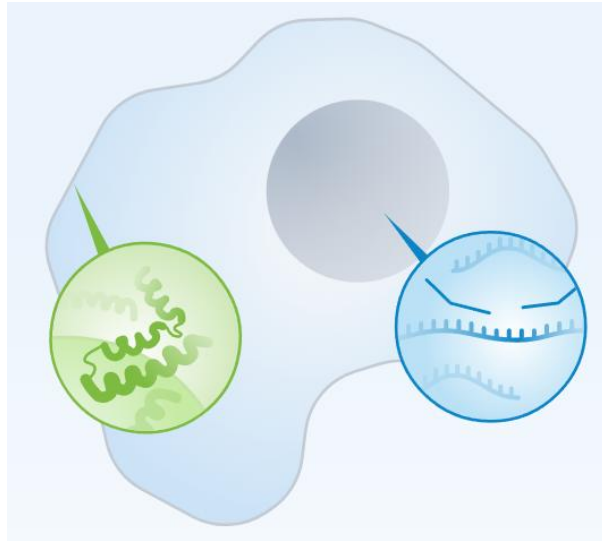
Software reads each barcode and maps every read back to its cell.

10x Chromium Single-cell Gene Expression



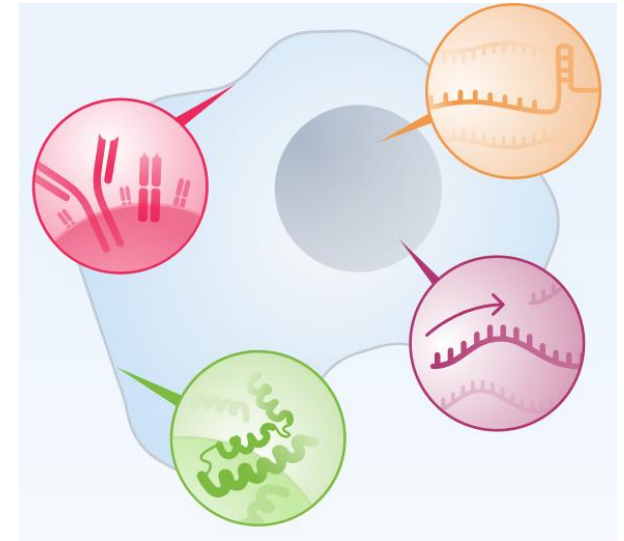
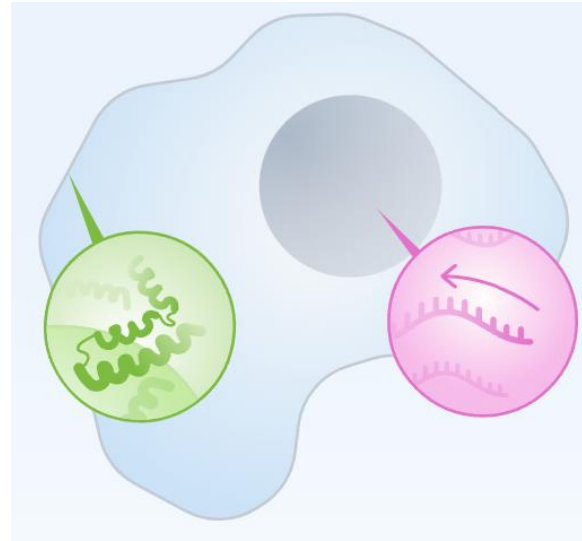
Probe based

GEM-X Flex Gene Expression



Reverse transcription based

GEM-X Universal 3' Gene Expression GEM-X Universal 5' Gene Expression



Compatible samples

GEM-X Flex Gene Expression



Cell suspension



Nuclei suspension



Fresh tissue



Frozen tissue



FFPE tissue



PFA-fixed tissue



Flow-sorted cells



Fixed whole blood



Organoids

GEM-X Universal 3' Gene Expression



Cell suspension



Nuclei suspension



Fresh tissue



Frozen tissue



Fixed human PBMCs



Flow-sorted cells



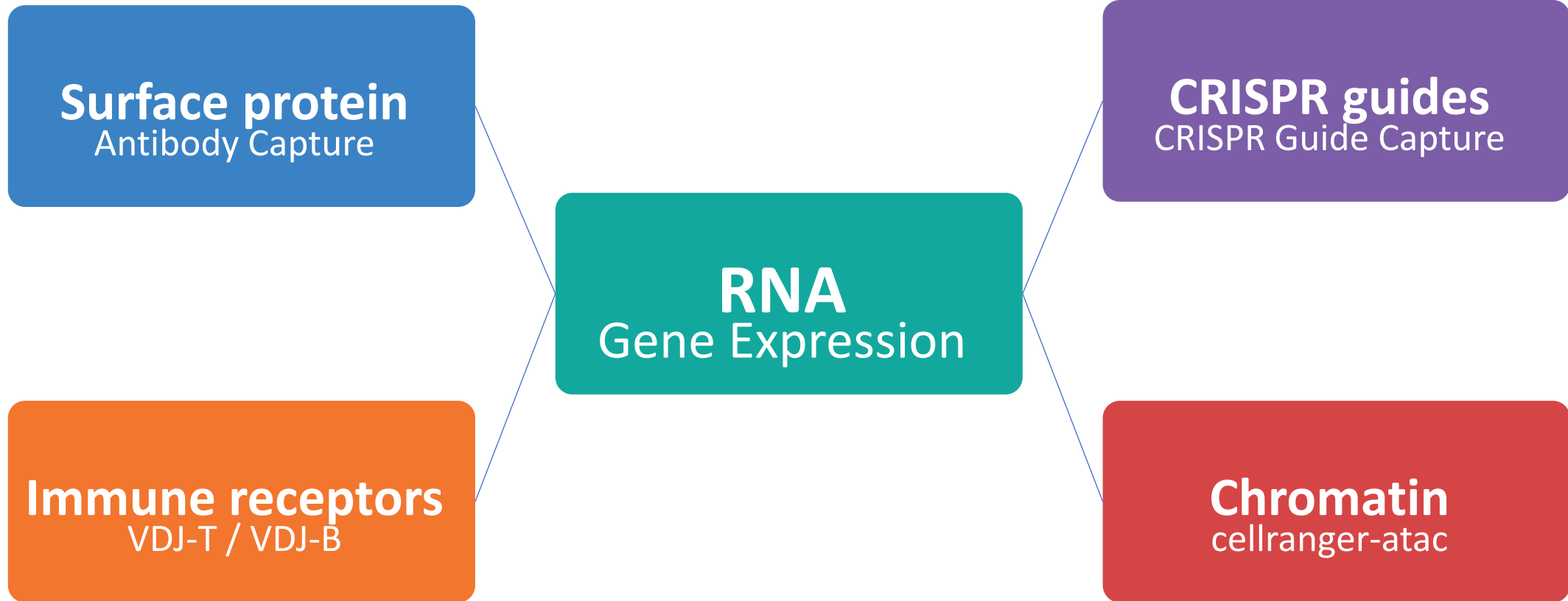
Organoids

GEM-X Universal 5' Gene Expression

Living Cells or Nuclei?

Approach	Input	Strength	Limitation
scRNA-seq	Live single cells	Full cytoplasmic RNA, surface protein labeling	Needs fresh, viable cells
snRNA-seq	Isolated nuclei	Frozen or hard-to-dissociate tissue	Nuclear RNA only, fewer UMIs
Flex (fixed)	Formaldehyde-fixed cells or nuclei	Decouples collection from run; up to 384 samples	Probe-based: only targeted transcripts

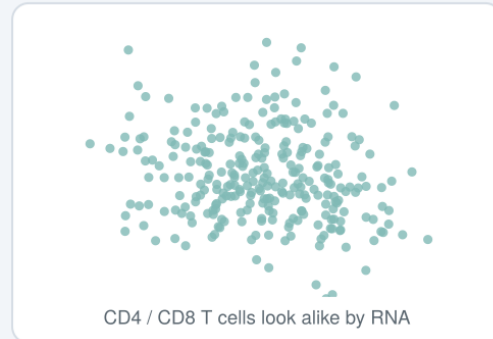
Multimic capabilities



Multimic capabilities

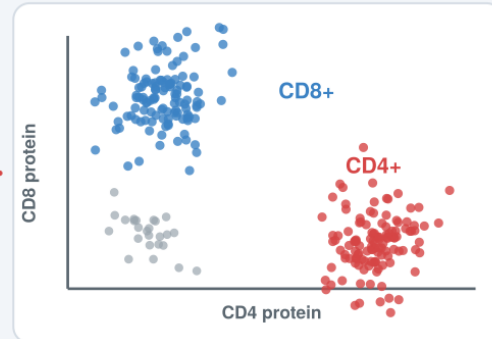
1 Protein + RNA

RNA only: overlapping clusters



+

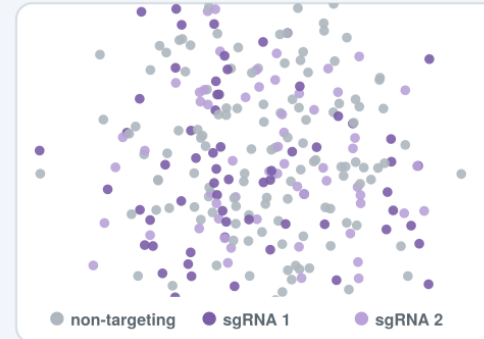
RNA + protein (CD4 vs CD8)



Surface markers separate populations that RNA blurs.

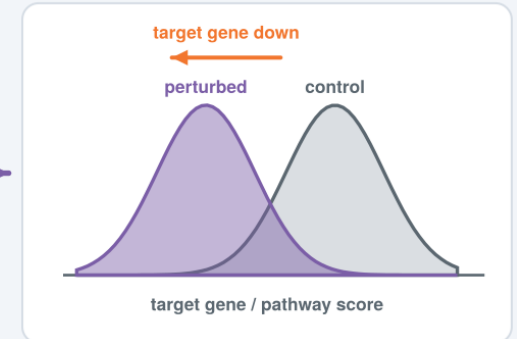
2 CRISPR + RNA

Guide assigned to each cell



+

Transcriptional phenotype



Links each perturbation to its transcriptional phenotype.

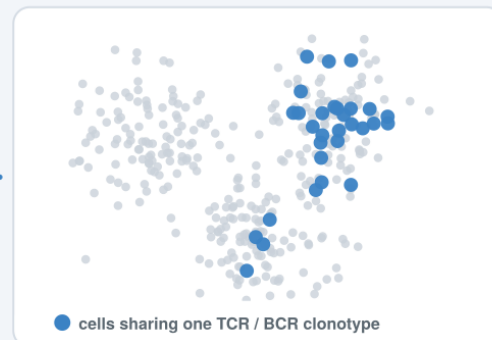
3 V(D)J + RNA

RNA only: cell states



+

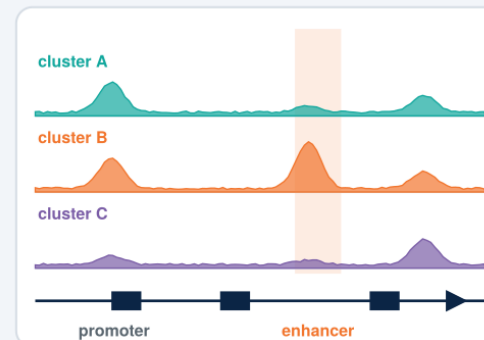
RNA + V(D)J: clonotype overlay



Connects clonotype to cell state.

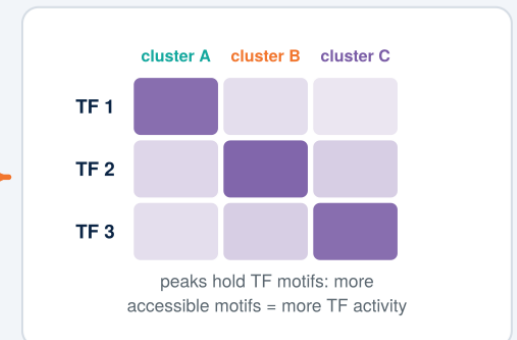
4 ATAC

Accessibility tracks (by cluster)



+

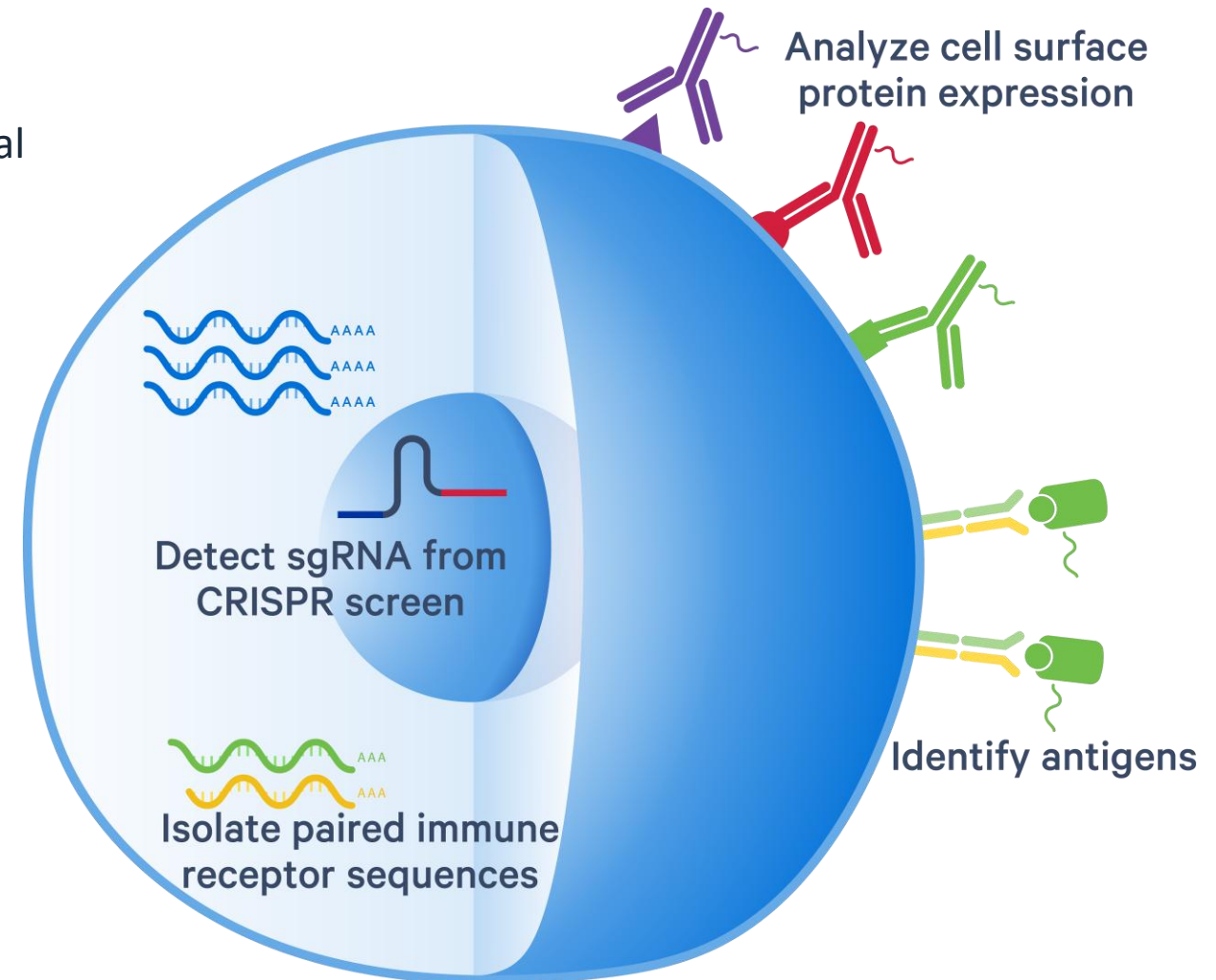
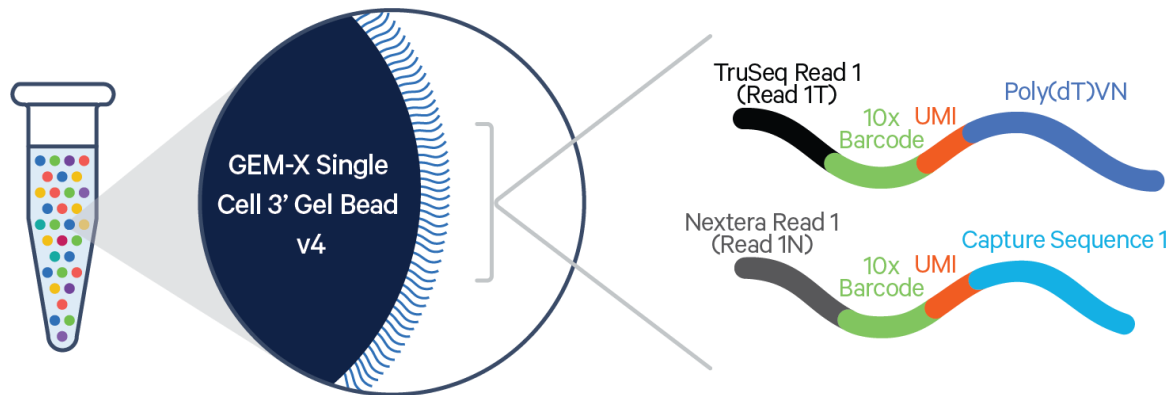
TF motif activity per cluster



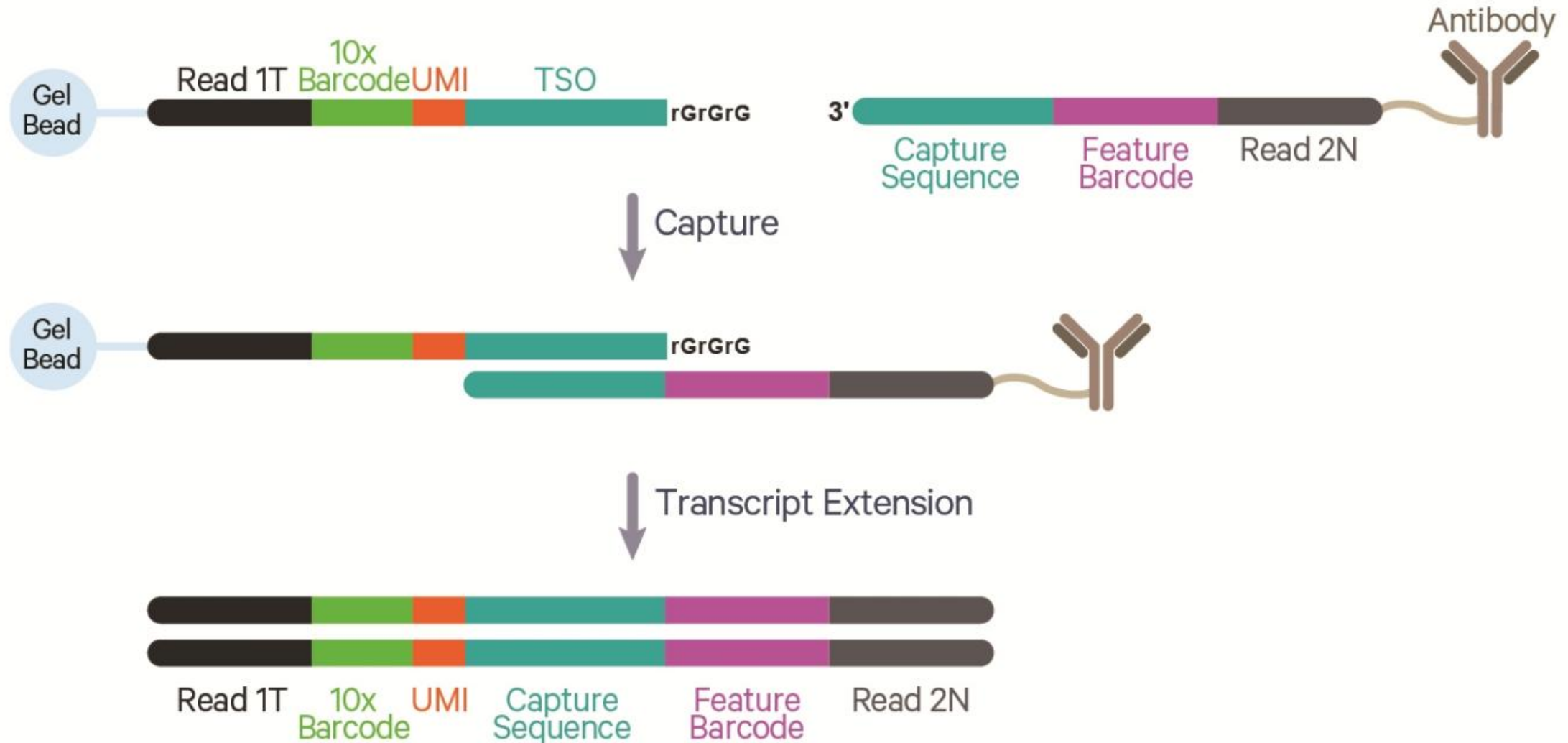
Reveals regulatory regions and transcription factor activity.

Feature Barcode Technology

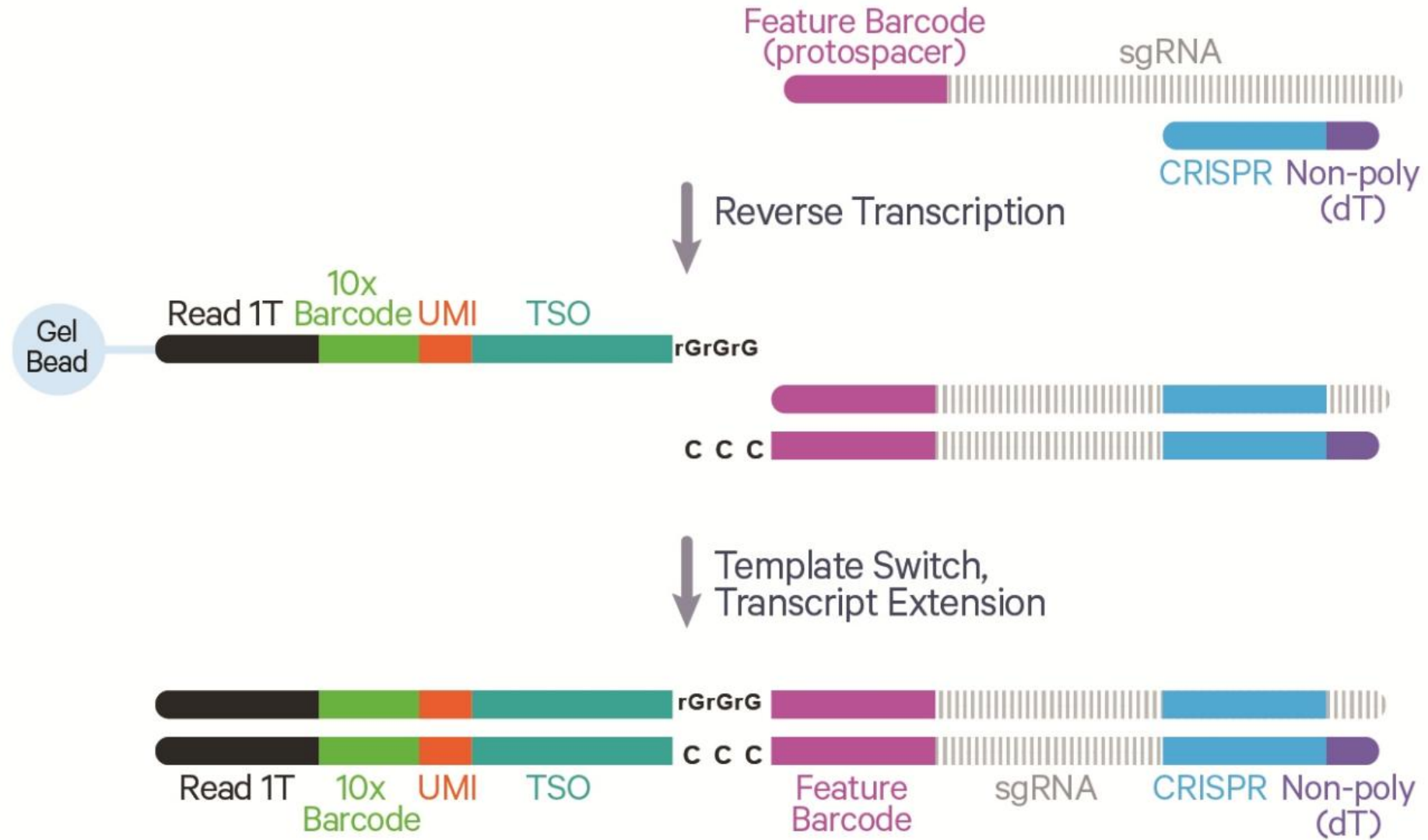
- **Surface Protein + RNA:** markers separate populations.
- **CRISPR + RNA:** links each perturbation to its transcriptional phenotype.
- **V(D)J + RNA:** connects clonotype to cell state.
- **3' multiplexing**
- **Custom Feature Barcode**

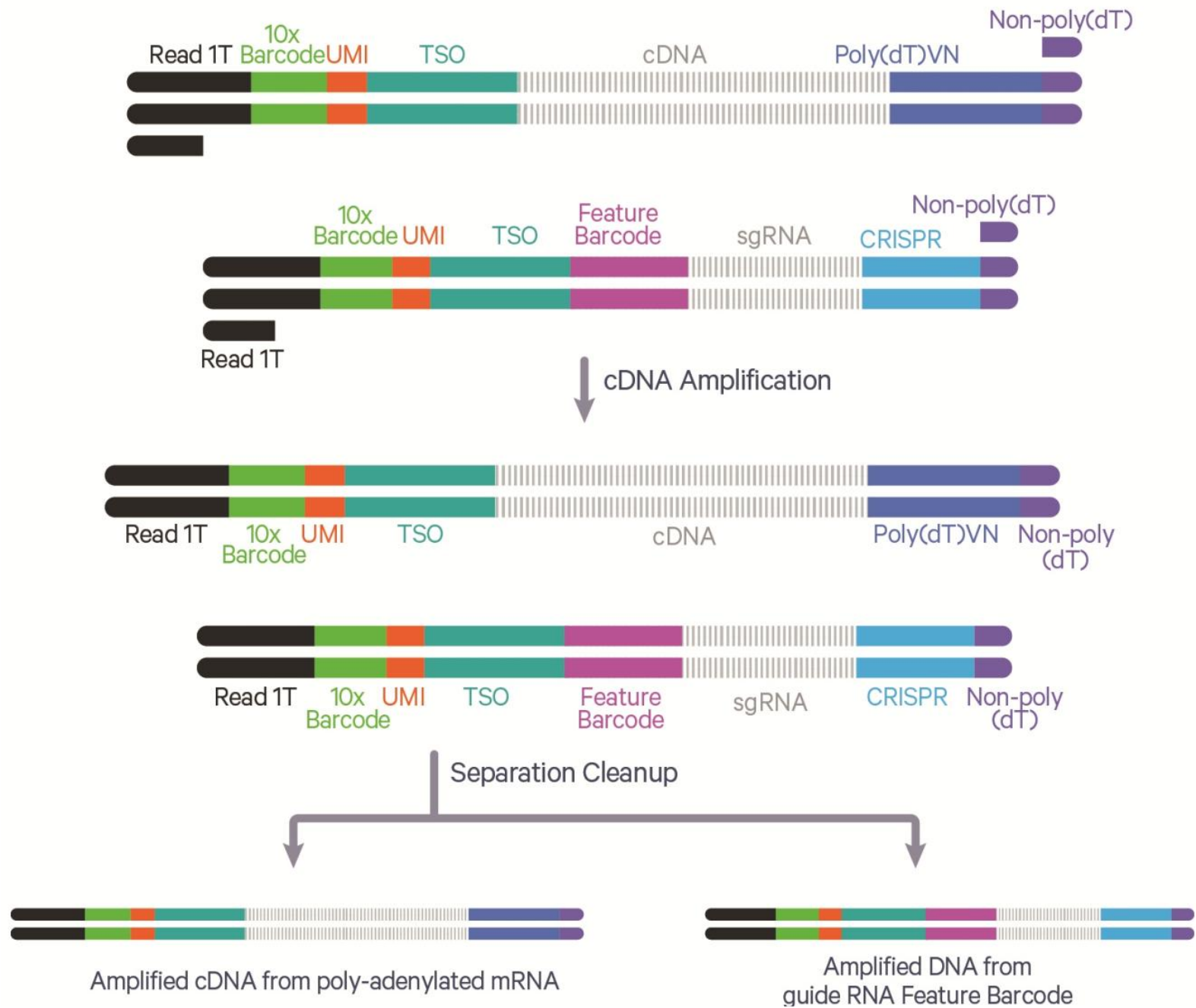


Cell Surface Protein



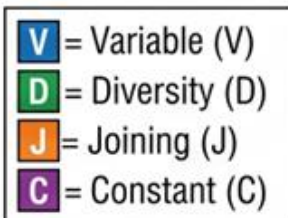
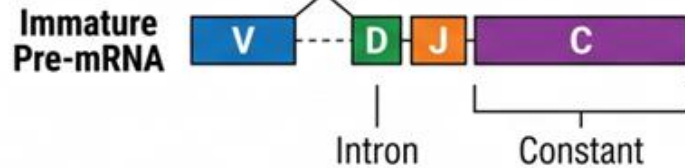
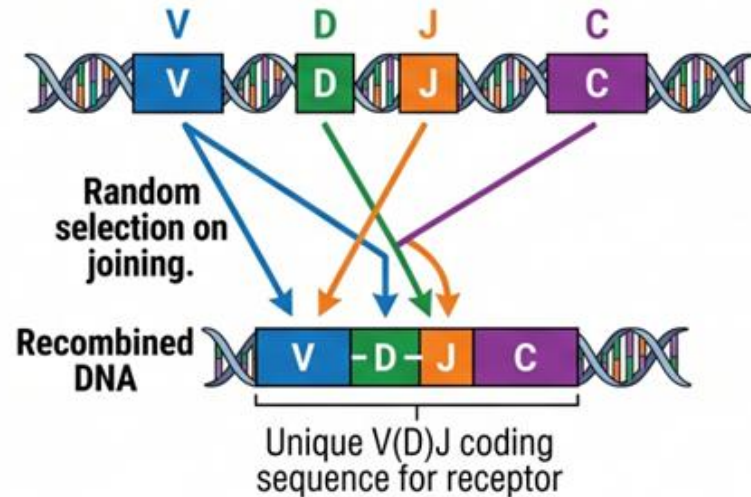
CRISPR Screening



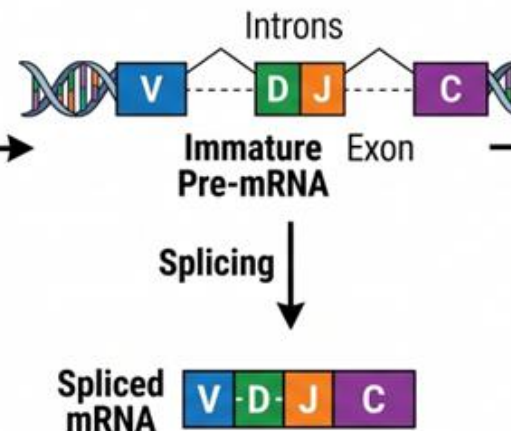


V(D)J Recombination

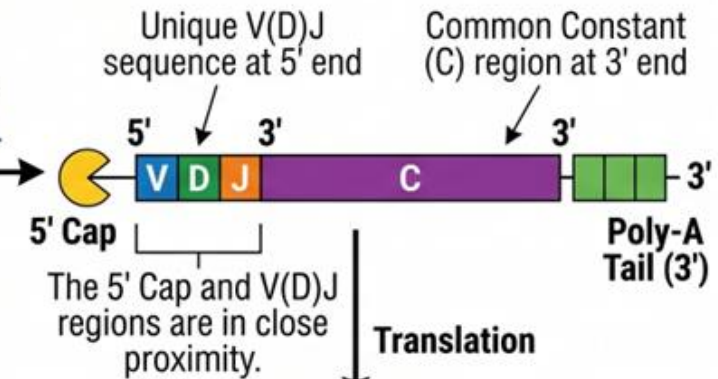
1. Germline DNA to V(D)J Recombination



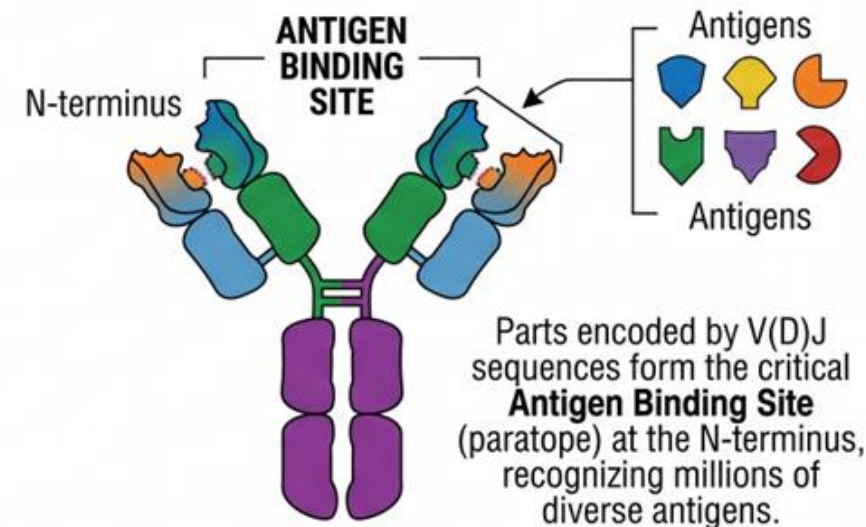
2. Transcription and Splicing



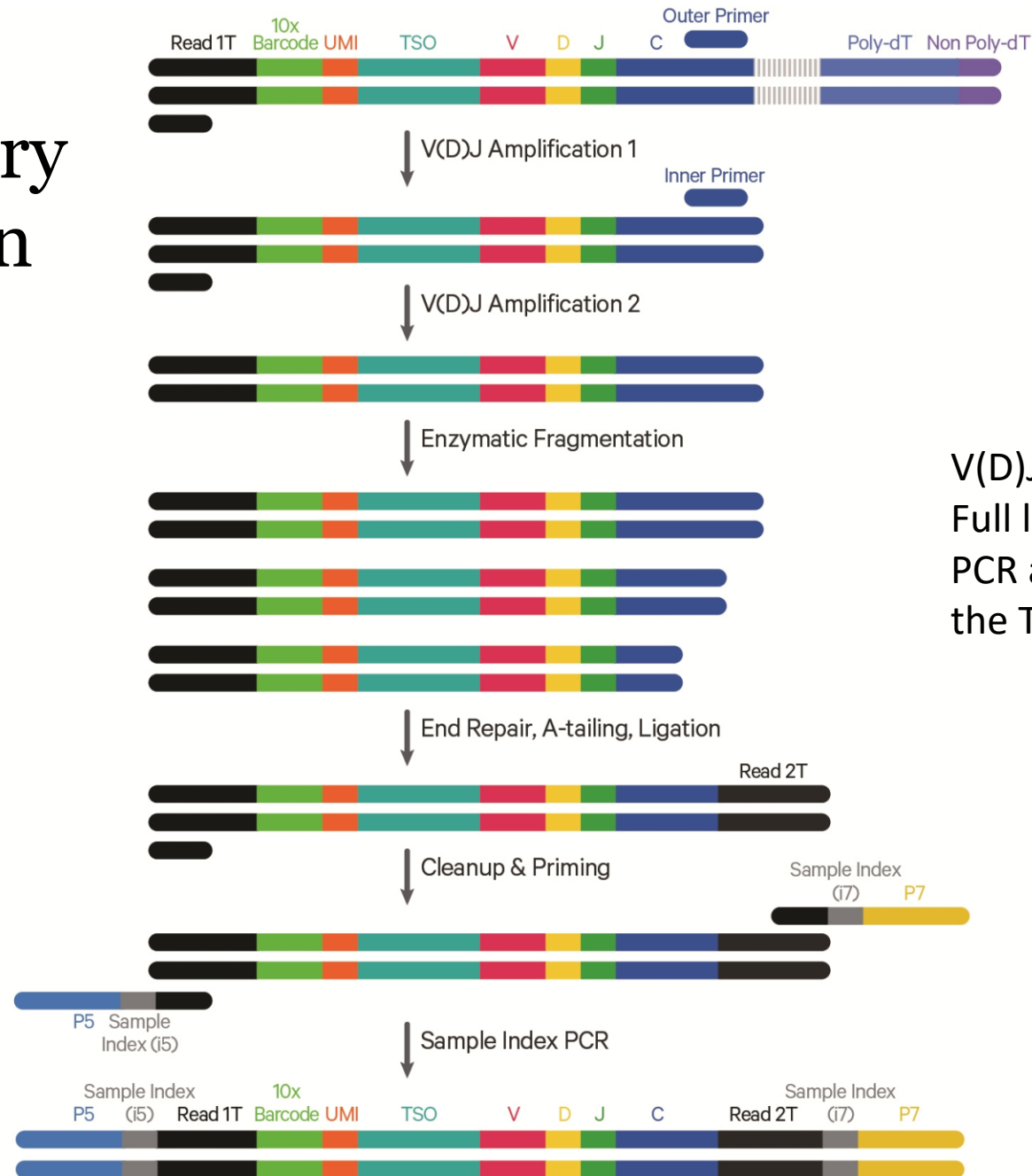
3. Mature mRNA Structure



4. Protein and Binding Site

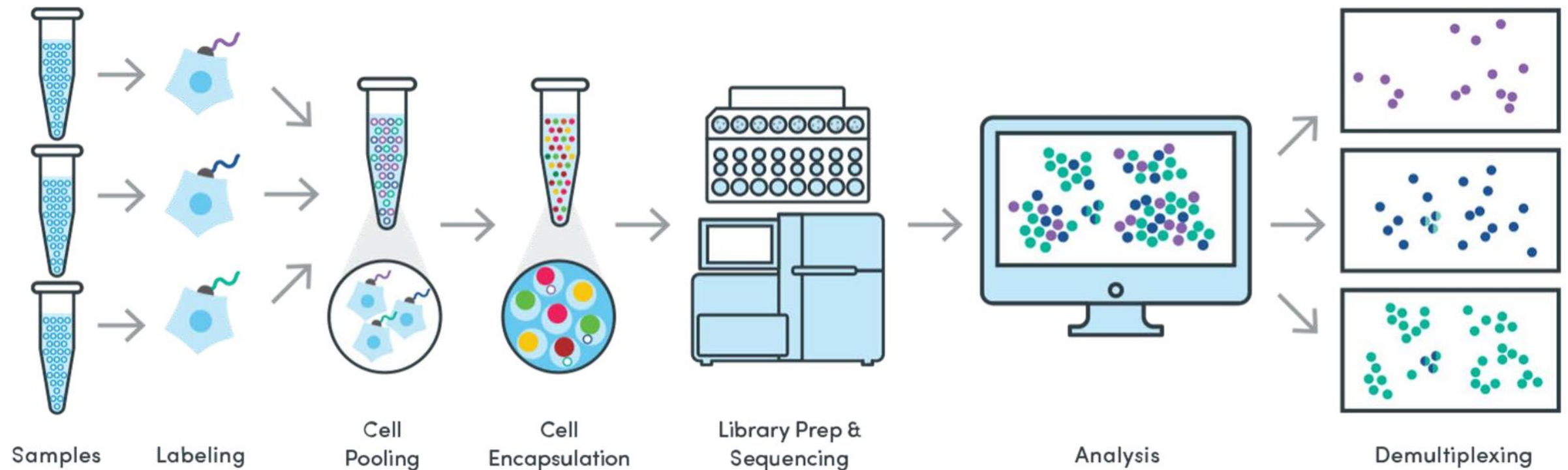


V(D)J Library Construction



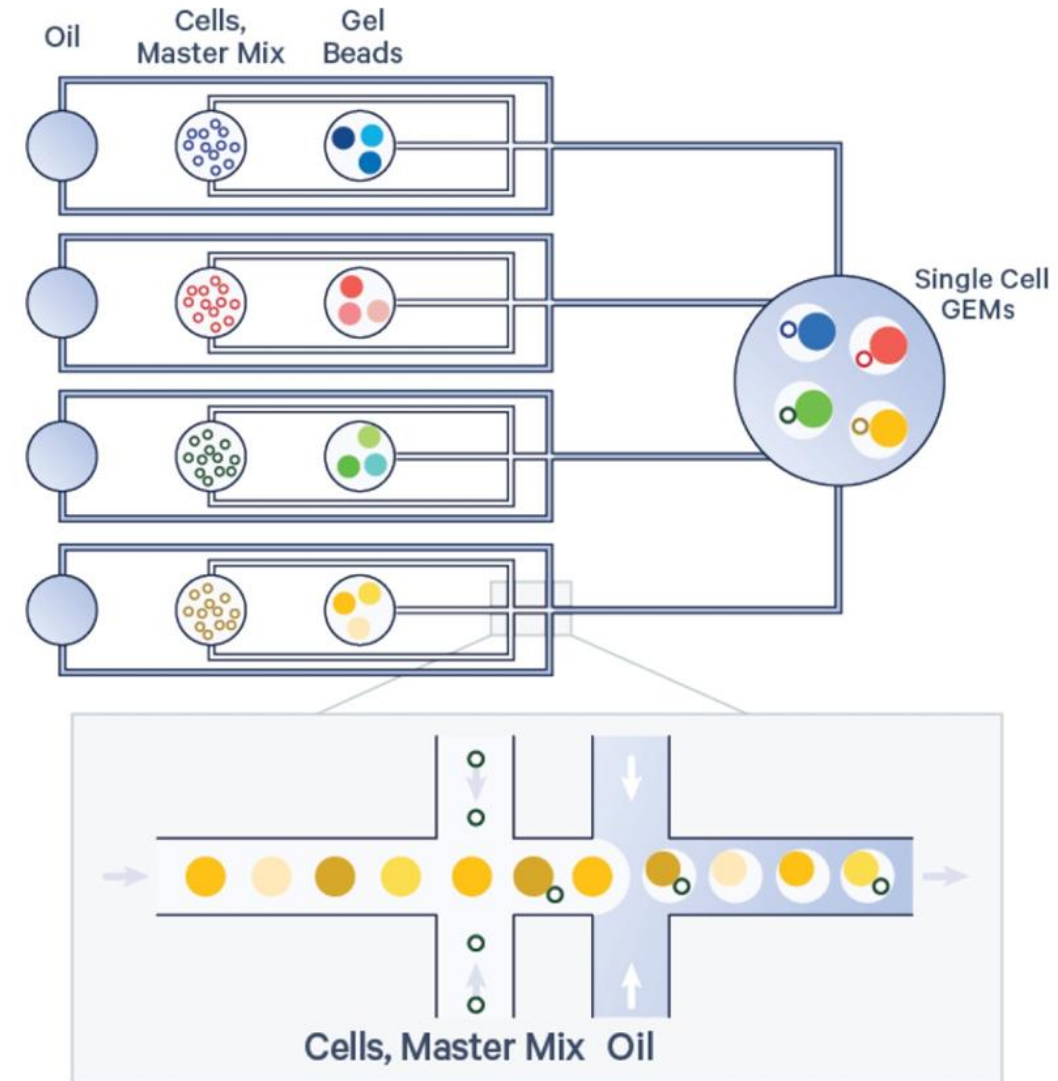
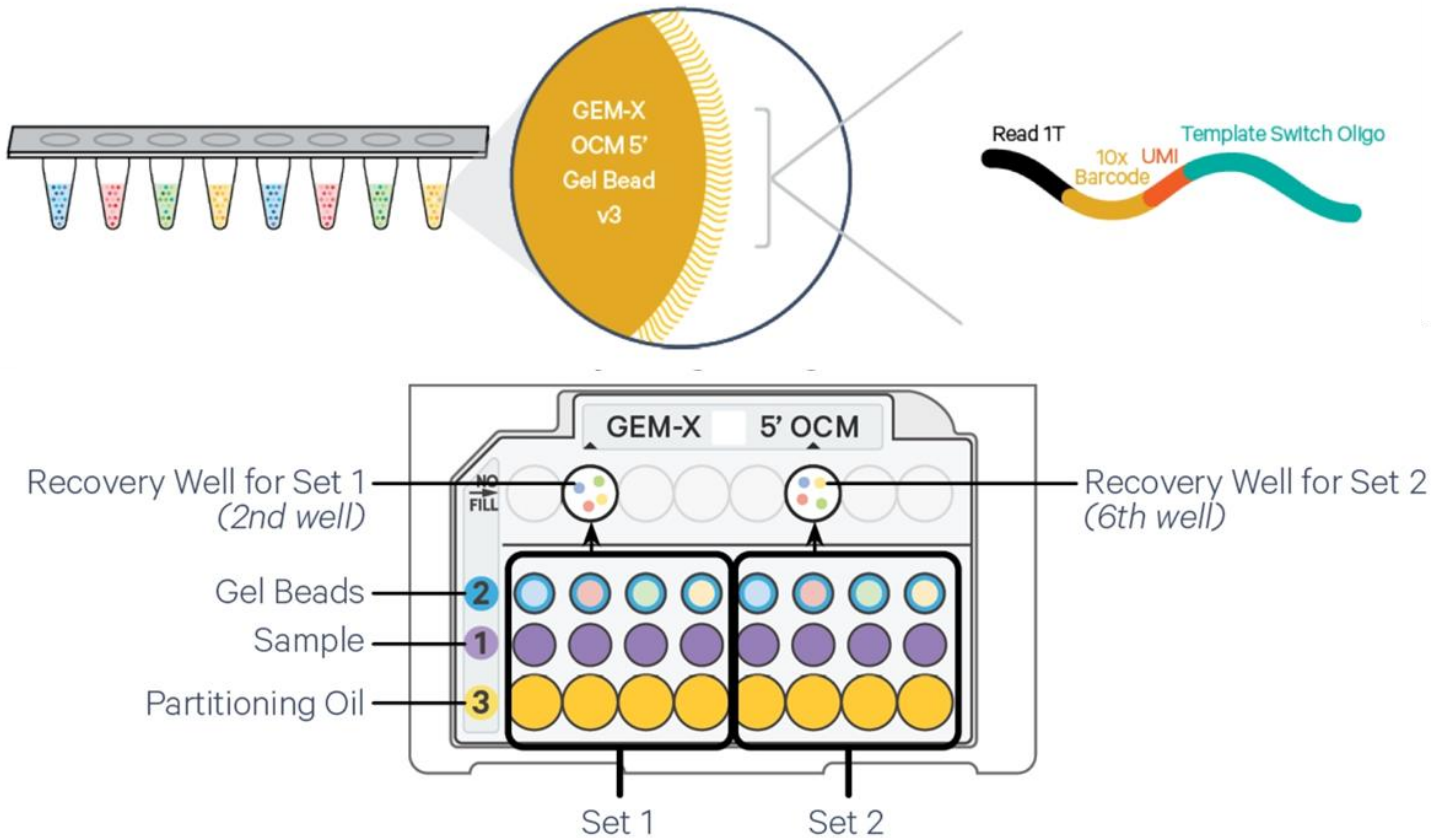
V(D)J Amplification from cDNA:
Full length V(D)J segments (10x Barcoded) via
PCR amplification with primers specific to either
the TCR or BCR constant regions.

3' multiplexing



OCM (on-chip multiplexing)

The 4 color coded gel beads, each with a unique list of ~900,000 barcodes



Sequencing Library

Chromium Single Cell V(D)J Library



Chromium Single Cell 5' Gene Expression Library



Chromium Single Cell 5' CRISPR Screening Library



Sequencing Library Structure



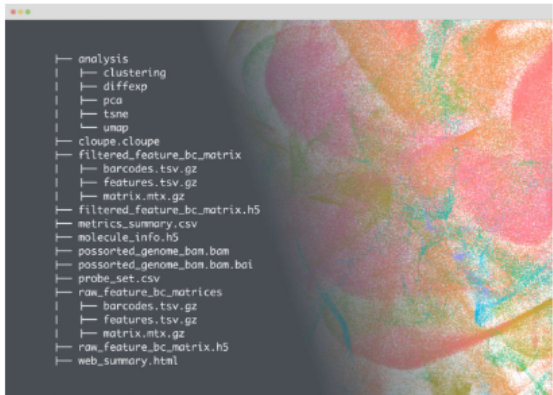
Index	Purpose
P5/P7	the library fragment to attach to the flow cell surface (Illumina)
Sample Index (i5/i7)	demultiplexing sequencing runs when running more than one sample per lane (Illumina)
Seq Read1/2	Illumina sequencing adaptors
10x Barcode (16 bp)	Cell Barcode (Bead specific) --- each Cell
UMI (12 bp)	12 nt Unique Molecular Identifier --- each RNA molecule
Capture Seq	The reverse complement of the sequence inserted into the DNA or RNA based Feature
Feature Barcode	Barcode for Capture Seq

FASTQ naming and organization

EXAMPLE

```
fastqs/  
├─ sampleA_S1_L001_R1_001.fastq.gz  
├─ sampleA_S1_L001_R2_001.fastq.gz  
├─ sampleA_S1_L001_I1_001.fastq.gz  
├─ sampleA_S1_L001_I2_001.fastq.gz  
├─ sampleA_S1_L002_R1_001.fastq.gz  
└─ ...  
  
# pattern:  
# <sample>_S<n>_L<lane>_<read>_001.fastq.gz
```

- The prefix before _S1_ is the sample name Cell Ranger selects with --sample or fastq_id.
- R1 = barcode + UMI, R2 = transcript, I1/I2 = index reads.
- Keep each library type (GEX, antibody, guide) in clearly named prefixes.
- Multiple lanes or flow cells: give a comma-separated list of folders.



What is Cell Ranger?

Cell Ranger is a set of analysis pipelines for processing Chromium single cell data. It performs barcode processing and UMI counting to quantify gene expression (from 3', 5', and Flex assays), assembles V(D)J immune receptor sequences, and analyzes Feature Barcode data for applications such as cell surface protein analysis and sample multiplexing.

 [Download Cell Ranger](#)

 [Use Cloud Analysis](#)

 [Get Started](#)

Find your primary pipeline and learn how to use it

Assay *

☐ Universal 3' Gene Expression

☐ Universal 5' Gene Expression

☐ Flex v1/v2 Gene Expression

Cell Multiplexing *

☐ Singleplex

☐ Multiplex

☐ CellPlex Multiplexing

☐ On Chip Multiplexing (OCM)

☐ Hashtag Multiplexing

Library Types *

(Select all that apply)

☐ GEX

☐ VDJ

☐ Antibody

☐ CRISPR

☐ BEAM

Select your library type(s) above, and we will guide you to the appropriate pipeline and corresponding documentation.

The Cell Ranger toolbox



count

One GEM well, GEX (+ feature barcode).



multi

GEX, V(D)J, feature barcode, multiplexing and Flex in one config.



aggr

Combine GEM wells or samples; depth normalization.



reanalyze

Rerun clustering with new parameters or barcode lists from Loupe.



mkref

Build a custom reference.



cellranger-atac

Separate binary for scATAC: count, aggr, reanalyze, mkref.

Installing Cell Ranger

BASH

```
# download the tarball from the 10x Downloads page
tar -xzvf cellranger-<version>.tar.gz
export PATH=$PWD/cellranger-<version>:$PATH

cellranger --version
cellranger testrun --id=tiny
cellranger sitecheck > sitecheck.txt
```

- Linux 64-bit; use a workstation or cluster node with plenty of cores and RAM.
- testrun runs a tiny pipeline to verify the installation.
- sitecheck collects system information for support.
- Downloads page also hosts references and probe sets.

What Cell Ranger needs

Input	Needed when	Notes
FASTQ files	Always	From the sequencer (demultiplexed)
Transcriptome reference	GEX libraries (non-Flex)	Folder with genome + GTF index
Feature Reference CSV	Feature Barcode library (antibody, CRISPR)	Described in the count section
Probe set reference CSV	Flex only	Human/mouse files on Downloads page
multi config CSV	cellranger multi only	Sections define libraries and samples

Reference transcriptome

1. Downloadable pre-built references transcriptomes:

<https://www.10xgenomics.com/support/software/cell-ranger/downloads#reference-downloads>

2. mkgtf & mkref

```
cellranger mkgtf ref_genome.gtf \  
    ref_genome.filtered.gtf \  
    --attribute=gene_biotype:protein_coding
```

```
cellranger mkref \  
    --genome=ref_genome \  
    --fasta=ref_genome.primary_assembly.fa \  
    --genes=ref_genome.filtered.gtf
```

```
refdata-cellranger-arc-GRCh38-2020-A-2.0.0/  
refdata-cellranger-arc-mm10-2020-A-2.0.0/  
refdata-cellranger-GRCh38-3.0.0/  
refdata-cellranger-medaka/  
refdata-cellranger-mm10-3.0.0/  
refdata-cellranger-mm10-3.0.0_Ssna1e2/  
refdata-cellranger-mm10-3.0.0_Ssna1ko/  
refdata-cellranger-vdj-GRCh38-alts-ensembl-5.0.0/  
refdata-cellranger-vdj-GRCm38-alts-ensembl-5.0.0/  
refdata-cellranger-zebrafish-GRCz11/  
refdata-gex-GRCh38-2020-A/  
refdata-gex-GRCh38-2024-A/  
refdata-gex-GRCh38_and_GRCm39-2024-A/  
refdata-gex-GRCh38-and-mm10-2020-A/  
refdata-gex-GRCm39-2024-A/  
refdata-gex-mm10-2020-A/
```

cellranger count

BASH

```
cellranger count --id=pbmc_1k_count \  
  --transcriptome=/path/to/GRCh38-2020-A \  
  --fastqs=/path/to/fastqs \  
  --sample=pbmc_1k_v3 \  
  --create-bam=true
```

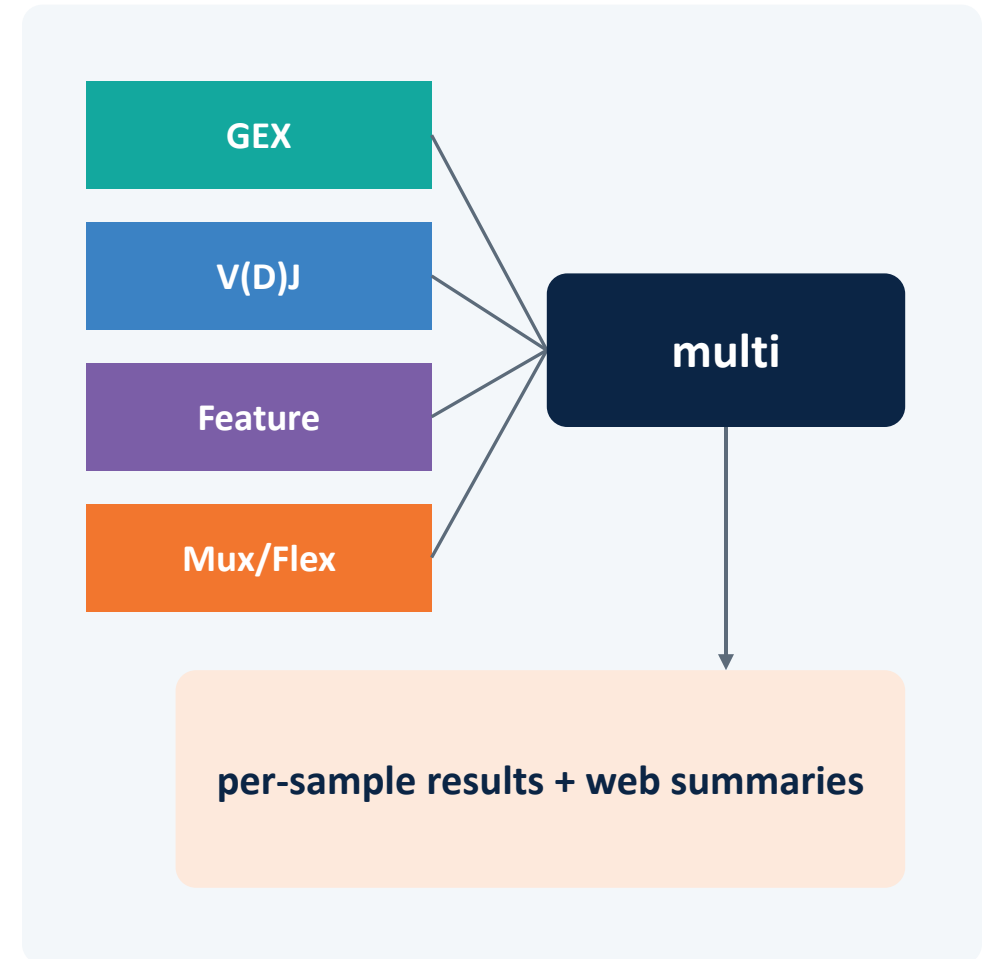
- **--id** run name and output folder.
- **--transcriptome** reference folder.
- **--fastqs / --sample** where the reads are and which prefix to use.
- **--create-bam** required flag; false saves time and disk space.

Useful count options

Option	What it does
<code>--expect-cells N</code>	Expected recovered cells, input to cell calling
<code>--force-cells N</code>	Bypass cell calling and keep top N barcodes (min. 10)
<code>--libraries, --feature-ref</code>	Library CSV and feature reference for feature barcoding
<code>--nosecondary</code>	Skip clustering and other secondary analysis
<code>--lanes, --r1-length, --r2-length</code>	Restrict lanes or hard-trim reads
<code>--localcores, --localmem, --jobmode</code>	Resources; run locally or on SGE, LSF, Slurm
<code>--dry</code>	Write the pipeline invocation and stop

cellranger multi

- One pipeline for GEX, V(D)J, Feature Barcode, multiplexed and Flex data.
- A single config CSV records libraries, references and samples in one auditable file.
- Per-sample outputs after demultiplexing (OCM, hashing, CMO, Flex).
- cellranger multi-template writes a starter CSV for your library combination.



The multi config CSV

CONFIG.CSV

```
[gene-expression]
reference,/path/to/transcriptome
create-bam,true
```

```
[feature]                # only if feature barcodes
reference,/path/to/feature_reference.csv
```

```
[libraries]
fastq_id,fastqs,feature_types
gex1,/path/to/fastqs,Gene Expression
abc1,/path/to/fastqs,Antibody Capture
```

```
[samples]                # only if multiplexed
sample_id,ocm_barcode_ids
```

- **[gene-expression]** reference, options.
- **[libraries]** one row per FASTQ set and library type.
- **[samples]** sample-to-tag mapping for demultiplexing.
- Save as .csv; keep exact headers.

Config: on-chip multiplexing

OCM CONFIG

```
[gene-expression]
reference, /path/to/transcriptome
create-bam, true

[libraries]
fastq_id, fastqs, feature_types
gex1, /path/to/fastqs, Gene Expression

[samples]
sample_id, ocm_barcode_ids
sample1, OB1
sample2, OB2
sample3, OB3
sample4, OB4
```

- OB1-OB4 map to gel bead colours.
- Technical replicates: give each its own OB and sample_id.
- Singleplex on an OCM chip still needs a [samples] section.
- Leave lanes with mock samples out.

Config: hashing and CMO

CMO CONFIG

```
[gene-expression]
reference, /path/to/transcriptome
create-bam, true

[libraries]
fastq_id, fastqs, feature_types
gex1, /path/to/fastqs, Gene Expression
mux1, /path/to/fastqs, Multiplexing Capture

[samples]
sample_id, cmo_ids
sample1, CMO301|CMO302
sample2, CMO303|CMO304
```

- **CMO:** Multiplexing Capture library + cmo_ids.
- **Hashing:** Antibody Capture library, [feature] reference, hashtag_ids.
- Several tags per sample are separated with |.

Config: Flex

FLEX CONFIG

```
[gene-expression]
reference, /path/to/transcriptome
probe-set, /path/to/probe_set.csv
create-bam, false
```

```
[libraries]
fastq_id, fastqs, feature_types
flex1, /path/to/fastqs, Gene Expression
```

```
[samples]
sample_id, probe_barcode_ids
sampleA, BC001
sampleB, BC002
```

- Probe set must match the probe kit version.
- `probe_barcode_ids` identify the Sample Barcode(s) per sample.
- Subpooled samples list several barcodes.
- Check the Flex multi page for all options (illustrative here).

Config: GEX + CRISPR

CONFIG.CSV

```
[gene-expression]
reference, /path/to/transcriptome
create-bam, true

[feature]
reference, /path/to/guides.csv

[libraries]
fastq_id, fastqs, feature_types
gex1, /path/to/fastqs, Gene Expression
crispr1, /path/to/fastqs, CRISPR Guide Capture
```

- Specify CRISPR Guide Capture as feature_types.
- The reference lists every guide.
- A crispr_analysis output folder is produced.

Config: OCM + antibody + CRISPR

CONFIG.CSV

```
[gene-expression]
reference,/path/to/transcriptome

[feature]
reference,/path/to/features.csv

[libraries]
fastq_id,fastqs,feature_types
gex1,/path/fq,Gene Expression
abc1,/path/fq,Antibody Capture
crispr1,/path/fq,CRISPR Guide Capture

[samples]
sample_id,ocm_barcode_ids
sample1,OB1
sample2,OB2
```

- features.csv holds antibody and guide rows together.
- Each OCM sample gets its own result.
- Best combined with a 5' V(D)J library if needed.

Config: hashing + antibody

CONFIG.CSV

```
[gene-expression]
reference,/path/to/transcriptome

[feature]
reference,/path/to/features.csv

[libraries]
fastq_id,fastqs,feature_types
gex1,/path/fq,Gene Expression
abc1,/path/fq,Antibody Capture

[samples]
sample_id,hashtag_ids
sampleA,HTO_A
sampleB,HTO_B
```

- Hashtag antibodies are rows of Antibody Capture in the reference.
- hashtag_ids in [samples] name the tags per sample.
- Cell Ranger v9+: feature_type Antibody Capture for hashing.

Output folder tour

TREE

```
sample_run/outs/  
├─ per_sample_outs/<sample>/  
│   ├── web_summary.html  
│   ├── metrics_summary.csv  
│   └── count/  
│       ├── sample_filtered_feature_bc_matrix/  
│       ├── sample_molecule_info.h5  
│       └── sample_cloupe.cloupe  
└─ multi/  
    └─ count/raw_feature_bc_matrix/
```

- **web_summary.html** quality report.
- **filtered matrix** cell barcodes x features, for Seurat/Scanpy.
- **.cloupe** open in Loupe Browser.
- **molecule_info.h5** input for aggr.

Cell Ranger multi outputs

-  raw_probe_bc_matrix.h5
-  raw_molecule_info.h5
-  raw_feature_bc_matrix.tar.gz
-  raw_feature_bc_matrix.h5
-  raw_cloupe.cloupe
-  qc_sample_metrics.csv
-  qc_report.html
-  qc_library_metrics.csv
-  probe_set.csv
-  multiplexing_analysis.tar.gz
-  filtered_feature_bc_matrix.tar.gz
-  filtered_feature_bc_matrix.h5
-  feature_reference.csv
-  config.csv
-  cell_types.tar.gz
-  per sample outs

OUTPUT FILES

Your output folder at a glance

Inputs

`config.csv`

`feature_reference.csv`

`probe_set.csv`

QC reports

`qc_report.html`

`qc_sample_metrics.csv`

`qc_library_metrics.csv`

Matrices and Loupe

`filtered_feature_bc_matrix.h5`

`filtered_feature_bc_matrix.tar.gz`

`raw_feature_bc_matrix.h5`

`raw_feature_bc_matrix.tar.gz`

`raw_probe_bc_matrix.h5`

`raw_molecule_info.h5`

`raw_cloupe.cloupe`

Analysis and samples

`multiplexing_analysis.tar.gz`

`cell_types.tar.gz`

`per_sample_outs/`

Inputs: how the run was set up

File	What it is	Why it matters
<code>config.csv</code>	The multi config CSV given to cellranger multi.	Lists FASTQ paths, reference, probe set, feature reference and the sample-to-barcode mapping.
<code>feature_reference.csv</code>	Antibody panel reference.	One row per target: id, name, read, pattern, barcode sequence and feature type. A wrong pattern gives zero antibody counts.
<code>probe_set.csv</code>	Flex probe set reference.	Defines which probes, and so which genes, are measured. Must match the genome reference used.

Keep these three files with the results: together they let anyone re-run the analysis.

QC: did the run work?

File	What it is	How to use it
<code>qc_report.html</code>	Interactive QC report.	Open this first, in any browser: cell counts, median genes and antibodies per cell, mapping rates, sequencing depth and demultiplexing.
<code>qc_sample_metrics.csv</code>	Metrics table, per sample.	The same kind of numbers as the report in plain text: compare samples side by side or collect many runs into one table.
<code>qc_library_metrics.csv</code>	Metrics table, per library.	Numbers for each sequenced library (gene library, antibody library): a library-level problem affects every sample in the well.

Read the report first, then use the two CSV files to check numbers sample by sample and library by library.

Matrices and Loupe: the data

File	What it contains	When to use it
<code>filtered_feature_bc_matrix.h5</code>	Gene + antibody counts for called cells only.	Default choice: load in Seurat or Scanpy.
<code>filtered_feature_bc_matrix.tar.gz</code>	Same filtered data as a 3-file bundle: <code>matrix.mtx.gz</code> , <code>features.tsv.gz</code> , <code>barcodes.tsv.gz</code> .	Tools that read the MTX format.
<code>raw_feature_bc_matrix.h5</code> / <code>.tar.gz</code>	All detected barcodes, including empty droplets and ambient RNA.	Your own cell calling or background estimates.
<code>raw_probe_bc_matrix.h5</code>	Flex only: counts per individual probe, before probes for one gene are combined.	Probe-level QC and troubleshooting.
<code>raw_molecule_info.h5</code>	Per-molecule record of every UMI, cell barcode and feature ID.	Aggregation with <code>cellranger aggr</code> , or downsampling.
<code>raw_cloupe.cloupe</code>	Loupe Browser file for the raw, unsplit data.	Open in Loupe Browser to explore without coding.

Raw vs. Filtered Matrix Comparison

The **filtered** matrix removes:

- 1.Empty Droplets:** Gel beads/droplets that contained no cells, only water/reagents.
- 2.Ambient RNA (Soup):** Low levels of cell-free RNA released by dying or lysed cells that float around in the cell suspension and get trapped in empty droplets.
- 3.Debris & Dead Cell Fragments:** Sub-cellular fragments or damaged cells with extremely low gene counts.

Analysis, per-sample results and what to open



`multiplexing_analysis.tar.gz`

Demultiplexing details: how cells were assigned to samples. Check it when sample counts look wrong.



`cell_types.tar.gz`

Automated cell-type predictions from the Cell Ranger annotation algorithm. A first look, not a final answer.



`per_sample_outs/`

One folder per sample with that sample's own results. This is the path used for cellranger aggr.

Quick start: what should you open?



Quality

`qc_report.html`



Explore

`raw_cloupe.cloupe` or a sample's `cloupe`



Analyse in R or Python

`filtered_feature_bc_matrix.h5`

Output Files

Overall outputs:

1. Web summary (HTML)
2. *_filtered_feature_bc_matrix.h5
3. Loupe Browser file (CLOUPE)

per_sample_outs:

1. web_summary.html
2. sample_filtered_feature_bc_matrix.h5
3. sample_cloupe.cloupe

Reading the web summary

Estimated cells

Close to your recovery target?

Reads per cell

Meets recommended depth?

Median genes per cell

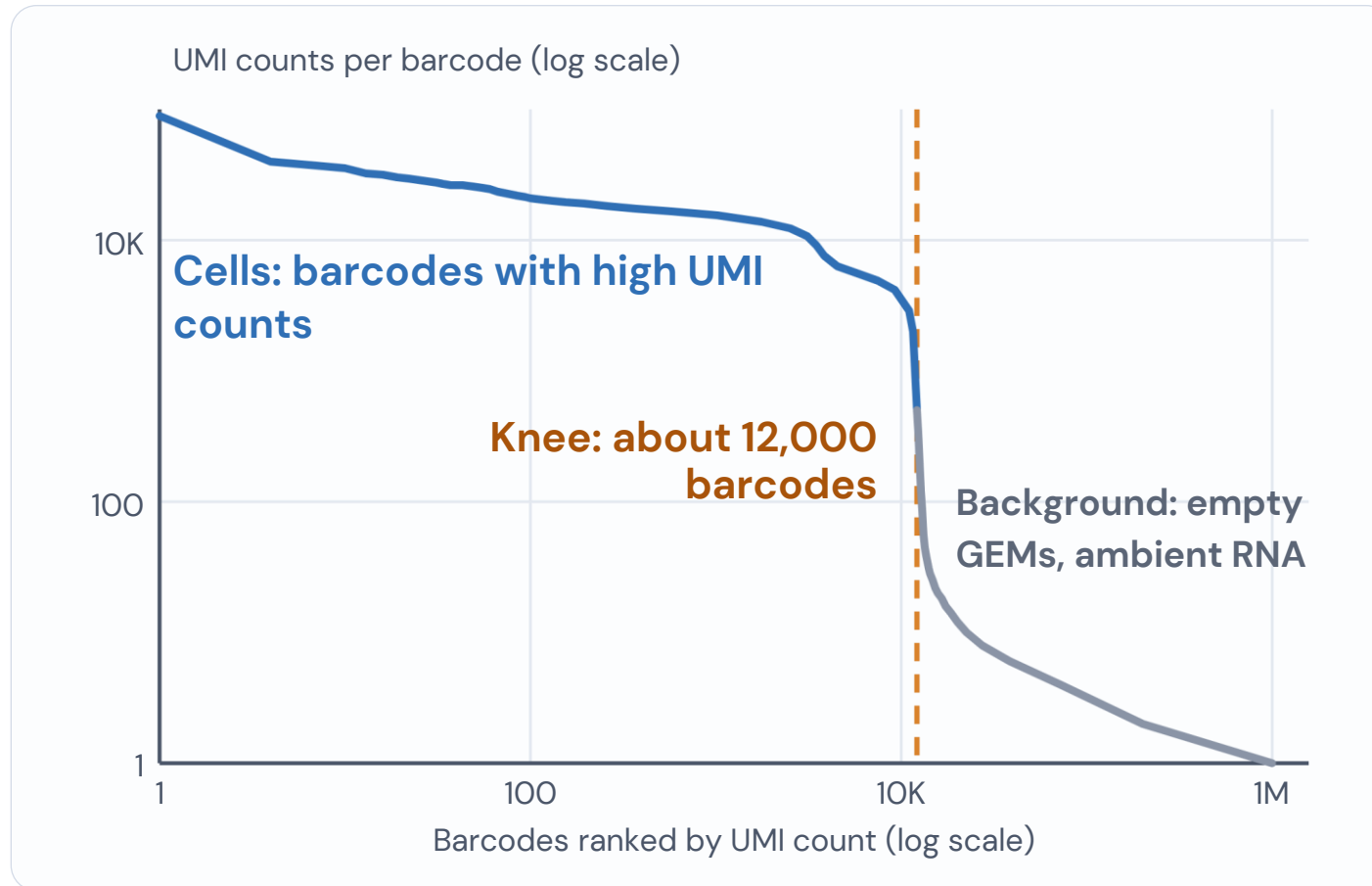
Sample-type dependent

Confidently mapped reads

Low value: wrong reference?

- **Alerts:** the summary flags issues such as low fraction reads in cells or low mapping.
- **Barcode rank plot:** a clear cliff separates cells from background.
- **Sequencing saturation:** tells you whether more depth would add information.
- **Multiplexed runs:** check per-sample cell counts and tag assignment.

GEX Barcode Rank Plot



1. Each point is a cell barcode

Barcodes are ranked from most to fewest UMIs. Both axes are log scale.

2. Cells form the plateau

GEMs holding a cell carry thousands of UMIs. Empty GEMs carry only a few.

3. The knee calls the cells

A sharp drop means clean separation. Here 12,078 cells, from 88,749 UMIs at the top down to about 500 at the knee.

Single-cell RNA-Seq Resources

Datasets created by 10x Genomics: www.10xgenomics.com/datasets

Public available single-cell RNA-Seq databases:

- NCBI GEO/SRA: www.ncbi.nlm.nih.gov/geo
- EMBL Single Cell Expression Atlas: www.ebi.ac.uk/gxa/sc/home
- Single Cell Portal: singlecell.broadinstitute.org/single_cell
- CZ Cell x Gene Discover: cellxgene.cziscience.com
- PanglaoDB: panglaodb.se
- Allen Brain Cell Atlas: portal.brain-map.org/atlas-and-data/bkp/abc-atlas

References

<https://www.10xgenomics.com>

<https://satijalab.org/seurat>

<https://scanpy-tutorials.readthedocs.io/en/latest/>

<https://cole-trapnell-lab.github.io/monocle3>

<https://github.com/theislab/scvelo>

<https://github.com/hbctraining/scRNA-seq>

<https://www.bioinformatics.babraham.ac.uk>

IMB Bioinformatics Core

<https://bc.imb.sinica.edu.tw>