Bioinformatics, 2024, **40(4)**, btae168
https://doi.org/10.1093/bioinformatics/btae168
Advance Access Publication Date: 5 April 2024
Applications Note

OXFORD

Databases and ontologies

A machine-readable specification for genomics assays

Ali Sina Booeshaghi [1,*], **Xi Chen** [2], **Lior Pachter** [3,4,*]

[1]Department of Bioengineering, University of California, Berkeley, CA, 94720, United States
[2]Department of Biology, School of Life Sciences, Southern University of Science and Technology, Shenzhen, 518055, China
[3]Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, CA, 91125, United States
[4]Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, 91125, United States

*Corresponding authors. Department of Bioengineering, University of California, University Avenue and Oxford St, Berkeley, CA, 94720, United States. E-mail: sinab@berkeley.edu (A.S.B.); Division of Biology and Biological Engineering and Department of Computing and Mathematical Sciences, California Institute of Technology, 1200 E California Blvd, Pasadena, CA, 91125, United States. E-mail: lpachter@caltech.edu (L.P.)

Associate Editor: Christina Kendziorski

Abstract

Motivation: Understanding the structure of sequenced fragments from genomics libraries is essential for accurate read preprocessing. Currently, different assays and sequencing technologies require custom scripts and programs that do not leverage the common structure of sequence elements present in genomics libraries.

Results: We present *seqspec*, a machine-readable specification for libraries produced by genomics assays that facilitates standardization of preprocessing and enables tracking and comparison of genomics assays.

Availability and implementation: The specification and associated *seqspec* command line tool is available at https://www.doi.org/10.5281/zenodo.10213865.

1 Introduction

The proliferation of genomics assays (Ogbeide *et al.* 2022) has resulted in a corresponding increase in software for processing the data (Zappia *et al.* 2018). Frequently, custom scripts must be created and tailored to the specifics of assays, where developers reimplement solutions for common preprocessing tasks such as adapter trimming, barcode identification, error correction, and read alignment (Cheow *et al.* 2016, Ma *et al.* 2020, Healey *et al.* 2022, Wu *et al.* 2022). When software tools are assay specific, parameter choices in these methods can diverge, making it difficult to perform apples-to-apples comparisons of data produced by different assays. Furthermore, the lack of preprocessing standardization makes reanalysis of published data in the context of new data challenging.

While genomics protocols can vary greatly from each other, the libraries they generate share many common elements. Typically, sequenced fragments will contain one or several ‘technical sequences’ such as barcodes and unique molecular identifiers (UMIs), as well as biological sequences that may be aligned to a genome or transcriptome. Standard library preparation kits generally require that DNA from the libraries is cut, repaired, and ligated to sequencing adapters (Fig. 1). Primers bind to the sequencing adapters, and initiate DNA sequencing whereby reads are subsequently generated. Illumina sequencing employs a sequencing by synthesis approach where fluorescently labeled nucleotides are incorporated into single-stranded DNA, and imaged, while PacBio uses zero-mode waveguides for single-molecule detection of dNTP incorporation. Oxford Nanopore on the other hand binds sequencing adapters to pores in a flow cell and DNA is sequenced by changes in electrical resistance across the pore (Iizuka *et al.* 2022).

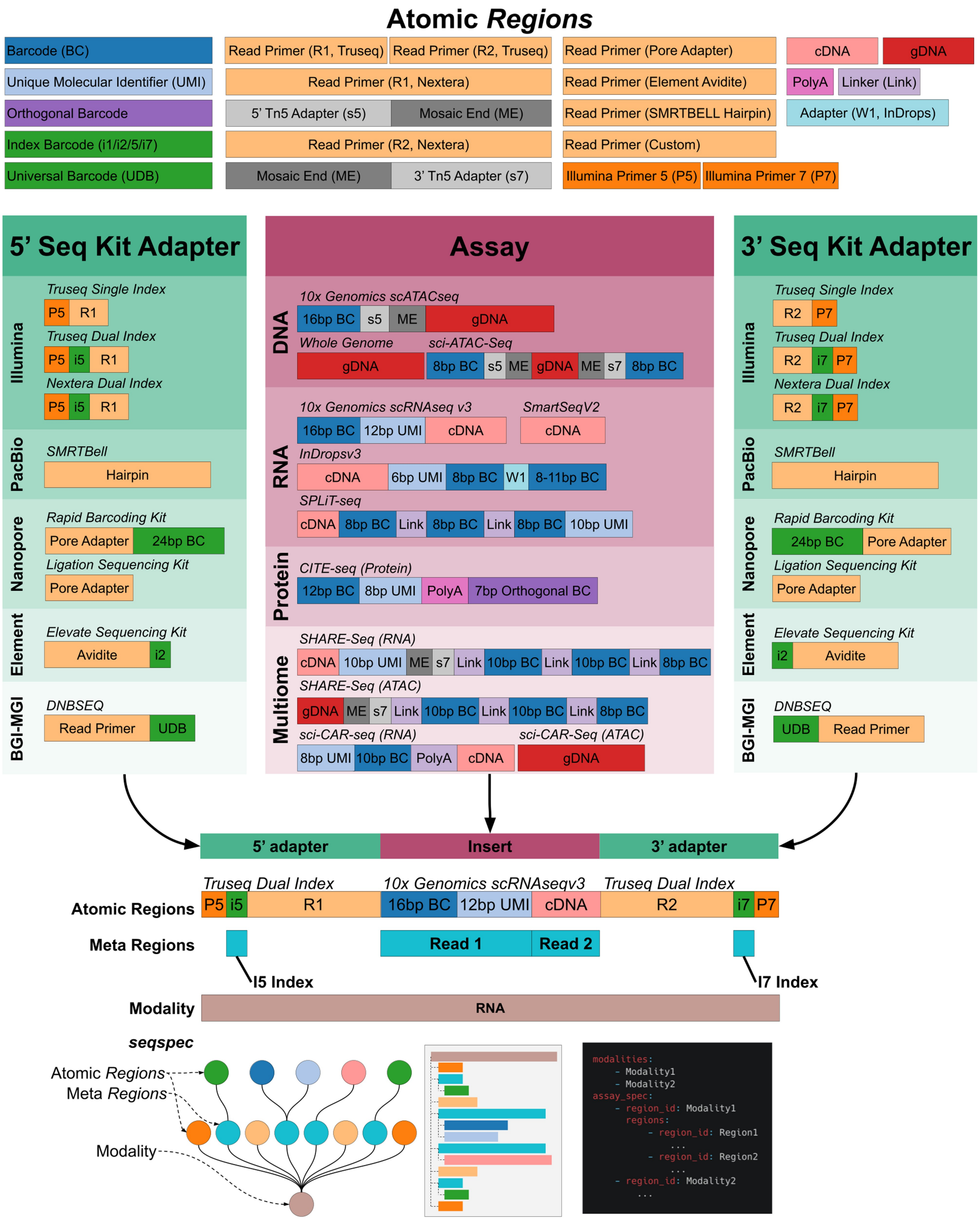
Many single-cell genomics assays introduce additional library complexity further complicating preprocessing. For example, the inDropsv3 (Klein *et al.* 2015) assay produces variable length barcodes while the 10× Genomics scRNA-seq assay (Zheng *et al.* 2017) produces fixed-length barcodes that are derived from a known list of possibilities.

Current file formats such as FASTQ, Genbank, FASTA, and workflow-specific files (Parekh *et al.* 2018) lack the flexibility to annotate sequenced libraries that contain these complex features. In the absence of sequence annotations, processing can be challenging, limiting the reuse of data that is stored in publicly accessible databases such as the Sequence Read Archive (Katz *et al.* 2022). To facilitate utilization of genomics data, a database of assays along with a description of their associated library structures was assembled in Chen (2020). While this database has proved to be very useful, the HTML descriptors are not machine readable. Moreover, the lack of a formal specification limits the utility and expandability of the database.

2 Results

The *seqspec* specification defines a machine-readable file format, based on YAML, that enables sequence library annotation. Assay- and sequencer specific molecules are annotated by *Regions* which can be nested and appended to create a

Received: 18 July 2023; **Revised:** 4 October 2023; **Editorial Decision:** 25 January 2024; **Accepted:** 4 April 2024



```
modalities:
    - Modality1
    - Modality2
assay_spec:
    - region_id: Modality1
      regions:
          - region_id: Region1
              ...
          - region_id: Region2
              ...
    - region_id: Modality2
        ...
```

Figure 1. The structure of molecules in genomic libraries. Sequencing libraries are constructed by combining Atomic *Regions* to form an adapter-insert-adapter construct. The *seqspec* for the assay annotates the construct with *Regions* and meta *Regions*.

seqspec in a manner that assumes perfect end-to-end sequencing of a perfectly constructed library. *Regions* are annotated with a variety of properties that simplify the downstream identification of library elements. The following are a list of properties that can be associated with a *Region*:

- **Region ID:** unique identifier for the *Region* in the *seqspec*.
- **Region type:** the type of region.
- **Name:** A descriptive name for the Region.
- **Sequence:** The specific nucleotide sequence for the Region.

- **Sequence type**: The type of sequence (fixed, onlist, random, joined).
- **Minimum length**: The minimum length of the sequence for the Region.
- **Maximum length**: The maximum length of the sequence for the Region.
- **Onlist**: The list of permissible sequences from which the Sequence is derived.

Importantly, *Regions*, known as meta *Regions*, can contain Regions; a property that is useful for grouping and identifying library elements that are sequenced together. The YAML format is a natural language to represent nested meta-*Regions* in a human-readable fashion. Python-style indentation and syntax can be used to create a human-readable file format without the excessive grouping delimiters of alternative languages such as JSON. In addition, nested Regions allow Assays to be represented as an Ordered Tree where the ordering of subtrees is significant: atomic Regions are 'glued' together in an order that is concordant with the design of the sequencing library in the 5′ to 3′ direction (Supplementary Fig. S1).

A key feature of *seqspec* files is that they are machine-readable, and *Region* data can be parsed, processed, and extracted with the *seqspec* command-line tool. The tool contains eleven subcommands that enable various tasks such as specification checking, finding, formatting, and indexing,

1) **seqspec check**: check the correctness of attributes against the *seqspec* schema.
2) **seqspec find**: print *Region* metadata.
3) **seqspec format**: auto populate *Region* metadata for meta Regions.
4) **seqspec index**: extract the 0-indexed position of *Regions.*
5) **seqspec info**: get info about the *seqspec* file.
6) **seqspec init**: initialize a *seqspec* with a newick-formatted string.
7) **seqspec modify**: modify *Region* attributes.
8) **seqspec onlist**: get the path to the onlist file for the specific region type.
9) **seqspec print**: print html, markdown, ascii, read diagram that visualizes the *seqspec.*
10) **seqspec split**: split *seqspec* into modalities.
11) **seqspec version**: get *seqspec* version and *seqspec* file version.

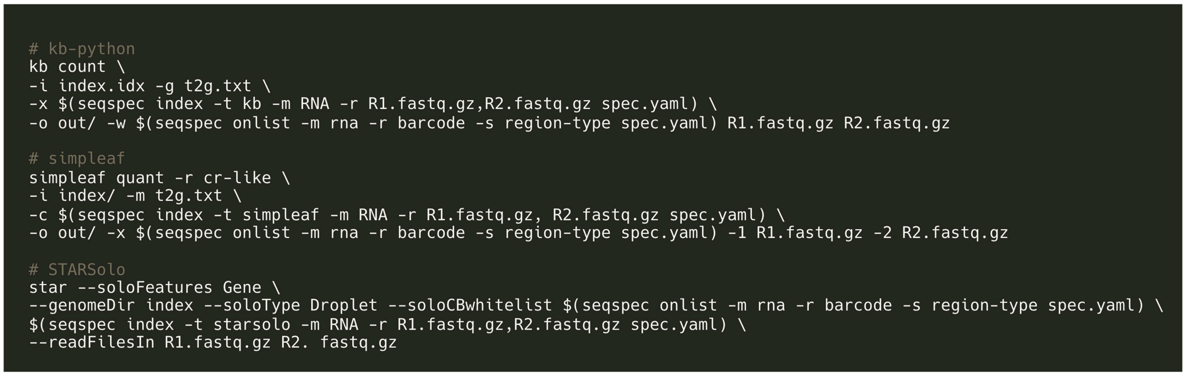
To illustrate how *seqspec* can be used to facilitate processing and analysis of single-cell RNA-seq reads, we implemented in the *seqspec index* command the facility to produce the relevant technology string for three single-cell RNA-seq preprocessing tools: *kallisto bustools* (Melsted *et al.* 2021), *simpleaf/alevin-fry* (He *et al.* 2022), and *STARsolo* (Kaminow *et al.* 2021) (Fig. 2). *Regions* associated with barcodes, UMIs, and cDNA are extracted, positionally indexed and formatted on a per-tool basis. The modularity of *seqspec* makes it simple to produce tool-compatible technology strings for other assay types.

3 Discussion

The *seqspec* specification and associated tool enable the annotation of a sequence library that has been generated by an assay to be processed with a sequencer-specific kit for sequencing. Associating *seqspecs* with sequencing data can greatly facilitate reprocessing and interpretation. For example, *seqspec* can help in investigating differences between sequence reads and library structure, aiding in the study of sequencing artifacts. In terms of facilitating preprocessing, *seqspec* innovates beyond existing methods such as *kb-python*'s technology string or the read geometry string in *simpleaf* (He *et al.* 2022) by providing both annotation of library structures as well as a suite of tools for format validation.

Standardized annotation of sequencing libraries in a human- and machine-readable format serves several purposes including the enablement of uniform processing, organization of sequencing assays by constitutive components, and transparency for users. The flexibility of *seqspec* should allow it to be used for all current sequence census assays (Wold and Myers 2008), and specifications should be readily adaptable to different sequencing platforms; our initial release of *seqspec* contains specifications for 49 assays (see https://igvf.github.io/seqspec/). In the future, we envision that *seqspec* could be extended to describe sequencer- or protocol-specific steps as well as utilized to annotate engineered sequences such as DNA constructs.

Comparison of *seqspecs* for different assays, immediately reveals shared similarities and differences that can be visualized with *seqspec print*. For example, the SPLiT-seq single-cell RNA and the multimodal SHARE-seq single-cell assays are aimed at different modalities and utilize different protocols to produce libraries, but the resultant structures are very similar (Fig. 1) since they both rely on split-pool barcoding (Rosenberg *et al.* 2018). The *seqspec* for the sci-CAR-seq assay (Cao *et al.* 2018), from which split-pool assays such as SHARE-seq are derived, shows that the cell barcoding is encoded in the Illumina indices. It should be possible to



```
# kb-python
kb count \
-i index.idx -g t2g.txt \
-x $(seqspec index -t kb -m RNA -r R1.fastq.gz,R2.fastq.gz spec.yaml) \
-o out/ -w $(seqspec onlist -m rna -r barcode -s region-type spec.yaml) R1.fastq.gz R2.fastq.gz

# simpleaf
simpleaf quant -r cr-like \
-i index/ -m t2g.txt \
-c $(seqspec index -t simpleaf -m RNA -r R1.fastq.gz, R2.fastq.gz spec.yaml) \
-o out/ -x $(seqspec onlist -m rna -r barcode -s region-type spec.yaml) -1 R1.fastq.gz -2 R2.fastq.gz

# STARSolo
star --soloFeatures Gene \
--genomeDir index --soloType Droplet --soloCBwhitelist $(seqspec onlist -m rna -r barcode -s region-type spec.yaml) \
$(seqspec index -t starsolo -m RNA -r R1.fastq.gz,R2.fastq.gz spec.yaml) \
--readFilesIn R1.fastq.gz R2. fastq.gz
```

Figure 2. Uniform processing enabled with *seqspec*. The *seqspec index* command produces a technology string that identifies appropriate sequence elements and can be passed into processing tools.

develop an ontology of assays by comparing the *seqspec* specifications of assays and quantifying their similarities and differences.

In demonstrating that *seqspec* can be used to define options for preprocessing tools, we have shown that *seqspec* is immediately useful for uniform processing of genomics data. The preprocessing applications will hopefully incentivize data generators to define and deposit *seqspec* files alongside sequencing reads in public archives such as the Sequence Read Archive. While *seqspec* is not a suitable format for general metadata storage, the precise specification of sequence elements present in reads, including sequencer-specific constructs, should be helpful in identifying batch effects even when metadata is missing or inaccurate.

Acknowledgements

We thank Delaney Sullivan for helpful discussions and Rahma Elsiesy for helpful feedback on Fig. 1. Discussions with the Impact of Genomics Variation on Function (IGVF) Single-Cell Focus Group helped to shape some features of *seqspec*. Thanks to Idan Gabdank for useful feedback on *seqspec* and for suggesting the md5 checksum. Meichen Fang contributed the sci-RNA-seq3 seqspec. This work was primarily undertaken while A.S.B. was at the California Institute of Technology.

Supplementary data

Supplementary data are available at *Bioinformatics* online.

Conflict of interest

None declared.

Funding

This work was supported in part by NIH [5UM1HG012077-02 to A.S.B. and L.P.]. The authors also acknowledge the Howard Hughes Medical Institute for funding A.S.B. through the Hanna H. Gray Fellows program.

Data availability

The specification and associated seqspec command line tool is available at https://www.doi.org/10.5281/zenodo.10213865 as well as on GitHub https://github.com/pachterlab/seqspec.

References

Cao J, Cusanovich DA, Ramani V *et al.* Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science* 2018;**361**:1380–5.

Chen X. Collections of library structure and sequence of popular single cell genomic methods. GitHub. 2020. https://github.com/Teichlab/scg_lib_structs.

Cheow LF, Courtois ET, Tan Y *et al.* Single-cell multimodal profiling reveals cellular epigenetic heterogeneity. *Nat Methods* 2016; **13**:833–6.

He D, Zakeri M, Sarkar H *et al.* Alevin-fry unlocks rapid, accurate and memory-frugal quantification of single-cell RNA-seq data. *Nat Methods* 2022;**19**:316–22.

Healey HM, Bassham S, Cresko WA. Single-cell iso-sequencing enables rapid genome annotation for scRNAseq analysis. *Genetics* 2022; **220**. https://academic.oup.com/genetics/article/220/3/iyac017/6526397.

Iizuka R, Yamazaki H, Uemura S. Zero-mode waveguides and nanopore-based sequencing technologies accelerate single-molecule studies. *Biophys Physicobiol* 2022;**19**:e190032.

Kaminow B, Yunusov D, Dobin A. Starsolo: accurate, fast and versatile mapping/quantification of single-cell and single-nucleus RNA-seq data. bioRxiv, https://doi.org/10.1101/2021.05.05.442755, 2021, preprint: not peer reviewed.

Katz K, Shutov O, Lapoint R *et al.* The sequence read archive: a decade more of explosive growth. *Nucleic Acids Res* 2022;**50**:D387–90.

Klein AM, Mazutis L, Akartuna I *et al.* Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* 2015; **161**:1187–201.

Ma S, Zhang B, LaFave LM *et al.* Chromatin potential identified by shared single-cell profiling of rna and chromatin. *Cell* 2020;**183**: 1103–16.e20.

Melsted P, Booeshaghi AS, Liu L *et al.* Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nat Biotechnol* 2021; **39**:813–8.

Ogbeide S, Giannese F, Mincarelli L *et al.* Into the multiverse: advances in single-cell multiomic profiling. *Trends Genet* 2022;**38**:831–43.

Parekh S, Ziegenhain C, Vieth B *et al.* zumis – a fast and flexible pipeline to process RNA sequencing data with umis. *Gigascience* 2018;**7**. https://academic.oup.com/gigascience/article/7/6/giy059/5005022?login=false.

Rosenberg AB, Roco CM, Muscat RA *et al.* Single-cell profiling of the developing mouse brain and spinal cord with split-Pool barcoding. *Science* 2018;**360**:176–82.

Wold B, Myers RM. Sequence census methods for functional genomics. *Nat Methods* 2008;**5**:19–21.

Wu H, Li X, Jian F *et al.* Highly sensitive single-cell chromatin accessibility assay and transcriptome coassay with metatac. *Proc Natl Acad Sci USA* 2022;**119**:e2206450119.

Zappia L, Phipson B, Oshlack A. Exploring the single-cell RNA-seq analysis landscape with the scRNA-tools database. *PLoS Comput Biol* 2018;**14**:e1006245.

Zheng GXY, Terry JM, Belgrader P *et al.* Massively parallel digital transcriptional profiling of single cells. *Nat Commun* 2017; **8**:14049.

Bioinformatics, 2024, 40, 1–4
https://doi.org/10.1093/bioinformatics/btae168
Applications Note