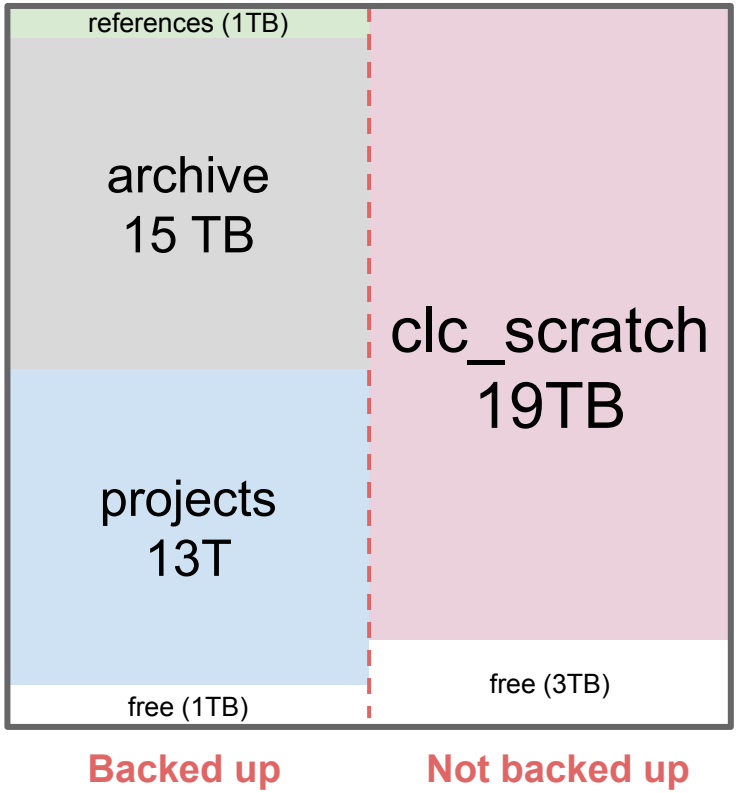
The GSC space for LCR bioinformatics

2021/03/01

/projects/clc/



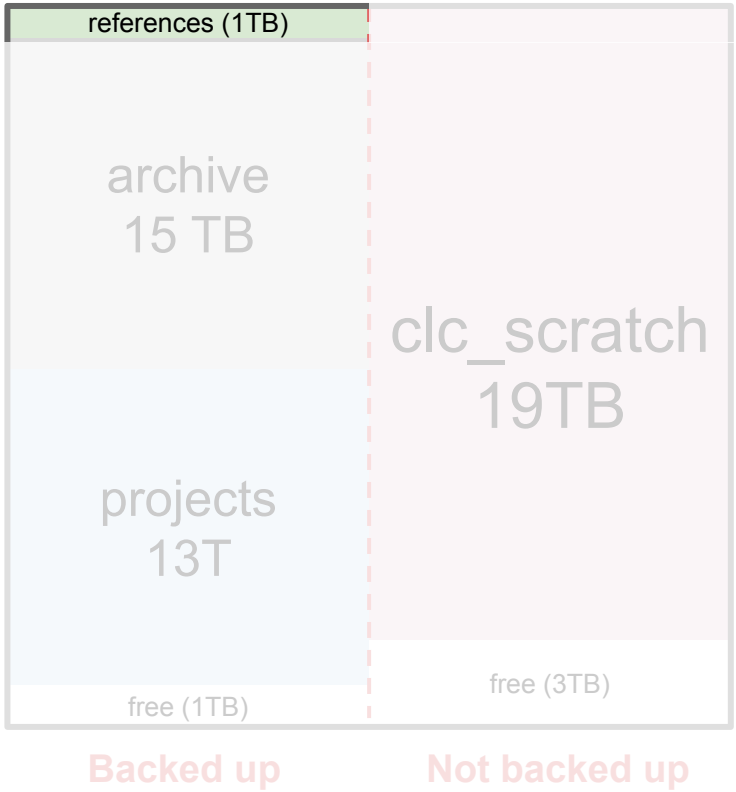
Usage:

- References
- Raw data
- Published projects
- Current projects
  - Final results
  - Scripts
- Binaries

Usage:

- Pipeline development
- Current projects
  - Testing scripts
  - Ongoing analyses, intermediate files
- Temporary datasets (e.g. EGA encryption)

/projects/clc/



References

RNAseq

- bowtie
- cellranger
- kallisto
- star
- Trinity_CTAT

Annotation

- COSMIC
- dbSNP
- SnpEff
- 1000 genomes
- VEP

Other

- GATK
- Genomes
- Exomes
- Indels
- LCR modules

GRCh38 vs. hg38?

The latest build of the human reference genome is officially named **GRCh38** (for Genome Research Consortium human build 38) but commonly nicknamed **hg38** (for Human genome build 38)

Released December 2013

GRCh38 highlights

The GRCh38 assembly provides four significant improvements over GRCh37 and other earlier versions:

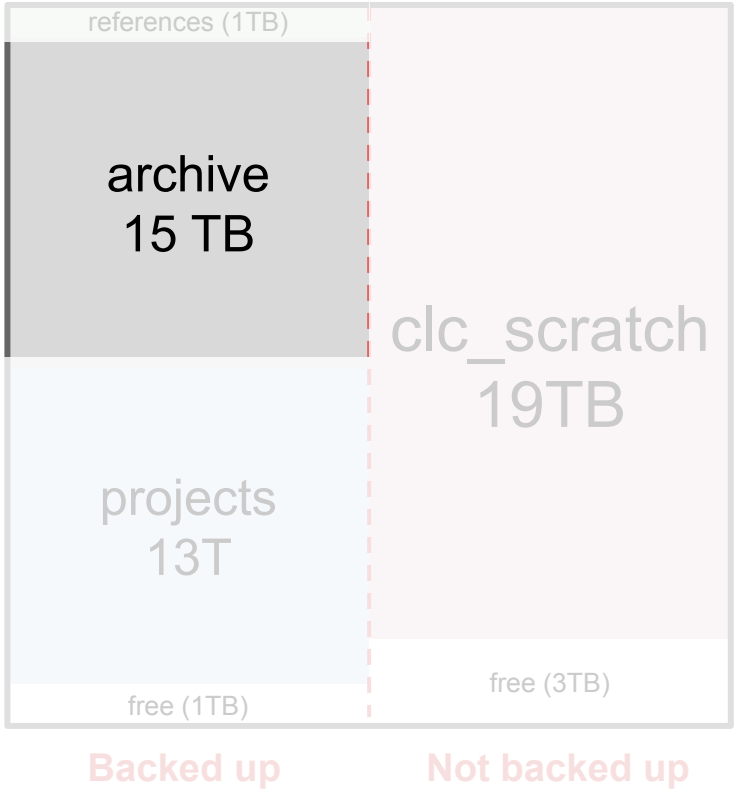
1. Inclusion of the mitochondrial genome
2. Sequence coverage of centromeres
3. General assembly updates - correction of thousands of small sequencing artifacts that cause false SNPs/indels to be called in previous versions
4. Better representation of variation (expanded repertoire of ALT contigs)

The latest human reference genome

- Hg38 is the most accurately sequenced version of the human genome
- Produced using Sanger Sequencing - reads as long as 1000 nt - **10X more accurate than high-throughput short-read sequencing**
- 8000 altered nt, correction of many misassembled regions, filled in gaps, added sequence for centromeres, and substantially improved diversity of reference by including 261 ALT loci over 178 regions
- **27% increase in exome size from to hg19**
  - Increase in total # exons: 327,058 (hg19) → 457,748 (hg38)
  - Increase in median # exons / gene: 13 (hg19) → 19 (hg38)
  - Increase in median # nt / exon: 140 (hg19) → 146 (hg38)
- Fewer SNVs and indels identified in hg38 → fewer false positives

Use the latest reference at the *beginning* of a project (not in the middle)

/projects/clc/



Archive

- Raw data that we want to keep indefinitely
- Organized by research ID (resID) / cell line name / external collaborator
  - E.g. 06-28798, OCI-Ly10, AW_collab
  - Raw analysis file (e.g. fastq, bam, cel)
- MiSeq and NextSeq
  - Full machine output

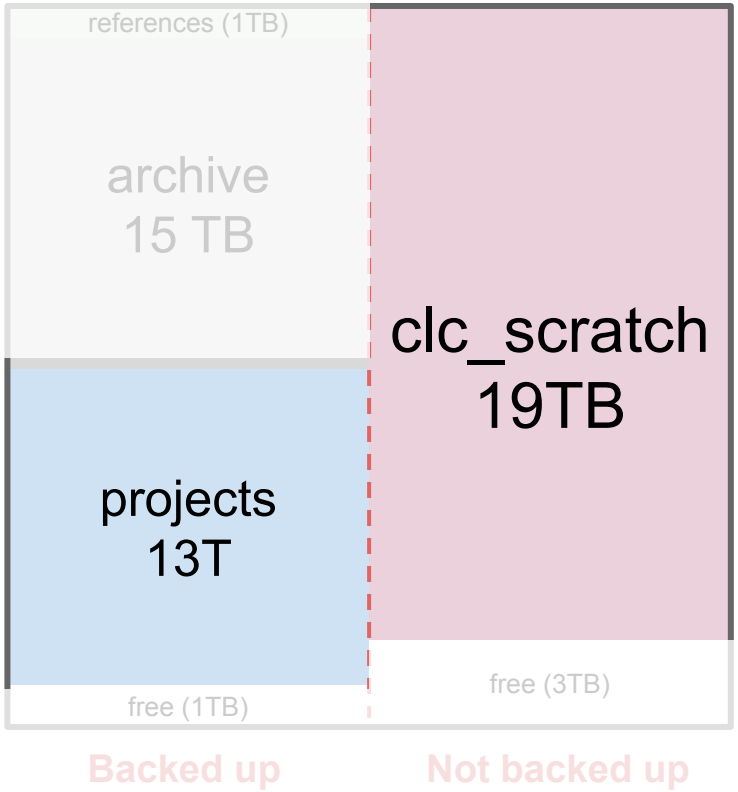
Archive - hierarchical structure

- Res ID
  - Technology
    - Library ID
      - Analysis type
        - Result type
          - Result file

```
03-25766
├── SureSelect
│   ├── A43115
│   │   └── bwaAligned
│   │       └── bam
│   │           ├── convert_bam_to_cram_IX2883_C4CY1ACXX_8_CGGAAT.sh
│   │           └── IX2883_C4CY1ACXX_8_CGGAAT.cram
│   └── PA075
│       └── bwa_mem
│           └── bam
│               ├── 03-25766_SureSelect_PA075.bam.bai
│               └── 03-25766_SureSelect_PA075.cram
└── TSCA
    ├── T0325766FF
    │   └── bowtie
    │       └── bam
    │           ├── convert_bam_to_cram_T0325766FF.bowtie.sorted.filtered.sh
    │           └── T0325766FF.bowtie.sorted.filtered.cram
    └── T0325766FFPET
        └── bowtie
            └── bam
                ├── convert_bam_to_cram_T0325766FFPET.bowtie.sorted.filtered.sh
                └── T0325766FFPET.bowtie.sorted.filtered.cram
```

Technologies:
WGSS, WES, SureSelect, TwoStep, TSCA, SNP6, WTSS, Chromium, ...

/projects/clc/



Projects

- /projects/clc/projects and /projects/clc/clc_scratch/projects

- Organized by user

```
[shung@gphost01 projects]$ ls -la
total 100
drwxrwsr-x 20 fcchan    clc 8192 Feb 26  2020 .
drwxrwsr-x 13 fcchan    clc 4096 May 12  2020 ..
drwxr-sr-x  2 bcollinge clc 4096 May 26  2020 bcollinge
drwx--S---  2 gduns     clc 4096 Jun 15  2020 bin
drwxrwsr-x  3 shung     clc 4096 Sep 26  2016 chother
drwx--S---  2 gduns     clc 4096 Jun 15  2020 config
drwxrwsr-x  8 fcchan    clc 4096 Jul 10  2017 fcchan
drwxrwsr-x 10 shung     clc 4096 Feb 25  2018 fkhan
drwxrwsr-x 15 lchong    clc 8192 Feb  5 17:39 gduns
-rw-r--r--  1 hwinata   clc  139 May 11  2020 hele.code-workspace
drwxrwx--- 13 hshulha   clc 4096 Aug  2  2018 hshulha
drwxr-sr-x  6 hwinata   clc 4096 Oct 26 10:48 hwinata
-rwxr-xr-x  1 gduns     clc  269 Feb 24  2020 launch_snakemake.sh
drwxrwsr-x 45 shung     clc 4096 Oct 13 08:53 lchong
drwx--S---  2 gduns     clc 4096 Jun 15  2020 linx
drwx--S---  2 gduns     clc 4096 Jun 15  2020 linx2
drwxr-sr-x  3 mton      clc 4096 Apr  1  2020 mton
drwxrwsr-x  7 fcchan    clc 4096 Jan 23  2018 raymondl
drwx--S---  2 gduns     clc 4096 Jun 15  2020 reference
drwxrwsr-x 66 shung     clc 4096 Feb  4 14:56 shung
-rw-r--r--  1 gduns     clc 1850 Feb 26  2020 Snakefile
drwxr-sr-x 14 gduns     clc 4096 Feb 26  2020 .snakemake
drwxrwsr-x  3 shung     clc 4096 Aug  2  2017 tboyarski
```

Maintenance of space

Regularly checking of:

1. Improper space usage and organization
2. Data files that can be compressed
3. Data files that can be deleted
4. Adequate space availability

Improper space usage and organization

```
[shung@gphost01 projects]$ ls -la
total 100
drwxrwsr-x 20 fcchan    clc 8192 Feb 26  2020 .
drwxrwsr-x 13 fcchan    clc 4096 May 12  2020 ..
drwxr-sr-x  2 bcollinge clc 4096 May 26  2020 bcollinge
drwx--S---  2 gduns     clc 4096 Jun 15  2020 bin
drwxrwsr-x  3 shung     clc 4096 Sep 26  2016 chother
drwx--S---  2 gduns     clc 4096 Jun 15  2020 config
drwxrwsr-x  8 fcchan    clc 4096 Jul 10  2017 fcchan
drwxrwsr-x 10 shung     clc 4096 Feb 25  2018 fkhan
drwxrwsr-x 15 lchong    clc 8192 Feb  5 17:39 gduns
-rw-r--r--  1 hwinata   clc  139 May 11  2020 hele.code-workspace
drwxrwx--- 13 hshulha   clc 4096 Aug  2  2018 hshulha
drwxr-sr-x  6 hwinata   clc 4096 Oct 26 10:48 hwinata
-rwxr-xr-x  1 gduns     clc  269 Feb 24  2020 launch_snakemake.sh
drwxrwsr-x 45 shung     clc 4096 Oct 13 08:53 lchong
drwx--S---  2 gduns     clc 4096 Jun 15  2020 linx
drwx--S---  2 gduns     clc 4096 Jun 15  2020 linx2
drwxr-sr-x  3 mton      clc 4096 Apr  1  2020 mton
drwxrwsr-x  7 fcchan    clc 4096 Jan 23  2018 raymondl
drwx--S---  2 gduns     clc 4096 Jun 15  2020 reference
drwxrwsr-x 66 shung     clc 4096 Feb  4 14:56 shung
-rw-r--r--  1 gduns     clc 1850 Feb 26  2020 Snakefile
drwxr-sr-x 14 gduns     clc 4096 Feb 26  2020 .snakemake
drwxrwsr-x  3 shung     clc 4096 Aug  2  2017 tboyarski
```

Potential space sink

Compression of files

https://www.bcgsc.ca/wiki/pages/viewpage.action?spaceKey=LBTD&title=Fastq+and+Bam+Compression+software+at+the+GSC

Spaces People Create ··· Search

Lab / BioInf TechD

Pages
Blog

SPACE SHORTCUTS
File lists
Product requirements
How-to articles

PAGE TREE
BioInf Meeting Minutes
Bioinformatics Process Developme
BioInf Tool Testing
16S identification tools
AmpliconVariantCalling2
Amplicon Variant calling
BAM compression testing
Comparison of CRAM and SPE
CRAM
Fastq and Bam Compression
Spiral Genetics / CRAM / Scrar
centos 6 tool tests
ChimeraScan
Copy Number on Tumours
deFuse Transcriptome Structura
DELLY
Differential Expression
Fastq and folder compression to

Dashboard / ... / BAM compression testing

Edit Save for later Watch Share ···

Fastq and Bam Compression software at the GSC

Created by Richard Corbett, last modified on Mar 27, 2019

Fastq compression / decompression

DSRC is about 30% more space efficient than GZIP. It is also significantly faster.

NOTE: DSRC will use as many threads as it sees fit - Use -t to limit

```
# To compress an unzipped fastq
/gsc/software/linux-x86_64-centos6/dsrc-2.0.2/dsrc c your.fastq your.fastq.dsrc

# To un-dsrc
/gsc/software/linux-x86_64-centos6/dsrc-2.0.2/dsrc d your.fastq.dsrc your.fastq

# To convert gzip to dsrc
zcat your_fastq.gz | /gsc/software/linux-x86_64-centos6/dsrc-2.0.2/dsrc c -s your_fastq.dsrc

# To convert gzip to dsrc with multi-thread (could use pigz-2.3.4):
pigz -p 8 -d -k -c your_fastq.gz | /gsc/software/linux-x86_64-centos6/dsrc-2.0.2/dsrc c -s -t8 your_fastq.dsrc
```

Compression can lead to up to 50% in space savings

BAM compression / decompression - UPDATED MARCH 2019

Storing your BAM in CRAM format will reduce the storage requirement for your alignments by up to 50%. Additionally, many analysis tools can now work directly on CRAM files which can remove the requirement to decompress the data when regenerating results.

These commands as written will preserve the presence or absence of MD and NM tags in your original BAM. Without adding the parameters below, the MD and NM tags will be calculated during decompression and may differ slightly from the original.

```
# To compress a BAM and preserve MD, NM tags, using 16 threads
/gsc/software/linux-x86_64-centos7/samtools-1.9/bin/samtools view -@ 16 -O CRAM,store_md=1,store_nm=1 -T genome.fa your.bam -o your.cram


# To decompress
/gsc/software/linux-x86_64-centos7/samtools-1.9/bin/samtools view -@ 16 --input-fmt-option decode_md=0 -O bam -T genome.fa the.cram -o the.cram.bam
```

Running analyses on the GSC

INTERACTIVE NODES

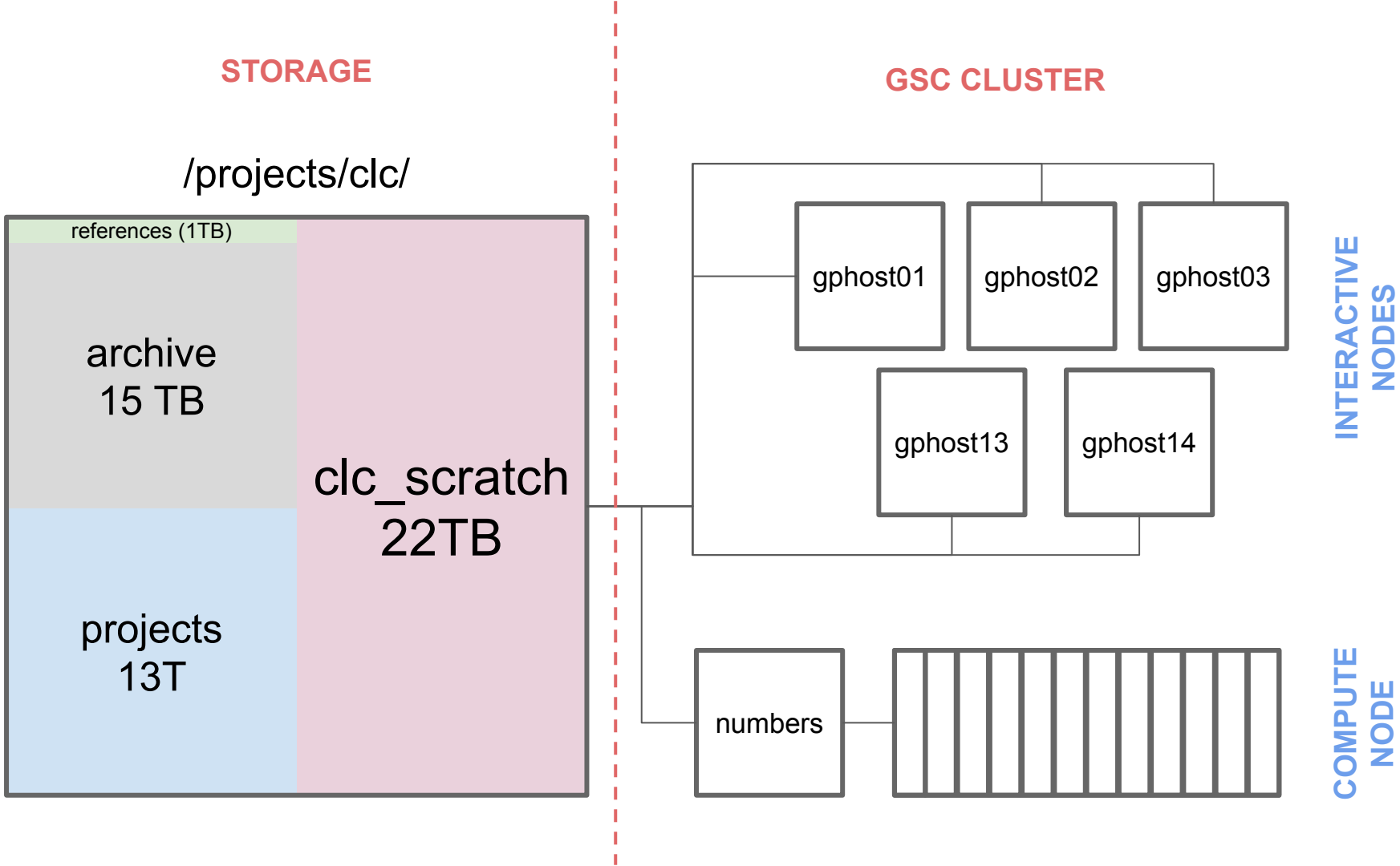
Usage:

- Editing
- Short / small jobs (e.g. compiling code, quick and small test runs, etc.)

COMPUTE NODE

Usage:

- Long / large / computationally intensive jobs


STORAGE
GSC CLUSTER
/projects/clc/
references (1TB)
archive
15 TB
projects
13T
clc_scratch
22TB
gphost01
gphost02
gphost03
gphost13
gphost14
INTERACTIVE NODES
numbers
COMPUTE NODE