INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC 1/SC 29/WG 11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC 1/SC 29/WG 11 N15346
Geneva, CH – February 2015

Source **Requirements**

Status **Updated Draft**

Title **Investigation on genomic information compression and storage**

Authors Claudio Alberti, Marco Mattavelli, Ana Hernandez (EPFL), Leonardo Chiariglione (CEDEO), Ioannis Xenarios, Nicolas Guex, Heinz Stockinger, Thierry Schuepbach, Pascal Kahlem, Christian Iseli, Daniel Zerzion, Dmitry Kuznetsov (SIB), Yann Thoma, Enrico Petraglio (HEIG-VD), Jaime Delgado (UPC)

Table of Contents

Executive Summary

The sequencing of the genetic information of human genome has become affordable due to high-throughput sequencing technology [1], [2]. This opens new perspectives for the diagnosis and successful treatment of cancer and other genetic illnesses. However, there remain challenges, scientific as well as computational, that need to be addressed for this technology to find its way into everyday practice in healthcare and medicine. The first challenge is to cope with the flood of sequencing data. For instance, a database covering the inhabitants of a small country like Switzerland would need to store a staggering amount of data, about 2'335'740 Terabytes. The second challenge is the ability to process such a deluge of data in order to 1) increase the scientific knowledge of genome sequence information and 2) search genome databases for diagnosis and therapy purposes. High-performance compression of genomic data is required to reduce the storage size, increase transmission speed and reduce the cost of I/O bandwidth connecting the database and the processing facilities.

The current trends in sequencing data generation show clearly that the storage and transfer (bandwidth) costs will soon become comparable to the costs of sequencing. This means that IT costs may soon become a major obstacle to such genome analysis applications as personalized medicine, early diagnostics and drugs discovery, unless genetic data compression reduces IT costs on par with sequencing costs.

This document has been drafted with the goal to help MPEG to assess the opportunity to start a standardization effort in genetic information processing, particularly compression, and provides

1. An overview of the current status of tools and technology supporting genomic information compression and storage
2. An analysis of related challenges for the stakeholders
3. A review of the existing compression tools and techniques.

1 Terminology

Term	Definition
Alignment	A sequence read mapped on a reference DNA sequence
BAM	Compressed binary version of SAM
CIGAR string	A CIGAR string is a sequence of base lengths and the associated operation used to indicate things like which bases align (either a match/mismatch) with the reference, are deleted from the reference, and are insertions that are not in the reference.
CRAM	GIR that includes SAM + Compression configuration
FastA	GIR that includes header and sequence reads (nucleotides sequence)
FastQ	GIR that includes FastA + Quality Scores
GIR	Genomic Information Representation
Indel	An additional or missing nucleotide in a DNA sequence with respect to a reference DNA sequence.
MAF	Mutation Annotation Format. File format used to mark the genes and other biological features in a DNA sequence
Mate pairs	Two reads from the same (long) DNA strand extracted by sequencing machines. The orientation is the opposite of paired ends.
Paired ends	Couple of reads produced by the same (short) DNA fragment by sequencing both ends. The orientation is the opposite of mate pairs.
Quality score	A quality score is assigned to each nucleotide base call in automated sequencing processes. It expresses the base-call accuracy.
Read header	Each sequence read stored in FastA and FastQ format starts with a textual field called “header” containing a sequence identifier and an optional description
SAM	GIR that is human readable and includes FastQ + Alignment and analysis information
Sequence read	The readout, by a specific technology more or less prone to errors, of a continuous part of a segment of DNA extracted from an organic sample

2 Genomic information generation and manipulation

Figure 1 shows the main stages of genomic information manipulation in existing bioinformatics applications. The steps depicted include:

1. Sequencing: expression of genomic information as strings (a.k.a. sequences or reads) of nucleotides identifiers.
2. Alignment/mapping: sequences arrangement to identify regions of similarity among them (*de-novo* assembly) or with respect to an external reference (a pre-constructed genome). Sequences are encoded in the form of SAM files and its binary dual named BAM [3].
3. Compression: data encoding to use less bit.
4. Storage: compressed data is stored and made available via database interfaces or files.
5. Decompression/access: access to data to perform analysis.
6. Update: previously sequenced genomic information might be updated by means of new alignment techniques or new sequencing (a.k.a. re-sequencing).

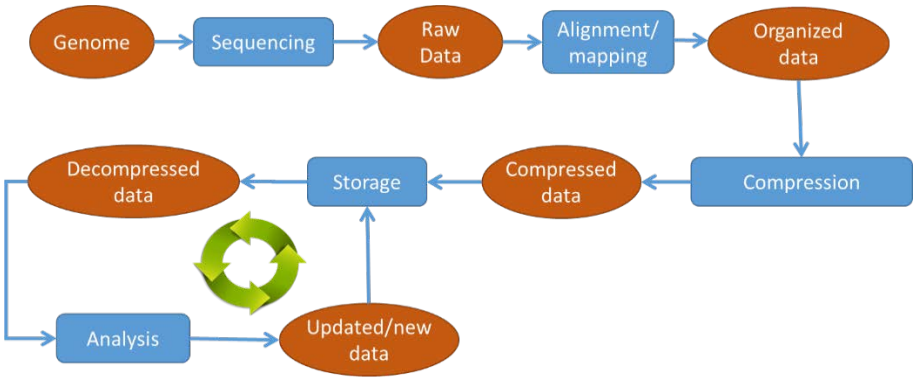


Figure 1 – Genomic information generation and manipulation stages

2.1 Genome versus sequence data compression

One important distinction that is worth stressing here is the difference between entire *genome compression* and *sequence data compression*.

Genome compression tools aim at encoding the genetic information of a living organism expressed as a sequence of symbols representing the nucleotides. This string is about 3.2 billion symbols long for the human being (organized in 23 chromosomes) and can be up to 100+ billion symbols long for other organisms. The encoding of an entire genome is the result of a long (error prone) process of analysis that today can only provide a close approximation to the real genetic sequence.

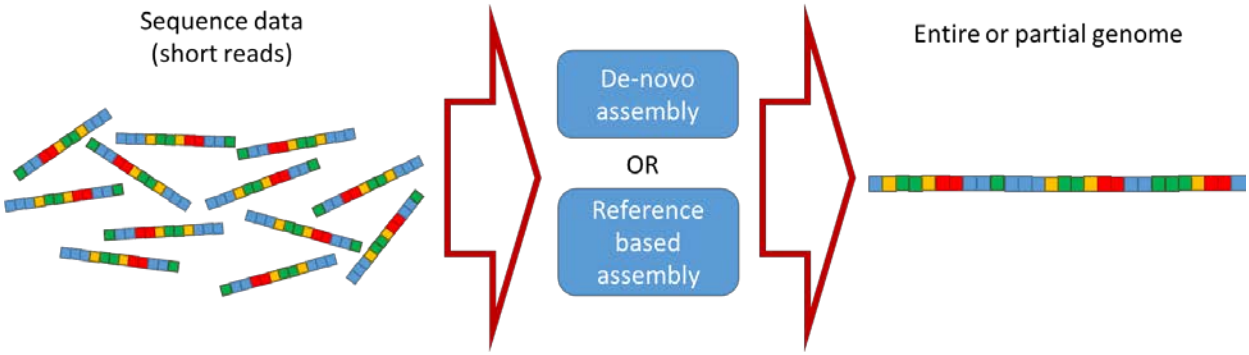


Figure 2 – From short reads to genome

On the other hand the *compression of sequence reads* focuses on encoding the output of new generation sequencing machines which are able to extract large amounts of short (from 35 to over 1,000) nucleotides sequences ("reads"). This is the type of information that is nowadays in need of efficient compression in order to enable the wide range of applications made possible by recent advances and discoveries in genomics. This document will focus on compression of sequence data (short reads).

2.2 DNA sequencing

The expression "Next Generation Sequencing" (NGS) designates the process of fast extraction of large amounts of genomic information from samples of organic material belonging to a living organism. Modern sequencing devices are able to produce several hundred million "reads" (genomic information related to pieces of the whole genome) per day. According to the specific technology employed, each read can contain from a few dozens to several thousand bases (the atomic unit of genomic information) together with optional metadata. This rate of information generation dramatically outpaces any progress in digital information storage and transmission. In this context, the last decade has witnessed several attempts at finding suitable solutions to compress genomic information efficiently and robustly. These efforts have been produced by

research institutions, universities, industries with a wide range of diverse priorities and drivers. The result is a proliferation of tools and formats able to address only the specific needs of their authors, without any perspective to be flexible enough to meet the various needs of the scientific and industrial communities as a whole.

As a consequence, nowadays most of the players of the "genomic revolution" are open to initiatives aiming at making the growing amount of genomic information more manageable and rapidly "consumable" by the tools used for analysis.

For instance the human genome is composed by a sequence of about 3 billion nucleotide bases. Research projects can produce with just one sequencing analysis, a volume of data (in the form of relatively small fragments of the genome) that reaches up to 400-500 times the size of the complete human genome. Faster sequencing technology produce a much higher volume of data with a much higher redundancy which requires much more efficient (in terms of both size and processing speed) compression than the current simple and non-standard methods available today.

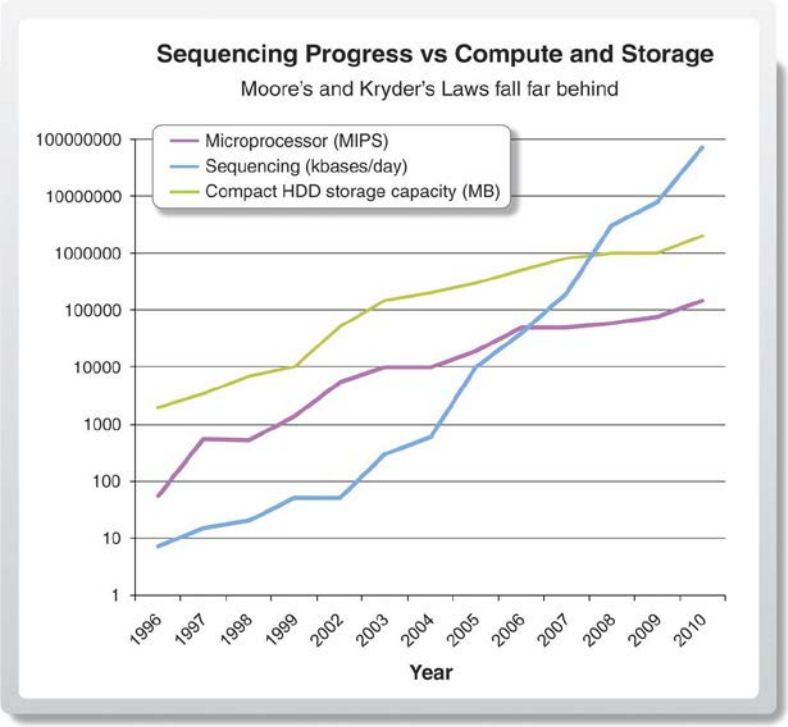


Figure 3 Moore's law versus Sequencing, from [1]

What is important to be remarked is that within such huge amount of data, even if most of the fragments can be considered redundant versus the theoretical size of the genome, these cannot be simply discarded, because on one side it is the statistical indication of the correctness of the reads and on the other side small differences in some fragments might indicate pathologies that might be appropriately taken care of.

2.2.1 The NGS Industry Landscape

A proliferation of new sequencing technology is rapidly spreading across the market of NGS machines. Illumina, Life Technologies (Thermo Fisher Scientific), 454 Lifesciences (acquired by Roche, but recently dropped in favour of nanopore technology) and Pacific Biosciences are companies that commercialize equipment relying on different sequencing methods briefly described below. New sequencing technology are proliferating as well, such as nanopore sequencing which seems to be promising in terms of reads length and cost of sequencing, but it is still far from the accuracy of commercial solutions.

The main difference among the sequencing devices commercialized by these companies is the technology employed to process the organic material to determine the precise order of nucleotides within DNA strands. The different methods result in extremely different segments lengths and performance in terms of speed, accuracy, cost and throughput.
Table 1 is taken from Wikipedia and integrated with an entry on nanopore technology. It compares the methods employed by the 4 players mentioned above plus the emerging technology based on nanopores.
One key difference among the methods is the length of the extracted reads. Two are the main classes of sequencing methods: the largest class includes those able to extract short reads (in the order of a few hundreds base pairs long), while other methods (e.g. from Pacific Bioscience) are able to extract several thousand base pairs long reads.

Method	Read length	Accuracy	Reads per run	Time per run	Cost per 1 million bases (in US$)	Advantages	Disadvantages
Single-molecule real-time sequencing (Pacific Biosciences)	5,500 bp to 8,500 bp avg (10,000 bp N50); maximum read length >30,000 bases	99.999% consensus accuracy; 87% single-read accuracy	50,000 per SMRT cell, or ~400 megabases	30 minutes to 2 hours	$0.33–$1.00	Longest read length. Fast. Detects 4mC, 5mC, 6mA	Moderate throughput. Equipment can be very expensive.
Ion semiconductor (Life Technologies)	up to 400 bp	98%	up to 80 million	2 hours	$1	Less expensive equipment. Fast.	Homopolymer errors.
Pyrosequencing (454 Lifesciences)	700 bp	99.9%	1 million	24 hours	$10	Long read size. Fast.	Runs are expensive. Homopolymer errors. ***This technology was dropped by Roche in 2013. No more relevant***
Sequencing by synthesis (Illumina)	50 to 300 bp	98%	up to 3 billion	1 to 10 days	$0.05 to $0.15	Potential for high sequence yield, depending upon sequencer model and desired application.	Equipment can be very expensive. Requires high concentrations of DNA.
Nanopore Techology (Oxford Nanopore and others)	Up to 80k bp	60% to 85%			$10 (2013)	Extremely long reads. Sequencing devices can be very small and cheap.	Accuracy is still low (60% to 85%).

Table 1 - Comparison of next-generation sequencing methods

2.3 Bioinformatics and genomics

The term *bioinformatics* designates the scientific domain that uses computing infrastructures to analyse biological data. The term is very broad and includes several fields and disciplines. Among them *genomics* applies DNA and RNA sequencing methods and computers to analyse the genomes of human beings and other organisms.
The main goals of genomics include:

1. The identification of the complete genome of organisms
2. The comparison among genomes of different organisms

3. The study of mutations in time of the genome of a given organism
4. The identification of genes (portions of a genome) functions
5. The definition of the spatial structure of genes within a genome

All applications of genomics such as genomic medicine, synthetic biology and bioengineering share the need of accessing and transporting genomic data rapidly and efficiently. The rapid evolution in genomic information generation is requiring dramatic advancements in databases technology, computational platforms, mathematical and statistical methods and theory to meet the requirements of the management and analysis of large biological datasets.
Nowadays tools used by bioinformaticians range from simple scripts to large commercial products. A large literature of open source software is available in various forms from development projects followed by communities of developers to simple reference software accompanying scientific publications. In some cases such tools implement very sophisticated compression schemes using entropy and arithmetic coding, but none of them meets all the requirements of the applications mentioned above. In particular, support for random access to data according to criteria expressed in a formal way is an important feature that the scientific community in looking for.

2.4 Applications

2.4.1 Diagnostics and Personalized Medicine

Genome-based diagnostic tests have been recently developed to make personalized treatment possible thanks to discovered links between specific genetic variants and diseases. Such tests have the potential to predict risk and drive preliminary therapeutic interventions, to detect onset of disease, or detect residual disease. Although clinicians and patients are still far from being educated in how best to apply genetic knowledge in better targeting (that is, in whom to intervene) and tailoring (how best to intervene) preventive efforts, improved health is a major goal of genomic research.

2.4.2 Drug Discovery

Complete knowledge of the functions of all human genes might dramatically change drug discovery development processes and drug research as a whole. The application of genomic technologies to the clinical development of new and existing drugs is known as pharmacogenomics. Thanks to the recent development in genomics and pharmacogenomics in clinical research and clinical medicine, diseases could be treated in a close future according to genetic and specific individual markers, so that medications and dosages could be optimized according to the genetic profile of individual patients.

2.4.3 Biomarker Discovery

In medicine a biomarker is an indicator of the presence of a disease state or any other physiological state. More generally anything that is measurable and related to the state of an organism can be considered a biomarker. The pharmaceutical industry is increasingly interested in biomarker discovery because biomarkers could represent early signals of disease in clinical trials, and possible drug targets.

3 Some relevant initiatives

3.1 ISO Technical Committees

3.1.1 ISO TC 215 - Health Informatics

The scope of this ISO Technical Committee is defined as: "*Standardization in the field of information for health, and Health Information and Communications Technology (ICT) to promote interoperability between independent systems, to enable compatibility and consistency for health information and data, as well as to reduce duplication of effort and redundancies.*"
Among the produced standard documents the most interesting is **ISO 25720:2009 - Genomic Sequence Variation Markup Language (GSVML)**.
The scope of this standard is the specification of a common format for the exchange of genomic sequence variation data among existing databases. The aim is to define a standard envelop able to carry all the major existing formats for human genomic data.
While this is the most interesting effort for the standardization of a file format for the encoding of genomic information, the design of ISO 25720:2009 does not specifically address issues around efficient data compression and support of next generation sequencing technologies. Some of the main weaknesses are listed below.

- It has not been revised in the light of the new sequencing technologies that produce both human, virus and bacteria sequences in shot.
- It is not conceived to improve storage efficiency as it's entirely XML based and annotation-based (variant annotation to be precise)
- It only meets the requirements of those applications interested in the variation at a single position in a gene. Since 2010 the field has evolved tremendously and this is not sufficient any more for a broad range of applications.

3.1.2 ISO TC 276 – Biotechnology

Among the mandates of this Technical Committee, the standardization of "Computing tools, bioinformatics for international comparability and integrability of data" is mentioned.
The most recent activity has been the organization of a Workshop in October 2011 on "International Standards for Biotechnology". The goal of the workshop was to create an opportunity "*to promote a dialogue among the organizations most active in standardization for biotechnology, to foster better understanding among the key players and to capture input, recommendations on relevant matters and possible priority action items which will be channeled for consideration to the existing ISO technical and governance bodies*".
The workshop outcome has been a set of recommendations on how ISO work in the biotechnology field should be structured. In particular it is interesting that one of the recommendations includes the need to ***standardize data structuring and processing for genomic applications.***

3.2 Non-ISO initiatives and groups

3.2.1 Pistoia Alliance Inc.

The Pistoia Alliance Inc. [4] is a private consortium of pharmaceutical industries, universities and research centres which aims at supporting collaboration in the development of tools and technology for the manipulation of biological data. Its mission is to "*lower barriers to innovation by improving the interoperability of R&D business processes through precompetitive collaboration*". While the scope of the Alliance is very broad, it is worth mentioning an initiative

promoted in 2012 for the comparison of the most popular and efficient tools for DNA information compression: the SequenceSqueeze contest [5].

3.2.2 Expert Committee on Biological Standardization of the WHO

The World Health Organization website [6] states that the "*Expert Committee on Biological Standardization is commissioned by WHO to establish detailed recommendations and guidelines for the manufacturing, licensing, and control of blood products, cell regulators, vaccines and related in vitro diagnostic tests. Members of the Expert Committee are scientists from national control agencies, academia, research institutes, public health bodies and the pharmaceutical industry acting as individual experts and not as representatives of their respective organizations or employers. The decisions and recommendations of the Committee are based entirely on scientific principles and considerations of public health*".
As of today the committee had no specific activity on the standardization of compression or file format for genomic information storage, but when contacted they have shown interest in following any activity in this sense.

3.2.3 Global Alliance for Genomics and Health

Founded in 2013, the Global Alliance for Genomics and Health (Global Alliance) is an international coalition, focusing on the implementation of effective genomic and clinical data sharing.
The most active members of the Alliance concerning data sharing and processing are the EBI, the Sanger Institute and the Broad Institute (MIT and Harvard).
The main activities and deliverables include:

1. **A Framework for data sharing.** Set of documents providing a principled and practical framework for the responsible sharing of genomic and health-related data. It contains foundational principles and core elements for responsible data sharing, and is guided by human rights, including privacy, non-discrimination, and procedural fairness.
2. The Genomics API is a freely available open standard for interoperability, which uses common web protocols to support serving and sharing of data on DNA sequences and genomic variation. It includes the SAM/BAM/CRAM formats and tools mentioned above.
3. Specific projects on real-world data sharing
   a. **Matchmaker Exchange**. Search of exome and genome matches through a federated platform (Exchange) to facilitate the matching of cases with similar phenotypic and genotypic profiles (matchmaking) through standardized application programming interfaces (APIs) and procedural conventions.
   b. Beacon Project. Web service interface to test the willingness of international sites to share genetic data in a simple technical context.
   c. The **BRCA challenge** aims to advance understanding of the genetic basis of breast cancer and other cancers by pooling data on BRCA genetic variants from around the world.

4 File Formats

Nowadays the largest majority of public repositories of sequence data provide data formatted in two - very similar - textual file formats named FastA and FastQ. FastQ exists in a few different flavors [7] defined by different sequencing machine vendors.
FastA and FastQ have been adopted in the recent past when the amount of generated information was not so important to create any issue of storage space. In addition, text files can easily be parsed and analyzed using scripting languages (e.g. bash, Perl, python) very popular on the UNIX platforms commonly used in this domain.

The explosion of the throughput of NGS machines pushed the adoption of popular file compression tools such as zip, tar and all the related flavors. These generic approaches to compression can anyway save between 50% and 75% of the original utilized space, which is currently becoming inadequate of at least one order of magnitude with respect to the requirements of faster and faster sequencing technology. The main drawback of this approach is the total lack of support for random access to portions of the compressed information.
FastA and FastQ are described and compared in Section 5.1 while Section 5.2 introduces a new standardized notation for nucleotides sequences aiming at merging the characteristics of FastA and FastQ towards a single file format.

4.1 Unmapped data. FastA/FastQ

FastA and FastQ are very similar text-based formats that are used for genomic information generated by NGS machines.
FastA and FastQ are organized as sequences of 2 (FastA) or 4 (FastQ) fields describing each read produced by a DNA sequencing machine. An example of these fields with a brief description is provided in the picture below.
Table 2 compares the two formats.

FASTQ	Field	FASTA
@HWUSI-EAS100R:6:73:941:1973#0/1	*Header (Unique ID plus other information). Only the first character is standard.*	**>HWUSI-EAS100R:6:73:941:1973#0/1**
GATTTGGGGT…..	*Nucleotides sequence*	**GATTTGGGGT……**
+SRR001666.1 071112_SLXA-EAS1_s_7	*Optional description. Only the first character is standard. This field is becoming obsolete and only "+" is used to separate the previous and the next field*	Not present
!''*((((*+)**	*Quality scores*	Not present

Table 2 - Comparison between FastQ and FastA

Both file formats start with a header field where only the first character ("@" for FastQ and ">" for FastA) is standardized to signal the start of a new read. The remaining text in the header usually identifies the originating experiment, the type of sequencing machine or technology adopted and other information aiming at identifying the source of the data.

The second field contains the symbols used to represent nucleotides in both FastA and FastQ. They are usually 5 types of symbols:

- A, C , G, T (T is replaced by U in case of RNA sequencing)
- A fifth symbol "N" used when the sequencing machine cannot take any decision.

FastQ has two additional fields:

- An optional container of additional metadata starting with "+"
- Quality scores expressing the level of confidence for each nucleotide encoded in the second field. The value and meaning of each symbol vary with the sequencing machine adopted.

4.2 The IUPAC ambiguity codes

The International Union for Pure and Applied Chemistry (IUPAC) has recently standardized a larger set of symbols (16, including the 5 currently used) that includes the information related to the incertitude of the read between one or more basis. This standard set of codes better represents and can replace the non-standard metadata previously used to indicate the "quality" of the base read. Therefore, supporting this slightly larger set of symbols can replace the support of the current non-standard metadata (the latter including quality scores that are usually machine dependent).
While the IUPAC standard is starting being supported by the latest Next Generation Sequencing machines, its use is still limited in the scientific community. Nonetheless the trend seems to points towards a progressive wider adoption with the gradual substitution of quality scores by IUPAC ambiguity codes. A FastA file format extended to 16 (or more) symbols seems to be on the (not so far) horizon.

4.3 Aligned data. SAM/BAM and CRAM.

The most popular and implemented public specifications for genomic information representation and compression are represented by:

- Sequence Alignment/Mapping (SAM) file format and its binary format (BAM).
  This file format is part of the SAMtools toolkit, an open source software library providing utilities to manipulate SAM/BAM files. The SAMtools have been originated by Heng Li, a bioinformatics research scientist working at the Broad Institute, and are now maintained by a community of software developers on the GitHub repository.
- The CRAM file format and toolkit.
  CRAM has been developed by the European Bioinformatics Institute (EBI) and it is currently at v2.1 of the file format specification. Version 3.0 is supposed to be released in the next months. It consists of a set of Java tools and APIs and includes several compression methods. It supports production pipelines of the European Nucleotides Archive.

The file formats and utilities from both initiatives are currently jointly managed by the Data Working Group File Formats Task Team of the Global Alliance for Genomics and Health.
It is important to stress that while CRAM and SAM/BAM are mainly conceived to encode ***aligned*** reads (i.e. the output of tools called “aligners” that map FastA/FastQ reads onto an existing reference genome), they can be used to encapsulate and compress the unmapped reads produced by sequencing machines.

Even if slightly (30% to 50%) less efficient than CRAM in terms of data compression, BAM is the most popular file format for genomic data distribution and storage. The main international repositories of public genomic data (from the US to Europe to Japan) are mainly populated by large BAM files that can have sizes of several hundred GBs according to the specific coverage.

4.3.1 The SAM and BAM file formats

This section provides a summary of the SAM v1 specification, more details can be found in the document available online: http://samtools.github.io/hts-specs. A good summary of SAM feature is available on this SAM wiki entry as well.

SAM is a TAB-delimited text format consisting of

- an optional header section starting with ‘@’

- an alignment section including
  - 11 mandatory fields
  - variable number of optional fields

If present, the header must be prior to the alignments.

Figure 4 and Figure 5 show how some sequence reads are formatted in SAM. The example is taken from the SAM v1 specification and includes read001/1 and read001/2 representing a read pair; r002 is a single read; r003 is a chimeric read and r004 represents a split alignment (a read which needs to be split in order to properly be mapped to the reference genome.

```
Coor     12345678901234  5678901234567890123456789012345
ref      AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

+r001/1        TTAGATAAAGGATA*CTG
+r002         aaaAGATAA*GGATA
+r003       gcctaAGCTAA
+r004                     ATAGCT..............TCAGC
-r003                           ttagctTAGGC
-r001/2                                      CAGCGGCAT
```

Figure 4 – The four reads aligned to a reference genome (ref)

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:45
r001  163 ref  7 30 8M2I4M1D3M = 37  39 TTAGATAAAGGATACTG *
r002    0 ref  9 30 3S6M1P1I4M *  0   0 AAAAGATAAGGATA    *
r003    0 ref  9 30 5S6M       *  0   0 GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004    0 ref 16 30 6M14N5M    *  0   0 ATAGCTTCAGC       *
r003 2064 ref 29 17 6H5M       *  0   0 TAGGC             * SA:Z:ref,9,+,5S6M,30,1;
r001   83 ref 37 30 9M         =  7 -39 CAGCGGCAT         * NM:i:1
```

Figure 5 - The alignment of Figure 4 formatted as SAM file (only the 11 mandatory columns are used here)

4.3.1.1 SAM Terminology

Template	A DNA/RNA sequence part of which is sequenced on a sequencing machine or assembled from raw sequences.
Segment	A contiguous sequence or subsequence.
Read	A raw sequence that comes off a sequencing machine. A read may consist of multiple segments. For sequencing data, reads are indexed by the order in which they are sequenced.
Linear alignment	An alignment of a read to a single reference sequence that may include insertions, deletions, skips and clipping, but may not include direction changes (i.e. one portion of the alignment on forward strand and another portion of alignment on reverse strand). A linear alignment can be represented in a single SAM record (e.g. r002 and r004 in the example above).
Chimeric	An alignment of a read that cannot be represented as a linear alignment. A

alignment	chimeric alignment is represented as a set of linear alignments that do not have large overlaps (e.g. r003 in the example above is composed by two linear alignments). Typically, one of the linear alignments in a chimeric alignment is considered the “representative" alignment and the others are called “supplementary" and are distinguished by the supplementary alignment flag.
Read alignment	A linear alignment (1 SAM record) or a chimeric alignment (several SAM records) that is the complete representation of the alignment of the read.
Multiple mapping	The correct placement of a read may be ambiguous, e.g. due to repeats. In this case, there may be multiple read alignments for the same read. One of these alignments is considered primary. All the other alignments are considered “secondary”. Typically the alignment designated primary is the best alignment, but the decision may be arbitrary.
Phred scale	Given a probability $0 < p \leq 1$, the phred scale of p equals -10 $\log_{10}$p, rounded to the closest integer.

4.3.1.2 The SAM header

The SAM specification states that “each header line begins with character `@' followed by a two-letter record type code. In the header, each line is TAB-delimited and except the @CO lines, each data field follows a format `TAG:VALUE' where TAG is a two-letter string that denes the content and the format of VALUE.”
The SAM header is optional, but when present it has some mandatory fields that are briefly introduced here. For the complete specification of both mandatory and optional fields please refer to the SAM Format Specification document.

Record	Sub-record	Description
@HD		This is the first header line.
	VN	Format version. Accepted format: `/^[0-9]+\.[0-9]+$/.`
@SQ		Reference sequence dictionary. The order of @SQ lines denes the alignment sorting order.
	SN	Reference sequence name. Each @SQ line must have a unique SN tag. The value of this field is used in the alignment records in RNAME and PNEXT fields. Regular expression: `[!-)+-<>-~][!-~]*`
	LN	Reference sequence length. Range: [1, 2^{31}-1]
@RG		
	ID	Read group identifier. Each @RG line must have a unique ID. The value of ID is used in the RG tags of alignment records. Must be unique among all read groups in header section. Read group IDs may be modified when merging SAM les in order to handle collisions.
@PG		Program (used to manipulate the)
	ID	Program record identifier. Each @PG line must have a unique ID. The value of ID is used in the alignment PG tag and PP tags of other @PG lines. PG IDs may be modified when merging SAM files in order to handle collisions.

Table 3 – SAM header mandatory fields

4.3.1.3 The alignment section: mandatory fields

In the SAM format, each alignment line typically represents the linear alignment of a segment. Each line has 11 mandatory fields. These fields always appear in the same order and must be present, but their values can be `0' or `*' (depending on the field) if the corresponding information is unavailable. The following table gives an overview of the mandatory fields in the SAM format:

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	[!-?A-~]{1,255}	Query template name
2	FLAG	Int	$[0,2^{16}-1]$	bitwise flag
3	RNAME	String	*|[!-()+-<>-~][!-~]*	Reference sequence NAME
4	POS	Int	$[0, 2^{31}-1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0, 2^{8}-1]$	MAPping Quality
6	CIGAR	String	*|([0-9]+[MIDNSHPX=])+	CIGAR string
7	RNEXT	String	*|=|[!-()+-<>-~][!-~]*	Ref. name of the mate/next read
8	PNEXT	Int	$[0,2^{31}-1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31}+1, 2^{31}-1]$	observerd Template LENgth
10	SEQ	String	*|[A-Za-z=.]+	segment SEQuence
11	QUAL	String	[!-~]+	ASCII of Phred-scaled base QUALity+33

Table 4 – SAM alignment section mandatory fields

4.4 Compressed SAM: BAM

The BAM file format is the binary equivalent of SAM obtained by compressing SAM using the BGZF (Blocked GNU Zip Format) compression tool.
BGZF implements block compression on top of the standard gzip file format with the goal of both providing good compression and allowing efficient random access to the BAM file.
A BGZF file is a series of concatenated BGZF blocks. Each BGZF block is itself a spec-compliant gzip archive which contains an “extra field" in the format described in RFC1952.
BAM files are essentially composed by a concatenation of BGZF compressed data blocks that can be randomly accessed via a BAM file index that uses *virtual offsets* into the BGZF file. Each virtual file offset is an unsigned 64-bit integer, defined as: `coffset<<16|uoffset`, where `coffset` is an unsigned byte offset into the BGZF file to the beginning of a BGZF block, and `uoffset` is an unsigned byte offset into the uncompressed data stream represented by that BGZF block. More details on the BAM file structure can be found in the SAM/BAM Format specification [8].

4.5 SAM/BAM manipulation

This section contains some examples of the main manipulations usually performed on SAM/BAM files. The first step is usually the alignment of raw data in the FastQ format with respect to a reference genome.

4.5.1 Aligning raw reads

Raw reads encoded as FastQ files are not sorted and contain no notion of relative order apart from the one provided by the sequencing machine according to the specific technology adopted.
The operation of rearranging the raw reads contained in a FastQ file so that they map to a reference genome already available is commonly referred to as “alignment”.
The tools used for alignment are called “aligners” and the most popular currently in use are:

- Burrows Wheeler Aligner (BWA)
- Bowtie (1 and 2)
- The SOAP package

When no reference genome is available and the reads are rearranged only with respect to the mutual similarities and nucleotides sequence overlaps the rearrangement is called (de-novo) “assembly”. The tools used in this case are called Multiple Sequence Aligners (MSA).
Some of the most popular are:

- MAFFT
- MUSCLE
- Clustal

4.5.2 SAMtools for SAM/BAM manipulation

Once the reads are aligned and expressed as SAM or BAM files, they can be manipulated by the SAMtools toolkit [9]. The CRAM toolkit [10] is fully compatible with SAMtools and provides the same functionality together with a more sophisticated support to compression.
The MPEG input document m35679 (111th MPEG meeting in Geneva) provides a short overview of the main SAMtools commands used to access and manipulate the information contained into SAM and BAM files.

5 Genome compression

Several genomic data compression tools have been developed by researchers and developers with different interests in terms of requirements to be met (compression ratio, speed, memory footprint). Comparing such tools is impaired by their very diverse approaches to the problem in terms of test dataset, output format, actual availability of implementations and the related specifications. This section aims at providing a summary of the main approaches and tools currently in use; it is nevertheless incomplete due to the rapid proliferation of new methodologies and implementations.

5.1 Methods

5.1.1 Naive bit encoding

These highly inefficient methods are worth mentioning here only because they were among the first to be used and are still used in some circumstances. They simply encode several nucleotides within the same byte using fixed-length encoding [11]. For example the 4 nucleotides (A, C, G, T) can be encoded using a 2 bit alphabet so that 1 byte can encode 4 nucleotides (compression ratio 4:1 with respect to textual encoding).

5.1.2 Dictionary based

A dictionary of repeated substrings is built at runtime or offline and then compression is performed by replacing each substring with a reference to the dictionary [12].

5.1.3 Statistical methods

Also referred to as entropy encoding algorithms, they derive a probabilistic model from the input data. When appropriately defined, the model is supposed to use the available information to

predict next symbols of the sequence. When a reliable model is built, these methods result in very high compression rates.

5.1.4 Referential methods

Also known as reference based approaches, these methods encode substrings by means of references to an external genome. For each substring of the input data that can be mapped to the reference, only the position and possible deviations with respect to the reference are encoded. With respect to dictionary-based approaches here the reference is static, while dictionaries are usually dynamic and can be updated during compression.

5.2 Tools

5.2.1 The SequenceSqueeze contest

The landscape of existing compression tools can be partitioned in two major classes according to the choice of using an external reference or not. When ordering reads with respect to an external reference, only the relative positions and the differences are encoded in the compressed output, therefore generating highest compression ratios. The limitation of such an approach is that the reference shall be available both at the encoding and decoding side and if not available it shall be transferred with the compressed data (affecting the efficacy of compression).
A recent call for technology has been issued to compare the performance of different approaches to FastQ compression (applicable to FastA as well). This initiative was prompted by the Pistoia Alliance [4] by means of the *SequenceSqueeze* competition.
The results of the comparison have been published in [5], are reported in Table 2, and are accessible at this URL:
http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0059190.
The list of submitted tools includes both FastQ and SAM compressors. The latter are therefore not exactly compressor of raw sequence data as a pre-processing stage to transform raw sequences to SAM is needed.

			SRR027520_1						SRR065390_1					
Prog	**Ref**	**Sort**	**Ratio**	**ID**	**Base**	**Qual**	**C.R.**	**Mem**	**Ratio**	**ID**	**Base**	**Qual**	**C.R.**	**Mem**
Raw FASTQ	N	ID	1.0000	419.9	8	8			1.0000	454.9	8	8		
Fastqz	N	ID	0.2195	11.7	1.71	2.96	3.8	1459	**0.1340**	15.6	**1.11**	1.53	3.8	1527
Fqzcomp(medium)	N	ID	0.2196	11.3	1.72	2.95	**22.0**	**312**	0.1524	14.8	1.52	1.52	**22.4**	**311**
Fqzcomp(slow)	N	ID	**0.2172**	11.3	**1.68**	**2.94**	8.2	4407	0.1341	14.8	1.16	**1.49**	8.3	4406
Quip	N	ID	0.2219	**11.2**	1.78	2.95	8.3	777	0.1584	**14.7**	1.64	1.51	9.0	776
Fastqz	Y	ID	0.1816	**11.7**	0.88	2.96	3.2	1365	**0.1000**	**15.6**	**0.40**	1.53	4.7	1352
Samcomp2	Y	ID	**0.1810**	**19.4**	**0.75**	**2.94**	**13.6**	**1079**	0.1022	19.9	0.43	**1.49**	17.1	**365**
Quip	Y	ID	0.1885	22.2	0.90	2.95	**16.4**	1515	0.1088	21.3	0.54	1.52	**19.1**	807
Fastqz	N	pos	0.2414	52.1	1.66	2.95	3.2	1527	0.1397	64.1	0.74	1.54	4.0	1527
Samcomp1	N	pos	**0.2360**	**49.8**	**1.59**	**2.94**	**15.1**	315	**0.1147**	58.7	**0.29**	1.50	**21.8**	288
Samcomp2	N	pos	0.2628	**49.8**	2.18	**2.94**	13.5	341	0.1982	58.7	2.04	**1.49**	15.2	341
Quip	N	pos	0.2453	50.5	1.78	**2.94**	9.3	776	0.1890	**58.6**	1.83	1.53	11.2	775
SAMtools (BAM)	N	pos	0.4013	137.8	2.79	4.21	13.7	**1**	0.2344	150.9	0.94	2.47	16.7	**1**
Fastqz	Y	pos	0.2009	52.1	0.77	2.95	2.9	1406	0.1184	64.1	0.29	1.54	4.4	1352
Samcomp1	Y	pos	**0.1852**	49.8	**0.47**	**2.94**	15.7	**378**	**0.1116**	58.7	**0.23**	1.50	**21.9**	**296**
Samcomp2	Y	pos	0.1920	49.8	0.62	**2.94**	14.2	1079	0.1163	58.7	0.33	**1.49**	20.1	365
Quip	Y	pos	0.1926	**49.2**	0.64	**2.94**	**16.6**	1516	0.1165	**58.6**	0.32	1.53	19.6	808
Goby[a]	Y	pos	0.2706	99.5	0.62	4.01	4.8	1797	0.1587	110.6	0.28	1.93	6.8	1250
CRAM	Y	pos	0.2504	92.1	0.58	3.71	5.0	1514	0.1676	105.9	0.27	2.17	7.9	898

Showing the compressed file size break down by bits per sequence identifier, per base-call and per quality value. In some cases these sizes refer to cases where a reference was previously used to map, but it has not been used during compression (e.g. BAM). The ID, Base and Qual columns are the number of bits required to store the complete sequence identifier, a single base nucleotide and a single quality value respectively. The C.R. column is the compression rate in MB per second. Mem is the amount of memory required during compression. References used were human hg19 and C.Elegans WS233. Non-reference based Quip used the "-a" assembly option for high compression mode.

[a]Goby does not store unmapped data. The Goby figures have been estimated by adding 2 bits per absent base-call and scaling up the name and quality figures by the percentage of unmapped reads.

doi:10.1371/journal.pone.0059190.t004

Table 5 – Results of the SequenceSqueeze competition.

The results of the *SequenceSqueeze* competition show that the best compression ratios are in the order of 20% of the original size for one dataset and 10% for the other.

5.2.2 The latest generation of tools

A new generation of compression tools tries to address some of the issues of the above mentioned approaches while at the same time trying to provide better compression ratios. In particular, the authors of the DeeZ compression tool [13], believe that the two main drawbacks of CRAM consist in:

1) redundantly encode the common sequence features of the reads mapping to the same locus. This is due to the fact that "CRAM Tools and Scramble represent the differences between each read and its mapping locus separately".
2) lossy compression of some of the SAM fields which are not completely reconstructed during decompression with an impact on downstream analysis.

DeeZ is a SAM file compression tool providing the best lossless compression ratios (2x gzip and 50% better than CRAM) together with random access to compressed data. It "uses a unique compression method for each field of the SAM record in order to exploit its specific properties: read names are 'tokenized' and compressed by the use of delta encoding; quality scores are encoded using an order-2 arithmetic coder, etc." [13].

Other tools with performance close to DeeZ's are Quip [14] and Samcomp [15]. They are based on arithmetic coding as well, but do not provide random access to the compressed data.

The most common approach to arithmetic coding applied to genomic data is to separate headers identifiers, sequences and quality scores in 3 separate streams and then compress them using different models. The key issue is of course to find the best models which have the best balance

between complexity and efficiency. The exploration in this field is still in a very early stage and the existing tools typically privilege simplicity with respect to efficiency in order to quickly achieve reasonable performance with not necessarily optimized implementations.

Tools belonging to the latest generation of genome compression tools can be essentially categorized in two classes:

1. Tools compressing raw reads in FastQ format (a minority)
2. Tools compressing aligned data in the form of SAM/BAM file (the majority)

While tools belonging to class 1 are trying to replace gzip to compress unstructured raw reads as generated by NGS machines, class 2 tools have to handle structured data that go far beyond simple base sequences and quality scores. SAM files can contain the entire history of a genome analysis experiment including information such as the used tools, the infrastructure where the experiment took place together with conclusions and considerations drawn by the authors of the analysis.
Figure 6 shows a functional block diagram of a generic genome analysis pipeline trying to highlight the different stage of data processing and the respective file formats. The picture shows how currently the most popular class 1 compressor is gzip while SAM/BAM and CRAM are (compatible) class 2 compression frameworks (they contain several tools actually).

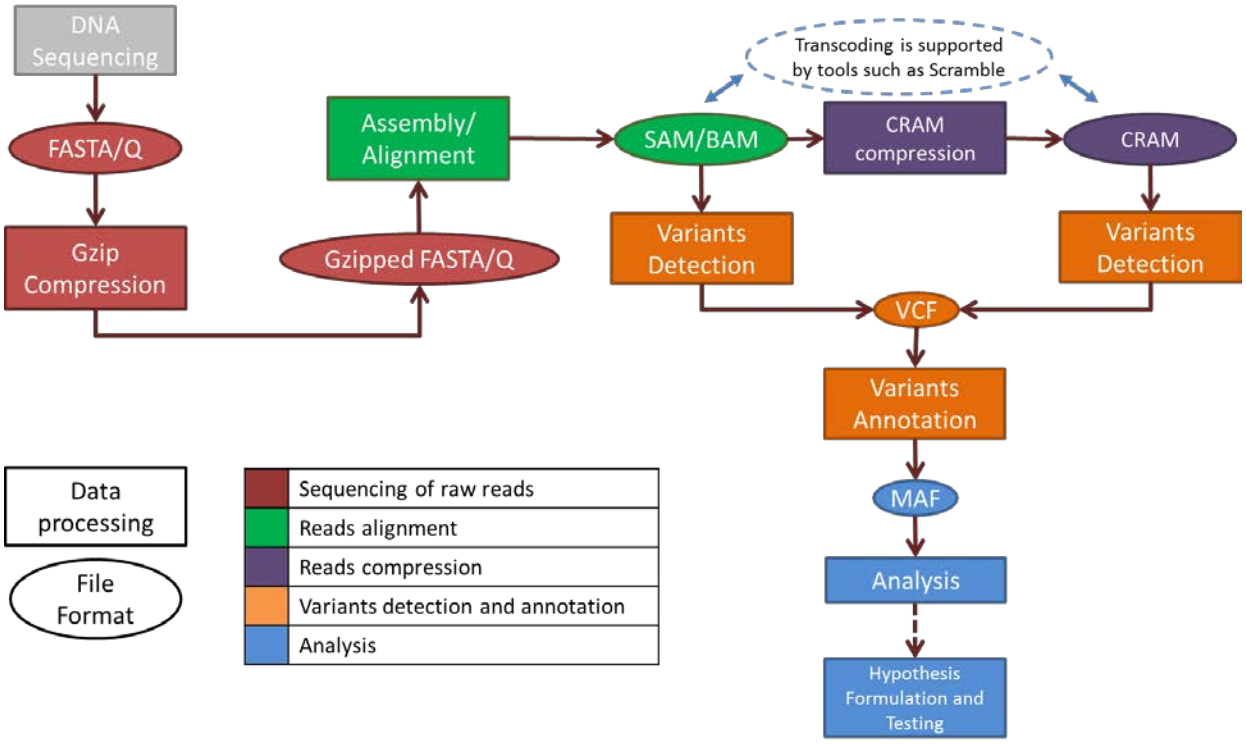


Figure 6 - Typical structure of a genomic information processing pipeline from sequencing to analysis

The paper describing "Deez" [13], one of the latest BAM compressors, contains an extremely comprehensive and complete comparison of the performance of the most popular and efficient compression tools used by the scientific community. URLs to their implementations are listed in Table 6.

Tool Name	URL	Input format
CRAM toolkit	http://www.ebi.ac.uk/ena/software/cram-toolkit	BAM
Deez	http://sfu-compbio.github.io/deez/	BAM
DSRC1-2	http://sun.aei.polsl.pl/dsrc/	FastQ
Goby	http://campagnelab.org/software/goby/	FastQ
Quip	http://homes.cs.washington.edu/~dcjones/quip/	FastQ, BAM
SAMtools	https://github.com/samtools/samtools	SAM/BAM
Sam_comp	http://sourceforge.net/projects/samcomp/	SAM/BAM
SCALCE	http://scalce.sourceforge.net/Home	FastQ
Scramble	http://sourceforge.net/projects/staden/files/io_lib/	SAM/BAM

Table 6 – Last generation compression tools

One of the most comprehensive reviews of a wide range of genomic data compression tools is provided in [16].

6 Compression tools comparison

This section contains a first draft comparison of performance of the most popular genomic information compression tools used on some of the samples contained by the reference dataset published during the 111th MPEG meeting held in Geneva in February 2015 and described in the output document N15092.
In the comparison an additional tool named ***tsc*** developed at TNT - Leibniz Universität Hannover and presented as input contribution to the 112th MPEG meeting in Warsaw has been added to the tools listed in section 6.2.2.
The goal is to compare tools performance on the same meaningful set of data which is supposed to represent a wide spectrum of genomic data from different species and sequencing technologies.
This activity will permit to start assessing the magnitude of the compression factors reachable with state of the art technology. When possible (i.e. when supported by the used tool) the metrics are provided for the classes of data composing the whole structure of genomic sequence data:

- Reads identifiers,
- Sequence reads
- Quality scores
- Auxiliary data (when present)

The annexed xls file w15346_GenomeCompressionTools.xls contains a detailed report on the performed tests. This section provides a summary of the most relevant metrics.

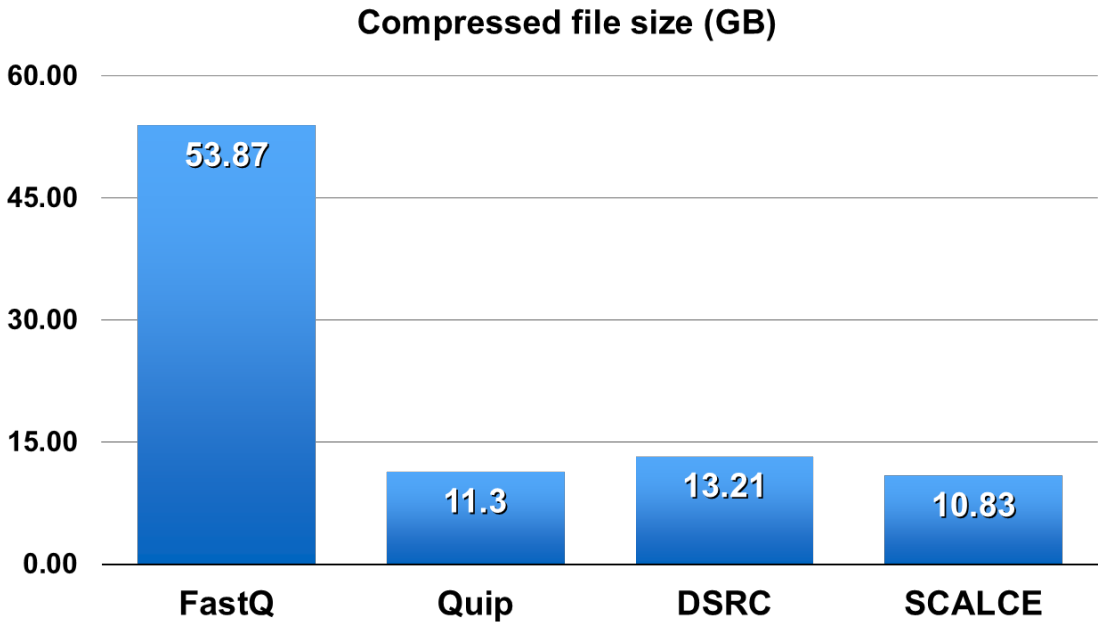
6.1 Raw sequence data (FastQ)

6.1.1 Homo Sapiens

Sample: ERR174310_1.fastq
Original file size: 53.87 GB

Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Quip	11.3	4.77	4.17%	55.61%	40.32%	0.00
DSRC	13.21	4.08	3.68%	56.67%	39.65%	0.00

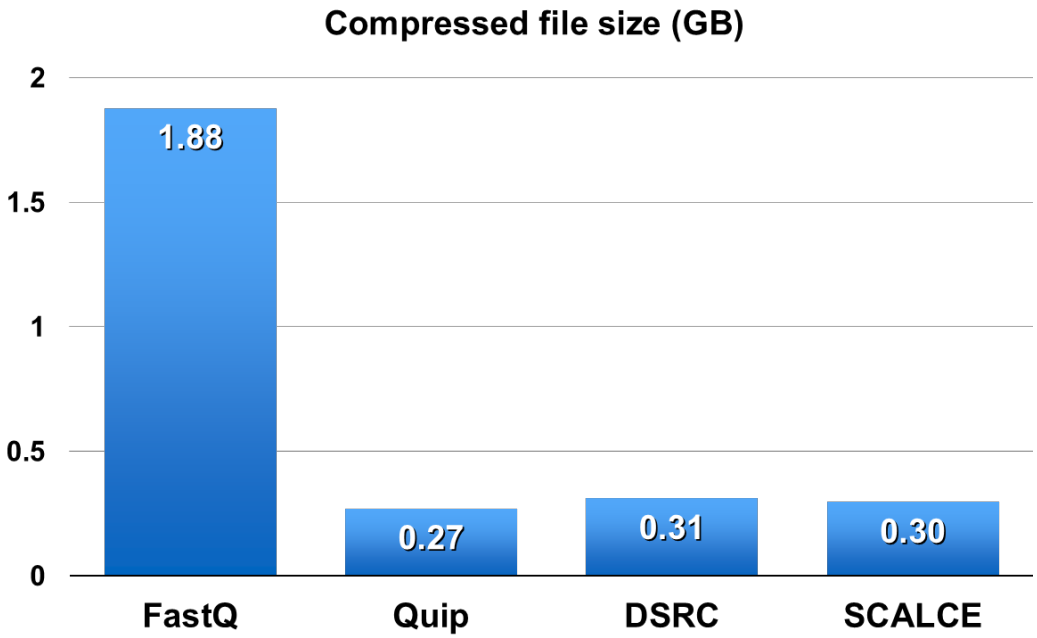
SCALCE	10.83	4.98	9.92%	62.22%	27.86%	0.00



6.1.2 Metagenomics : Human gut

Sample: MH0001_081026_clean_1.fq
Original file size: 1.88 GB

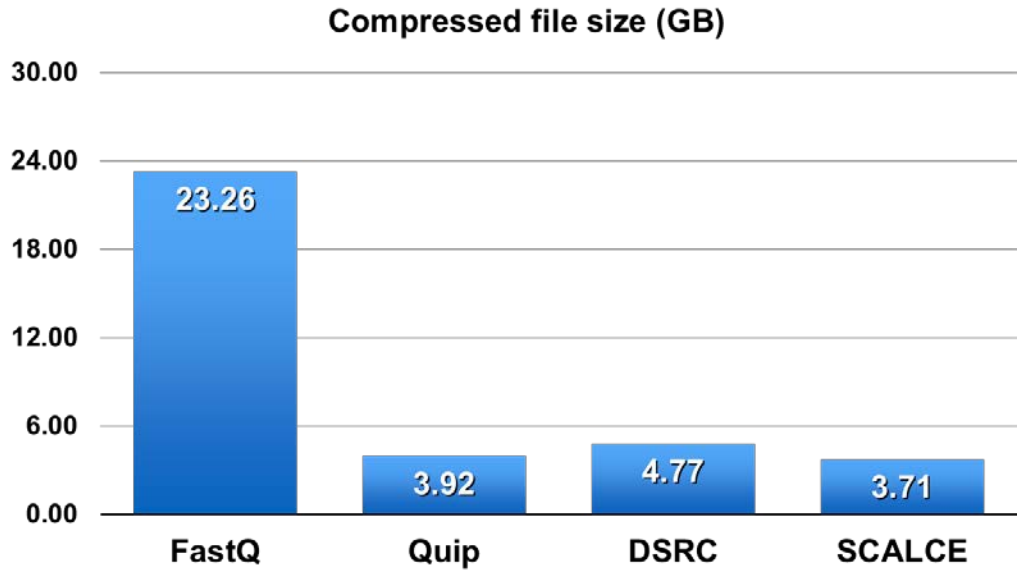
Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Quip	0.27	6.92	6.00%	52.13%	41.87%	0.00
DSRC	0.31	6.02	5.28%	53.69%	41.03%	0.00
SCALCE	0.30	6.32	25.75%	50.49%	23.76%	0.00



6.1.3 Plants : Cacao

Sample: SRR870667_1.fastq
Original file size: 23.26 GB

Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Quip	3.92	5.93	5.18%	57.44%	37.38%	0.00
DSRC	4.77	4.88	4.11%	56.71%	39.18%	0.00
SCALCE	3.71	6.27	9.39%	63.72%	26.89%	0.00

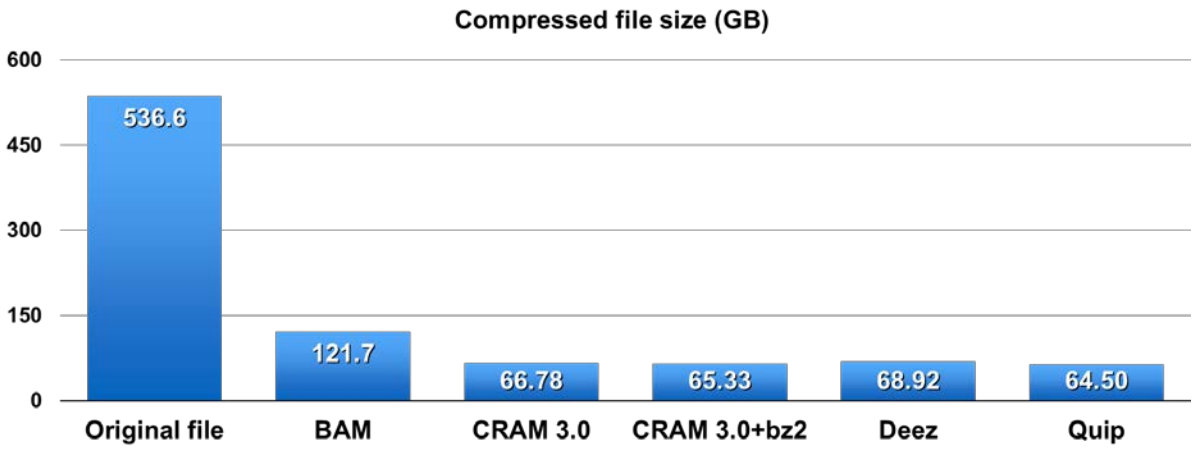


6.2 Aligned data (SAM/BAM)

6.2.1 Homo Sapiens (High coverage)

Sample: NA12878_S1.sam
Original file size: 536.6 GB

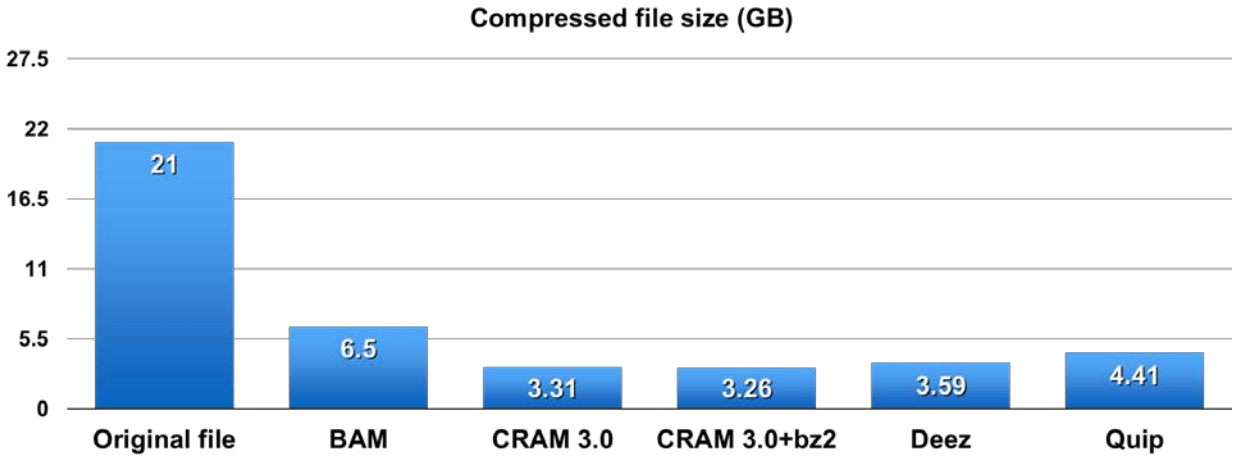
Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Samtools (BAM)	121.7	4.41				
CRAM 3.0	66.78	8.04	13.91%	75.62%	8.57%	1.90%
CRAM 3.0 + bz2	65.33	8.21	12.47%	77.29%	8.76%	1.47%
Deez	68.92	7.79	10.61%	68.08%	15.70%	5.61%
Quip	64.50	8.32	12.03%	67.91%	14.17%	5.83%



6.2.2 Homo Sapiens (Low coverage)

Sample: 9827_2#49.sam
Original file size: 21.06 GB

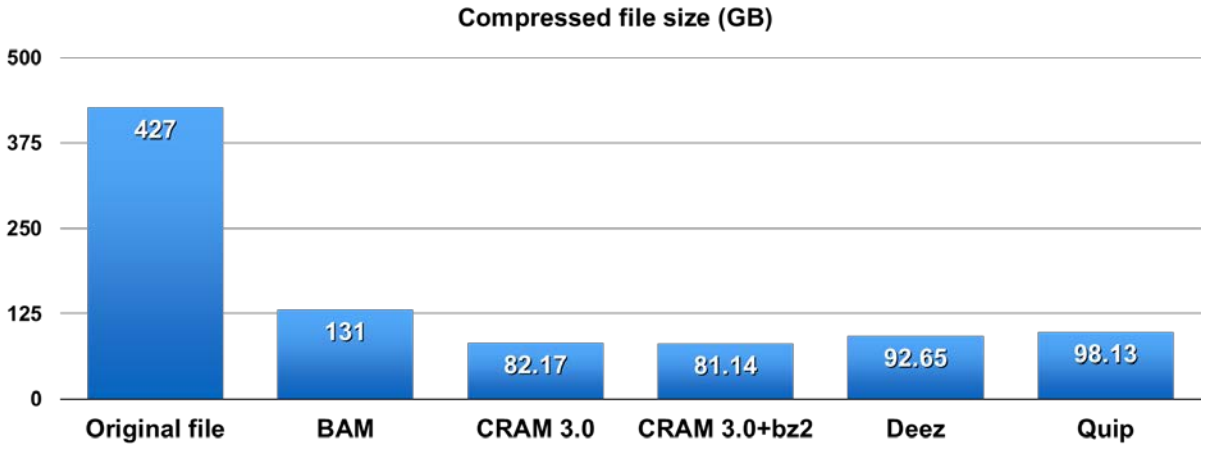
Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Samtools (BAM)	6.5	3.24				
CRAM 3.0	3.31	6.37	8.71%	80.83%	6.28%	4.18%
CRAM 3.0 + bz2	3.26	6.45	8.40%	81.91%	6.37%	3.32%
Deez	3.59	5.86	6.18%	71.50%	12.21%	10.10%
Quip	4.41	4.78	5.94%	53.59%	34.62%	5.84%



6.2.3 Cancer cell lines

UCSC ARTIFICIAL MIXED SAMPLE
Sample : HCC1954.mix1.n80t20.sam
Original file size: 427.03 GB

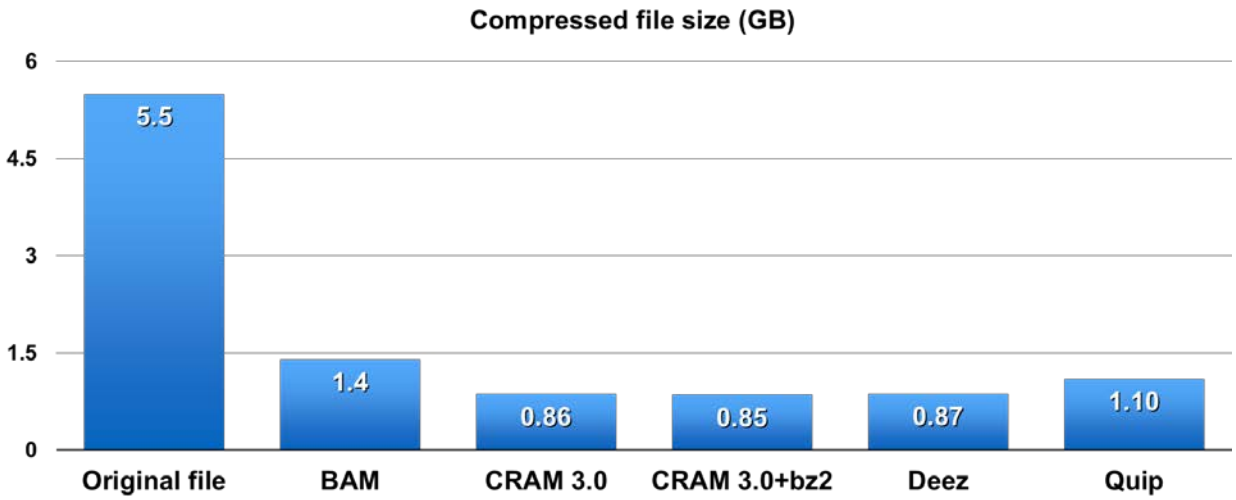
Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Samtools (BAM)	131	3.26				
CRAM 3.0	82.17	5.20	7.01%	40.62%	10.87%	41.50%
CRAM 3.0 + bz2	81.14	5.26	6.11%	41.14%	11.01%	41.75%
Deez	92.65	4.61	6.18%	35.12%	8.97%	49.73%
Quip	98.13	4.35	6.68%	35.77%	29.27%	42.18%



6.2.4 Bacteria (Low coverage)

DH10B (E.Coli)
Sample: MiSeq_Ecoli_DH10B_110721_PF.sam
Original file size: 5.58 GB

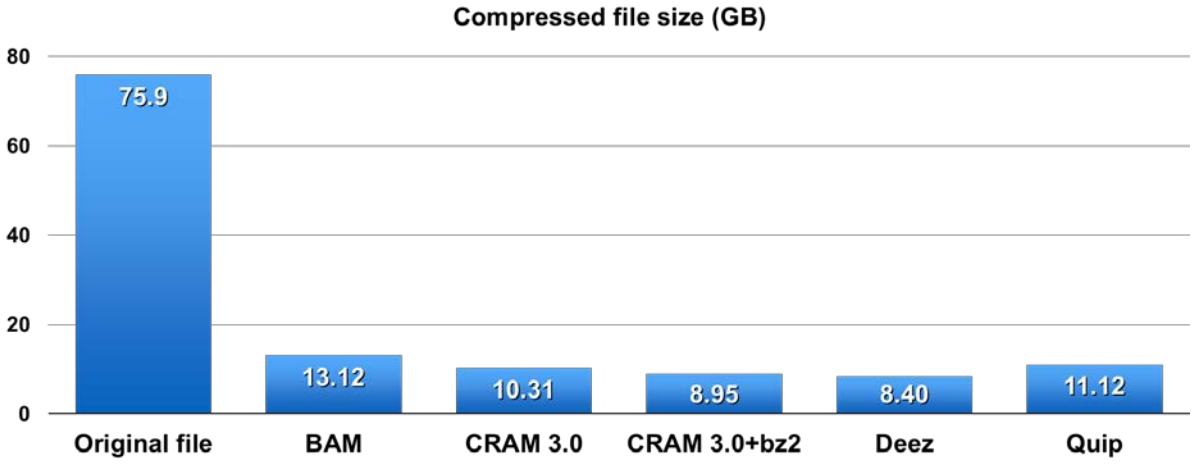
Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Samtools (BAM)	1.4	3.99				
CRAM 3.0	0.86	6.46	7.57%	82.75%	4.35%	5.34%
CRAM 3.0 + bz2	0.85	6.56	6.90%	84.00%	4.41%	4.69%
Deez	0.87	6.41	5.29%	77.58%	8.93%	8.20%
Quip	1.10	5.07	5.03%	57.63%	33.11%	4.23%



6.2.5 RNAseq

Sample: K562_cytosol_LID8465_TopHat_v2.sam
Original file size: 75.92 GB

Tool	Compressed file size (GB)	Compression Factor	Identifiers	Quality Scores	Sequence	Aux
Samtools (BAM)	13.12	5.79				
CRAM 3.0	10.31	7.36	16.05%	62.49%	9.99%	1.79%
CRAM 3.0 + bz2	8.95	8.48	15.03%	72.02%	11.48%	1.47%
Deez	8.40	9.04	11.79%	72.80%	9.98%	5.42%
Quip	11.12	6.83	9.16%	53.92%	31.94%	4.97%



7 Available sequence data

Sequence data for research purposes are published by several organizations around the globe. Among the richest dataset we can mention:

- The 1000 genome project [17] with more than 2000 sequence data from human genomes
- The Gene Expression Omnibus (GEO) repository [18] from the US National Center for Biotechnology Information (NCBI) [19]
- The European Nucleotide Archive (ENA) [20] of the European Bioinformatics Institute (EBI) [21]

The vast majority of sequence data available on these repositories exists in the form of zipped FastQ files with sizes in the order of several GB. All publications describing the principles and functioning of compression tools usually refer to some of the data available from these repositories when measuring performance. Comparison among different works is usually difficult as the chosen dataset are different both in terms of biological characteristic (originating organisms) and sequence technology.

One important step towards a coherent comparison among different compression tools and approaches would be the definition of a shared reference dataset that covers the widest possible range of organisms and sequencing technology.

8 Genomic information reference dataset

In order to assess the features and performance of available and new compression techniques, MPEG has identified a limited set of genomic data publicly available covering the largest possible extent of sequencing technology and type of experiments.

A separate output document (N15092) produced at the 111th MPEG meeting in Geneva documents a first selection of reference data to be used to test existing and new compression techniques.

9 Requirements from identified applications

MPEG has created an ad-hoc group for the definition of requirements for compression of genomic information to be used to validate and assess the existing compression tools.

In order to be meaningful, precise requirements shall be formulated in the context of each stage of a genomic processing pipeline because different stages (e.g. sequencing, alignment, analysis) can have very different requirements.

A separate document (N15093) produced at the 111th MPEG meeting in Geneva provides a first draft list of requirements at every stage of the typical genomic analysis pipeline from sequencing to variant calls and mutation annotation.

10 Beyond storage

Efficient genomic information compression can help the scientific community not only by saving transfer time and storage space but also in improving the performance of another critical stage of genomics that is *de-novo* assembly. *De-novo* assembly tries to build (parts of) a genome from raw sequence reads without the help of an external reference. This is usually implemented using de Brujin graphs [22] that might require hundreds of GB of memory during processing. Studies have shown that efficient compression can help in reducing the memory usage of *de-novo* assembly by one order of magnitude [23].

Another application of efficient genome compression is the training of expert models on one specific sequence to be able to use the acquired knowledge to align another similar sequence. The resulting aligners are shown to have a higher quality despite a lower speed [24].

11 Conclusions

In a period of investigation for improved sequencing data representations this document aims at providing a rough (even though incomplete) overview of existing tools and approaches for data compression. While performance in terms of compression and speed might be acceptable in some cases, what appears to be missing is a solution that meets at least the requirements mentioned in section 9 and listed in document N15093. Such a solution would enable the scientific and industrial community working on genomic information to address the challenges of a domain where the variety amongst individuals is higher than what was expected only a few years ago.

The two driving elements of the design process should be on one hand the reuse of existing well-established technologies for representation, compression, storage, access, etc., and on the other hand the flexibility to incrementally address current and future needs without being bound to specific application constraints.

12 References

[1] S. D. Kahn, "On the Future of Genomic Data," *Science,* vol. 331, pp. 728-729, 2011.

[2] S. Wandelt, M. Bux and U. Leser, "Trends in Genome Compression," *Journal of Current Bioinformatics,* 2013.

[3] H. Li, "SAM/BAM and related specifications," [Online]. Available: http://samtools.github.io/hts-specs/.

[4] "Pistoia Alliance," [Online]. Available: http://www.pistoiaalliance.org/.

[5] J. K. Bonfield and M. V. Mahoney, "Compression of FASTQ and SAM Format Sequencing Data," 2013. [Online]. Available: http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0059190.

[6] WHO, "WHO Expert Committee on Biological Standardization," [Online]. Available: http://www.who.int/biologicals/expert_committee/en/.

[7] P. Cock, C. Fields, N. Goto, M. Heuer and P. Rice, "The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants," *Nucleic Acid Research,* vol. 38, no. 6, pp. 1767-1771, 2009.

[8] The SAM/BAM Format Specification Working Group, "Sequence Alignment/Map Format Speci," 2014.

[9] "SamTools," [Online]. Available: http://samtools.sourceforge.net/.

[10] "Cram Toolkit," ENA European Nucleotide Archive, [Online]. Available: http://www.ebi.ac.uk/ena/software/cram-toolkit.

[11] L. Chen, S. Lu and J. Ram, "Compressed pattern matching in dna sequences," in *Proceedings of IEEE Computational Systems Bioinformatics Conference*, 2004.

[12] A. Moffat and J. Larsson, "Off-line Dictionary Based Compression," in *Proceedings of the 1999 Conference on Data Compression*, 2000.

[13] F. Hach, I. Numanagic and S. Sahinalp, "DeeZ: reference-based compression by local assembly," *Nature Methods,* pp. 1082-1084, 2014.

[14] D. Jones, W. Ruzzo, X. Peng and M. Katze, "Compression of next-generation sequencing reads aided by highly efficient de novo assembly," *Nucleic Acids Res.,* 2012.

[15] J. Bonfield and M. Mahoney, "Compression of FASTQ and SAM Format Sequencing Data," *PLoS ONE*, 2013.

[16] S. Deorowicz and S. Grabowski, "Data compression for sequencing data," *Algorithms for Molecular Biology*, vol. 8, no. 25, 2013.

[17] "1000 Genomes," [Online]. Available: http://www.1000genomes.org/.

[18] "Gene Expression Omnibus," [Online]. Available: http://www.ncbi.nlm.nih.gov/geo/.

[19] "The National Center for Biotechnology Information," [Online]. Available: http://www.ncbi.nlm.nih.gov/.

[20] "European Nucleotide Archive," [Online]. Available: https://www.ebi.ac.uk/ena.

[21] "European Bioinformatics Institute," [Online]. Available: https://www.ebi.ac.uk/.

[22] P. Compeau, P. Pevzner and G. Tesler, *How to apply de Bruijn graphs to genome assembly*, vol. 29, 2011, pp. 987-991.

[23] T. Conway and A. Bromage, "Succinct data structures for assembling," *Bioinformatics*, vol. 27, no. 4, pp. 479-486, 2011.

[24] M. Cao, T. Dix and L. Allison, "A genome alignment algorithm based on compression," *BMC Bioinformatics*, vol. 11, no. 1, p. 599, 2010.

[25] M. H.-Y. Fritz, R. Leinonen, G. Cochrane and E. Birney, "Genome Research," 2011. [Online]. Available: http://genome.cshlp.org/content/21/5/734.long.

[26] J. Bonfield, "sam_comp," [Online]. Available: http://sourceforge.net/projects/samcomp/.

[27] F. Campagne, K. C. Dorff, N. Chambwe, J. T. Robinson, J. P. Mesirov and T. D. Wu, "Compression of structured high-throughput sequencing data," 2013. [Online]. Available: http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0079871.

[28] P. Li, X. Jiang, S. Wang, J. Kim, H. Xiong and L. Ohno-Machado2, "HUGO: Hierarchical mUlti-reference Genome cOmpression for aligned reads," *Journal of the American Medical Informatics Association*, 2014.

[29] B. G. Chern, I. Ochoa, A. Manolakos, A. No, K. Venkat and T. Weissman, "Reference Based Genome Compression," in *IEEE ITW - IEEE Information Theory Workshop*, Lausanne, 2012.

[30] S. Grabowski, S. Deorowicz and Ł. Roguski, "Disk-based genome sequencing data compression," *Bioinformatics*, 2014.

[31] Genome Research Limited, "Samtools," Genome Research Limited, [Online]. Available: http://www.htslib.org/doc/samtools-1.2.html.