Scientific Journal of Impact Factor(SJIF): 3.134

e-ISSN(O): 2348-4470
p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2,Issue 1, January -2015

GENOME DATABASES FORMATS AND THEIR TOOLS -THE STATE OF THE ART SURVEY

More Shivaprasad[1], Dr. Kulhalli Kshama[2]

[1] *Department of Computer Science and Engineering, Assistant Prof, Sou.Sushila Danchand Ghodawat Charitable Trust's Sanjay Ghodawat Group of Institutions, Atigre, Kolhapur, (MS) India, shivaprasadmore@gmail.com*
[2]*Department of Information and Technology, Prof, Dr. D.Y. Patil College of Engineering & Technology, Kolhapur, (MS) India, kvkulhalli@gmail.com*

Abstract*: Biological sciences have large amount of genomic data and there is challenge to deal with this huge amount of data for the researchers. Genomic data are commonly represented in tables stored as plain text files and requires parsing for analysis, which is very time consuming and error prone method. The indexing facilities provide efficient access to data along with providing useful methods of summarizing columns. Analysis of code can also be substantially simpler as well as being uniform across different data formats. These benefits of reduced code complexity and greatly increased performance allow users much greater freedom to explore their data.*

Keywords: *VCF, BAM, SAMtools, GFF, BEDtool, BigWig, BigBed, Bwtool, Tabix, SAM tools, NCList, Ensembl, Indexing*

I. VARIANT TOOL CHEST: AN IMPROVED TOOL TO ANALYZE AND MANIPULATE VARIANT CALL FORMAT (VCF) FILES

Ebbert *et al*., In paper Variant Tool Chest(VTC): an improved tool to analyze and manipulate variant call format (VCF) files elaborates that the variant call format(VCF) has become standard file format for storing variants identified in next-generation sequencing(NGS) and other studies.VCF files have total eight fixed fields. They developed software package called Variant Tool Chest for manipulating, comparing and analyzing VCF file which provides better functionality as compare to existing tools.VTC functionality can cope up with existing software in well efficient manner. In VCF summary files contains minimal information like chromosome (CHROM), position (POS), reference allele (REF), variant allele (ALT) and genotypes. The Variant Tool Chest supports existing softwares by extending their capabilities without replicating existing solutions for working with VCF files.VTC can work with a combination of multi- and single-sample VCF files.VTC can handle a mix of single and multi-sample VCF files, with the user defining which sample(s) to use from each of the VCF files. VTC contains a powerful set operation tool named "SetOperator" designed to perform simple or complex set operations like intersects, complements, and unions using VCF files. VTC can perform set operations on a single multi sample VCF file, or a combination of multi- and single sample VCF files. It is helpful and makes sense for a researcher to store all genotypes for a single family in a single VCF file. VTC currently has five genotype-level intersect methods and two record-level intersect methods. The genotype-level intersect methods are as follows: (1) heterozygous; (2) homozygous variant; (3) heterozygous or homozygous variant; (4) homozygous reference; and (5) match sample exactly across variant pools. The record-level intersect methods are: (1) variant; and (2) position. The genotype-level intersect methods require that all sample genotypes involved in the intersect fall into the specified category. One exception is that the heterozygous genotype requires the sample to have a reference allele. This distinction is made assuming that researchers interested in identifying heterozygotes will assume the samples have a reference allele. This also greatly simplifies several corner cases when dealing with multiple variants at a single location. The record-level intersect methods ignore genotypes and only consider whether the variant pools included in the analysis contain the variant. The "position" method only considers chromosome, position, and the reference allele, while the "variant" method also includes the alternate allele(s). For the "variant" method, records with multiple alternates are considered to intersect if at least one of the alternates matches. There are currently three complement methods: (1) heterozygous or homozygous variant; (2) exact genotype matches; and (3) variant.

II. BIGWIG AND BIGBED: ENABLING BROWSING OF LARGE DISTRIBUTED DATASETS

Kent *et al*., In paper BigWig and BigBed: enabling browsing of large distributed Datasets explains BigWig and BigBed files are compressed binary indexed files containing data at several resolutions that allow the high performance display of next-generation sequencing experiment results in the UCSC Genome Browser. As a result, only the data needed to support the current browser view is transmitted rather than the entire file, enabling fast remote access to large distributed data sets. BigBed files are generated from Browser Extensible Data (BED) files. It is simple text file consist fields like chromosome name, start position and end position. BigWig files are derived from text-formatted wiggle plot (wig) or bedGraph files. They

associate a floating point number with each base in the genome, and can accommodate missing data points. In the UCSC Genome Browser, these files are used to create graphs in which the horizontal axis is the position along a chromosome and the vertical axis is the floating point data. To create a BigBed or BigWig file, one first creates a text file in BED, fixedStep, variableStep or bedGraph format and then uses the bedToBigBed, wigToBigWig or bedGraphToBigWig command-line utility is used to convert the file into indexed binary format. Once a BigBed or BigWig file is created, it can be viewed in the UCSC Genome Browser by using the custom track mechanism. In brief the indexed file is put on a website accessible via HTTP, HTTPS or FTP, and a line describing the file type and data location in the form: When the custom track is loaded and displayed, the Genome Browser fetches only the data it needs to display at the resolution appropriate for the size of the region being viewed.

III. TABIX: FAST RETRIEVAL OF SEQUENCE FEATURES FROM GENERIC TAB-DELIMITED FILES

Heng Li explains Tabix is the first generic tool that indexes position sorted files in TAB-delimited formats such as GFF, BED, PSL, SAM and SQL export, and quickly retrieves features overlapping specified regions. Tabix features include few seek function calls per query, data compression with gzip compatibility and direct FTP/HTTP access. Tabix is implemented as a free command-line tool as well as a library in C, Java, Perl and Python. It is particularly useful for manually examining local genomic features on the command line and enables genome viewers to support huge data files and remote custom tracks over networks. Interval queries are frequently needed for retrieving features overlapping specified regions. Interval queries may require to read entire file few times, or preload the file in memory if interval queries are frequently performed. A few specialized binary formats including bigBed/bigWig and BAM have been developed very recently to achieve efficient random access to huge datasets while supporting data compression and remote file access, which greatly helps routine data processing and data visualization. At present, these advanced indexing techniques are only applied to BED, Wiggle and BAM. Nonetheless, as most TAB-delimited biological data formats (e.g. PSL, GFF, SAM, VCF and many UCSC database dumps) contain chromosomal positions, one can imagine that a generic tool that indexes for all these formats is feasible and this tool is Tabix.

IV. BWTOOL: A TOOL FOR BIGWIG FILES

Pohl *et al*., In paper bwtool: a tool for bigwig files explains BigWig files are a compressed, indexed, binary format for genome wide signal data for calculations (e.g. GC percent) or experiments (e.g. ChIP-seq/RNA-seq read depth). bwtool is a tool designed to read bigwig files rapidly and efficiently, providing functionality for extracting data and summarizing it in several ways, globally or at specific regions. The tool enables the conversion of the positions of signal data from one genome assembly to another, also known as 'lifting'. bwtool can be useful for the analyst frequently working with bigWig data, which is becoming a standard format to represent functional signals along genomes. The bigWig format (Kent et al., 2010) was created as a means for the UCSC Genome Browser to access real-valued signal data remotely hosted on HTTP/FTP servers worldwide. BigWig is specific to numerical data. WIG and BAM are both common data formats and are used by many applications, command-line software under the UNIX operating system called bwtool in a similar spirit to bedtools (Quinlan and Hall, 2010) or samtools (Li et al., 2009) that offers the possibility to carry out a number of diverse operations on bigWigs in a convenient way. Until now, the common procedure to access the data within bigWig files has been to use the tools available from UCSC: bigWigToWig, bigWigSummary, bigWigAverageOver Bed, bigWigMerge, bigWigCorrelate or bigWigInfo. These offer some basic usability for bigWigs. bigWigInfo provides instant information about a bigWig file and is useful for glancing at the overall mean and standard deviation as well as seeing how many bases are covered by the signal. bigWigToWig is indispensible, as it is occasionally necessary to convert a bigWig into the original WIG to use legacy software. Beyond those two, bwtool provides additional features and flexibility not found in other software.

V. THE VARIANT CALL FORMAT AND VCF TOOLS

Danecek *et al*., In paper The variant call format and VCF tools elaborates The variant call format (VCF) is a generic format for storing DNA polymorphism data such as SNPs, insertions, deletions and structural variants, together with rich annotations. VCF is usually stored in a compressed manner and can be indexed for fast data retrieval of variants from a range of positions on the reference genome. VCFtools is a software suite that implements various utilities for processing VCF files, including validation, merging, comparing and also provides a general Perl API. Variant call format (VCF) as a standardized format for storing the most prevalent types of sequence variation, including SNPs, indels and larger structural variants, together with rich annotations. The VCF format is scalable so as to encompass millions of sites with genotype data and annotations from thousands of samples by adopting a textual encoding, with complementary indexing, to allow easy generation of the files while maintaining fast data access. VCFtools is an open-source software package for parsing, analyzing and manipulating VCF files. The software suite is broadly split into two modules. The first module provides a general Perl API, and allows various operations to be performed on VCF files, including format validation, merging, comparing, intersecting, making complements and basic overall statistics. The second module consists of C++ executable primarily used to analyze SNP data in VCF format, allowing the user to estimate allele frequencies, levels of linkage disequilibrium and various quality control metrics.

VI. THE SEQUENCE ALIGNMENT/MAP FORMAT AND SAM TOOLS

Li *et al*., explains The Sequence Alignment/Map (SAM) format is a generic alignment format for storing read alignments against reference sequences, supporting short and long reads (up to 128 Mbp) produced by different sequencing platforms. It is flexible in style, compact in size, efficient in random access and is the format in which alignments from the 1000 Genomes Project are released. SAMtools implements various utilities for post-processing alignments in the SAM format, such as indexing, variant caller and alignment viewer, and thus provides universal tools for processing read alignments. It supports single- and paired-end reads and combining reads of different types, including color space reads from AB/SOLiD. It is designed to scale to alignment sets of 1011 or more base pairs, which is typical for the deep resequencing of one human individual.

The SAM format consists of one header section and one alignment section. The lines in the header section start with character '@', all lines are TAB delimited. In SAM, each alignment line has 11 mandatory fields and a variable number of optional fields. They must be present but their value can be a '*' or a zero (depending on the field) if the corresponding information is unavailable. The optional fields are presented as key-value pairs in the format of TAG: TYPE: VALUE. They store extra information from the platform or aligner. The standard CIGAR description of pair wise alignment defines three operations: 'M' for match/mismatch, 'I' for insertion compared with the reference and 'D' for deletion. The extended CIGAR proposed in SAM added four more operations: 'N' for skipped bases on the reference, 'S' for soft clipping, 'H' for hard clipping and 'P' for padding. These support splicing, clipping, multi-part and padded alignments.

To improve the performance, a companion format Binary Alignment/Map (BAM), which is the binary representation of SAM and keeps exactly the same information as SAM. BAM is compressed by the BGZF library, a generic library developed by us to achieve fast random access in a zlib-compatible compressed file. A SAM/BAM file can be unsorted, but sorting by coordinate is used to streamline data processing and to avoid loading extra alignments into memory. A position-sorted BAM file can be indexed. UCSC binning scheme and simple linear indexing is combined to achieve fast random retrieval of alignments overlapping a specified chromosomal region. SAMtools is a library and software package for parsing and manipulating alignments in the SAM/BAM format. It is able to convert from other alignment formats, sort and merge alignments, remove PCR duplicates, generate per-position information in the pileup format, call SNPs and short index variants, and show alignments in a text-based viewer. For the example alignment of 112 Gbp Illumina GA data, SAMtools took about 10 h to convert from the MAQ format and 40 min to index with <30MB memory. Conversion is slower mainly because compression with zlib is slower than decompression.

VII. BEDTOOLS: A FLEXIBLE SUITE OF UTILITIES FOR COMPARING GENOMIC FEATURES

Quinlan *et*. al., introduces a new software suite for the comparison, manipulation and annotation of genomic features in Browser Extensible Data (BED) and General Feature Format (GFF) format. BEDTools also supports the comparison of sequence alignments in BAM format to both BED and GFF features. The tools are extremely efficient and allow the user to compare large datasets (e.g. next-generation sequencing data) with both public and custom genome annotation tracks. BEDTools can be combined with one another as well as with standard UNIX commands, thus facilitating routine genomics tasks as well as pipelines that can quickly answer intricate questions of large genomic datasets. BEDTools was written in C++. BEDTools can easily find out common base pairs the term called as 'intersecting' or 'overlapping'. It also supports UNIX environment and works seamlessly with existing UNIX utilities. BEDTools incorporates the genome-binning algorithm used by the UCSC genome browser to identify overlapping features. Few BEDTools supports the wide range of operations like pairToBed, bamToBed, pairToPair, sortBed, linksBed, complementBed, fastaFromBed, intersectBed.

VIII. A FRAMEWORK FOR VARIATION DISCOVERY AND GENOTYPING USING NEXT GENERATION DNA SEQUENCING DATA

Recent advances in sequencing technology make it possible to comprehensively catalogue genetic variation in population samples, creating a foundation for understanding human disease, ancestry and evolution. The amounts of raw data produced are prodigious and many computational steps are required to translate this output into high-quality variant calls. The analytic framework supports to discover and genotype variation among multiple samples simultaneously that achieves sensitive and specific results across five sequencing technologies and three distinct, canonical experimental designs. The process includes (1) initial read mapping; (2) local realignment around indels; (3) base quality score recalibration; (4) SNP discovery and genotyping to find all potential variants; and (5) machine learning to separate true segregating variation from machine artifacts common to next-generation sequencing technologies. The framework phases are as follows:

Phase 1: raw read data with platform-dependent biases is transformed into a single, generic representation with well-calibrated base error estimates, mapped to their correct genomic origin, and aligned consistently with respect to one another. Mapping algorithms place reads with an initial alignment on the reference genome,

either generated in, or converted to, the technology-independent SAM/BAM reference file format. Next, molecular duplicates are eliminated, initial alignments are refined by local realignment, and then an empirically accurate per-base error model is determined.

Phase 2: the analysis-ready SAM/BAM files are analyzed to discover all sites with statistical evidence for an alternate allele present, among the samples including SNPs, short indels, and CNVs. CNV discovery and genotyping methods, though part of this conceptual framework, are described elsewhere.

Phase 3: technical covariates, known sites of variation, genotypes for individuals, linkage disequilibrium, and family and population structure are integrated with the raw variant calls from phase 2 to separate true polymorphic sites from machine artefacts, and at these sites high-quality genotypes are determined for all samples.

IX. NESTED CONTAINMENT LIST (NCLIST): A NEW ALGORITHM FOR ACCELERATING INTERVAL QUERY OF GENOME ALIGNMENT AND INTERVAL DATABASES

The Nested Containment List (NCList), whose query time is O(nþlog N), where N is the database size and n is the size of the result set. This query algorithm is 5–500-fold faster than other indexing methods, such as MySQL multi-column indexing, MySQLbinning and R-Tree indexing. Performance comparison is given both in simulated datasets and real-world genome alignment databases, across a wide range of database sizes and query interval widths. It also includes an in-place NCList construction algorithm that yields database construction times that are 100-fold faster than other methods available. The NCList data structure appears to provide a useful foundation for highly scalable interval database applications.

X. THE UCSC GENOME BROWSER AND ASSOCIATED TOOLS

The UCSC Genome Browser (http://genome.ucsc.edu) is a graphical viewer for genomic data. Since the early days of the Human Genome Project, it has presented an integrated view of genomic data of many kinds. The Browser presents visualization of annotations mapped to genomic coordinates. The ability to juxtapose annotations of many types facilitates inquiry-driven data mining. Gene predictions, mRNA alignments, epigenomic data from the ENCODE project, conservation scores from vertebrate whole-genome alignments and variation data may be viewed at any scale from a single base to an entire chromosome. The browser also includes many other widely used tools, including BLAT, which is useful for alignments from high-throughput sequencing experiments. Private data uploaded as Custom Tracks and Data Hubs in many formats may be displayed alongside the rich compendium of precomputed data in the UCSC database. The Table Browser is a full-featured graphical interface, which allows querying, filtering and intersection of data tables. The Saved Session feature allows users to store and share customized views, enhancing the utility of the system for organizing multiple trains of thought. Binary Alignment/Map (BAM), Variant Call Format and the Personal Genome Single Nucleotide Polymorphisms (SNPs) data formats are useful for visualizing a large sequencing experiment (whole-genome or whole-exome), where the differences between the data set and the reference assembly may be displayed graphically. Support for high-throughput sequencing extends to compact, indexed data formats, such as BAM, bigBed and bigWig, allowing rapid visualization of large datasets from RNA-seq and ChIP-seq experiments via local hosting. The basic paradigm of the UCSC Genome Browser is to show as much high quality, whole genome annotation data as possible and enable researchers to use their expertise to interpret data themselves. The Genome Browser displays a wide variety of data juxtaposed onto a common coordinate system, providing a high degree of control over what is viewed and how it is configured.

XI. THE GENOME ANALYSIS TOOLKIT: A MAPREDUCE FRAMEWORK FOR ANALYZING NEXT-GENERATION DNA SEQUENCING DATA

Genome Analysis Toolkit (GATK), a structured programming framework designed to ease the development of efficient and robust analysis tools for next-generation DNA sequencers using the functional programming philosophy of MapReduce. The GATK provides a small but rich set of data access patterns that encompass the majority of analysis tool needs. Separating specific analysis calculations from common data management infrastructure enables us to optimize the GATK framework for correctness, stability, and CPU and memory efficiency and to enable distributed and shared memory parallelization.The GATK was designed using the functional programming paradigm of MapReduce. This approach makes a contract with the developer, in which analysis tools are constructed so that the underlying framework can easily parallelize and distribute processing; this methodology has been used by companies like Google and Yahoo! to manage massive computing infrastructures in a scalable way. MapReduce divides computations into two separate steps; in the first, the larger problem is subdivided into many discrete independent pieces, which are fed to the map function; this is followed by the reduce function, joining the map results back into a final product. Calculations like SNP discovery and genotyping naturally operate at the map level of MapReduce, since they perform calculations at each locus of the genome independently. On the other hand, calculations that aggregate data over multiple points in the genome, such as peak calling in chromatin immune precipitation with massively parallel sequencing (ChIP-seq) experiments, would utilize the reduce function of MapReduce to integrate the heights of read pileups across loci to detect sites of transcriptional regulation (Pepke et al. 2009). The GATK is structured

into traversals, which provide the division and preparation of data, and analysis modules (walkers), which provide the map and reduce methods that consume the data. The traversal provides a succession of associated bundles of data to the analysis walker, and the analysis walker consumes these bundles of data, optionally emitting an output for each bundle to be reduced. Since many analysis methods for next-generation sequencing data have similar access patterns, the GATK can provide a small but nearly comprehensive set of traversal types that satisfy the data access needs of the majority of analysis tools. The small number of these traversal types, shared among many tools, enables the core GATK development team to optimize each traversal for correctness, stability, CPU performance, and memory footprint and in many cases allows them to automatically parallelize calculations.

XII. GENCODE: THE REFERENCE HUMAN GENOME ANNOTATION FOR THE ENCODE PROJECT

The GENCODE Consortium aims to identify all gene features in the human genome using a combination of computational analysis, manual annotation, and experimental validation. Since the first public release of this annotation data set, few new protein-coding loci have been added, yet the number of alternative splicing transcripts annotated has steadily increased. The GENCODE 7 release contains 20,687 protein-coding and 9640 long noncoding RNA loci and has 33,977 coding transcripts not represented in UCSC genes and RefSeq. It also has the most comprehensive annotation of long noncoding RNA (lncRNA) loci publicly available with the predominant transcript form consisting of two exons. Completed transcript annotations are 35% of transcriptional starts sites are supported by CAGE clusters and 62% of protein-coding genes have annotated polyA sites. Over one-third of GENCODE protein coding genes are supported by peptide hits derived from mass spectrometry spectra submitted to Peptide Atlas. New models derived from the Illumina Body Map 2.0 RNA-seq data identify 3689 new loci not currently in GENCODE, of which 3127 consist of two exon models indicating that they are possibly unannotated long noncoding loci. GENCODE 7 is publicly available from gencodegenes.org and via the Ensembl and UCSC Genome Browsers.

XIII. ENSEMBL 2013

The Ensembl project provides genome information for sequenced chordate genomes with a particular focus on human, mouse, zebrafish and rat. Ensembl data are accessible through the genome browser at http://www.ensembl.org and through other tools and programmatic interfaces. Ensembl supports 70 species including 61 species fully supported on our main site. Of these, full gene annotations for 58 chordates and have imported annotation data for three non-chordate model organisms (Saccharomyces cerevisiae, Caenorhabditis elegans and Drosophila melanogaster) to facilitate comparative analysis. Five new species were included during the past year with full support: Atlantic cod (Gadus morhua), coelacanth (Latimeria chalumnae), ferret (Mustela putorius furo), Nile tilapia (Oreochromis niloticus) and Chinese softshell turtle (Pelodiscus sinensis). An additional nine species are currently available with limited support on the Ensembl.

XIV. ACCURATE WHOLE HUMAN GENOME SEQUENCING USING REVERSIBLE TERMINATOR CHEMISTRY

DNA sequencing yields an unrivalled resource of genetic information. Characterization is possible in individual genomes, transcriptional states and genetic variation in populations and disease. Until recently, the scope of sequencing projects was limited by the cost and throughput of Sanger sequencing. The raw data for the 3 billion base (3 gigabase, Gb) human genome sequence, completed in 20041, was generated over several years for ~$300 million using several hundred capillary sequencers. More recently an individual human genome sequence has been determined for ~$10 million by capillary sequencing 2. Several new approaches at varying stages of development aim to increase sequencing throughput and reduce cost 3–6. They increase parallelisation dramatically by imaging many DNA molecules simultaneously. One instrument run produces typically thousands or millions of sequences that are shorter than capillary reads. Another human genome sequence was recently determined using one of these approaches7. However, much bigger improvements are necessary to enable routine whole human genome sequencing in genetic research. Massively parallel synthetic sequencing approach that transforms our ability to use DNA and RNA sequence information in biological systems. Demonstration utility includes re-sequencing an individual human genome to high accuracy. This approach delivers data at very high throughput and low cost, and enables extraction of genetic information of high biological value, including single nucleotide polymorphisms (SNPs) and structural variants.

XV. ANALYSIS AND DESIGN OF RNA SEQUENCING EXPERIMENTS FOR IDENTIFYING ISOFORM REGULATION

Through alternative splicing, most human genes express multiple isoforms that often differ in function. To infer isoform regulation from high-throughput sequencing of cDNA fragments (RNA-seq), the mixture-of-isoforms (MISO) model, a statistical model that estimates expression of alternatively spliced exons and isoforms and assesses confidence in these

estimates. Incorporation of mRNA fragment length distribution in paired-end RNA-seq greatly improved estimation of alternative-splicing levels. MISO also detects differentially regulated exons or isoforms. Application of MISO implicated the RNA splicing factor hnRNP H1 in the regulation of alternative cleavage and polyadenylation, a role that was supported by UV crosslinking– immunoprecipitation sequencing (CLIP-seq) analysis in human cells. This result provides a probabilistic framework for RNA-seq analysis, give functional insights into pre-mRNA processing and yield guidelines for the optimal design of RNA-seq experiments for studies of gene and isoform expression.

XVI. ISAX: DISK-AWARE MINING AND INDEXING OF MASSIVE TIME SERIES DATASETS

Current research in indexing and mining time series data has produced many interesting algorithms and representations. The algorithms and the size of data considered have generally not been representative of the increasingly massive datasets encountered in science, engineering, and business domains. This work, introduces a novel multi-resolution symbolic representation which can be used to index datasets which are several orders of magnitude larger than anything else considered in the literature. Simple tree-based index structure which facilitates fast exact search and orders of magnitude faster, approximate search. For example, with a database of one-hundred million time series, the approximate search can retrieve high quality nearest neighbors in slightly over a second, whereas a sequential scan would take tens of minutes. The experimental evaluation demonstrates that the representation allows index performance to scale well with increasing dataset sizes. Additionally, it provides analysis concerning parameter sensitivity, approximate search effectiveness, and lower bound comparisons between time series representations in a bit constrained environment. Also this paper involves how to exploit the combination of both exact and approximate search as sub-routines in data mining algorithms, allowing for the exact mining of truly massive real world datasets, containing tens of millions of time series.

XVII. RAPID STORAGE AND RETRIEVAL OF GENOMIC INTERVALS FROM A RELATIONAL DATABASE SYSTEM USING NESTED CONTAINMENT LISTS

Efficient storage and retrieval of genomic annotations based on range intervals is necessary, given the amount of data produced by next-generation sequencing studies. The indexing strategies of relational database systems (such as MySQL) greatly inhibit their use in genomic annotation tasks. This has led to the development of stand-alone applications that are dependent on flat-file libraries. This paper introduces MyNCList, an implementation of the NCList data structure within a MySQL database. MyNCList enables the storage, update and rapid retrieval of genomic annotations from the convenience of a relational database system. Range-based annotations of 1 million variants are retrieved in under a minute, making this approach feasible for whole-genome annotation tasks.

XVIII. CONCLUSION

In this survey paper, we try to express various file formats for biological data and their tools available in genome research area are given. But still there are some file formats are not supported by some tools so there is some mechanism is require to support remaining file formats.The volume of data being produced in biological research is growing rapidly, but the tools available to end users to process data are still mostly based on parsing plain text. This approach is very inefficient, and leads to several undesirable outcomes. Firstly, and most obviously, a researcher's productivity is inevitably constrained while waiting several hours for the result of a simple calculation. Without flexible indexing, working with a subset of a data file usually requires the creation of another file consisting of the subset in question, requiring extra storage and maintenance. Additionally, code quality is reduced, since testing over the entire dataset is infeasible and it is less likely that the effects of changing arbitrary analysis parameters will be systematically examined.

The classical approach to solving problems of this type is to use a relational database, which provides sophisticated data management techniques. However, relational databases are unsuitable for storing static datasets as they are complex to use and incur many unnecessary overheads. That's why mechanism should inculcate database technology like packed binary storage of values; random access to rows; general purpose indexing without additional complexities and overheads. With further extension we can able to add compression feature with well efficient manner. It is straightforward because it is far simpler and does not need to be compatible with the relational model.

REFERENCES

[1] Li H: Tabix: fast retrieval of sequence features from generic TAB-delimited files. Bioinformatics 2011, 27(5):718–719.

[2] Seltzer M, Bostic K: Berkeley, DB. In The Architecture of Open Source Applications, Volume 1. Edited by Brown A, Wilson G. ISBN 978-1-257-63801-7 2012.

[3] Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST, McVean G, Durbin R, 1000 Genomes Project Analysis Group: The variant call format and VCFtools. Bioinformatics 2011, 27(15):2156–2158.
[4] Kent WJ, Zweig AS, Barber G, Hinrichs AS, Karolchik D: BigWig and BigBed: enabling browsing of large distributed datasets. Bioinformatics 2010, 26(17):2204–2207.
[5] Alekseyenko,A.V. and Lee,C.J. (2007) Nested containment list (NCList): a new algorithm for accelerating interval query of genome alignment and interval databases. Bioinformatics, 23, 1386–1393.
[6] Li,H. et al. and 1000 Genome Project Data Processing Subgroup (2009) The Sequence Alignment/Map (SAM) Format and SAMtools. Bioinformatics, 25, 2078–2079.
[7] Rhead,B. et al. (2010) The UCSC Genome Browser database: update 2010. Nucleic Acids Res., 38, D613–D619.
[8] McKenna,A.H. et al. (2010) The genome analysis toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Res., 20, 1297–1303.
[9] Bentley,D.R. et al. (2008) Accurate whole human genome sequencing using reversible terminator chemistry. Nature, 456, 53–59.
[10] Kent,W.J. et al. (2002) The human genome browser at UCSC. Genome Res., 12, 996–1006.
[11] Quinlan,A.R. and Hall,I.M. (2010) BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics, 26, 841–842.
[12] Raney,B.J. et al. (2011) ENCODE whole-genome data in the UCSC genome browser (2011 update). Nucleic Acids Res., 39, D871–D875.
[13] Shieh,J. and Keogh,E. (2009) iSAX: disk-aware mining and indexing of massive time series datasets. Data Min. Knowl. Disc., 19, 24–57.
[14] Harrow,J. et al. (2012) GENCODE: the reference human genome annotation for the ENCODE Project. Genome Res., 22, 1760–1774.
[15] Katz,Y. et al. (2010) Analysis and design of RNA sequencing experiments for identifying isoform regulation. Nat. Methods, 7, 1009–1015.
[16] Kuhn,R.M. et al. (2013) The UCSC genome browser and associated tools. Brief. Bioinform., 14, 144–161.
[17] DePristo MA, Banks E, Poplin R, Garimella KV, Maguire JR, Hartl C, Philippakis AA, del Angel G, Rivas MA, Hanna M, McKenna A, Fennell TJ, Kernytsky AM, Sivachenko AY, Cibulskis K, Gabriel SB, Altshuler D, Daly MJ: A framework for variation discovery and genotyping using next generation DNA sequencing data. Nat Genet 2011, 43:491-498.
[18] Flicek,P. et al. (2013) Ensembl 2013. Nucleic Acids Res., 41, D48–D55.

