Storage aware data management system for Genomics

Zeeshan Ali Shah
University of Derby School of Computing and
Mathematics,z.shah1@unimail.derby.ac.uk

Mohsen Farid
University of Derby School of Computing and
Mathematics, http://www.derby.ac.uk

ABSTRACT

In recent years, nucleotide sequencing has become increasingly instrumental in both research and clinical settings. This has led to explosive growth in sequencing data produced worldwide along with an increase in complex analysis algorithms. As the amount of data and analysis increases, so does the need for automated solutions for processing and analysis. The concept of workflows has gained favor in the bioinformatics community, but there is little in the scientific literature describing end-to-end operational automation systems. We provided an automation system that aims at providing a solution to the genomics related operational challenges that face sequencing of both research and clinical facilities. We built on existing open-source technologies, with a modular design allowing for a community-driven effort to create plug and play services. In this research, we describe the system and elaborate on the underlying conceptual framework. Which can be reduced to 3 conceptual levels: Data tagging (using metadata automation), Classifying Storage systems (the steps involved in the classification of storage systems), and execution (using a series of rules to move data around on an operational level).

CCS CONCEPTS

• **Information Systems-Data management systems**;

KEYWORDS

meta data, automation, genomics operational pipeline

ACM Reference Format:
Zeeshan Ali Shah and Mohsen Farid. 2023. Storage aware data management system for Genomics. In *2023 5th International Conference on Big-data Service and Intelligent Computation (BDSIC 2023), October 20–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3633624.3633628

1 INTRODUCTION

Genomics is commonly defined as the study of the complete genetic material of any given organism [8]. This hereditary material is deoxyribonucleic acid (DNA) in the shape of a double helix. DNA itself consists of four different kinds of chemical bases called nucleotides: Adenine, Thymine, Guanine, and Cytosine [6]. Pairs of these four bases are twisted into a ladder shape, also known as the double helix. More, in terms of the pairs, adenine and thymine exclusively bond with each other and the same can be said for guanine and cytosine. What makes each human unique is the way the nucleotides are arranged; each stretch of DNA is arranged in different orders and is of different lengths [3]. Sometimes, however, the enzymes responsible for the process of DNA translation and transcription make mistakes and lead to often time deadly mutations [10], or alterations from the normal genome code. For example, the change of a single nucleotide base from the normal "AGCT**T**TGC" to "AGCT**A**TGC" could lead to negative health effects. Interpretation of genetic material holds the key to the future of medicine [5]; it leads to the beginning of an era defined by Genomics medicine, as each human as his or her own personal sequence of DNA and therefore, his or her own personal genome, filled with various mutations, some helpful and some harmful.

In a genomics sequencing facility, there are instruments that take biological samples from diverse organisms as input and generate the computer representation of their genomics sequences as output. The underlying processes on how to perform the sequencing are purely platform-specific [2]. The focus in this thesis will be on the output sequences, where the in-silico data management and its handling intricacies begin [7], as opposed to upstream wet-lab sample processing.

[11] Genomics based medicine, usually referred to as genomics medicine, became an important component in the healthcare system. This could not have been the case without the recent advancements in next generation sequencing (NGS) technology, which reduced the cost and time of reading the genome. NGS is currently used in the clinic to find variants (mutations) related to the disease to improve the diagnosis, prognosis, or to find optimized treatment plans. For computational scientists, the wide use of NGS in the clinic has introduced new challenges. The clinical grade data analysis requires more optimized algorithms to reach reliable results, which accordingly increases the running time. Moreover, to reach a list of variants with the necessary information for the clinic, a sophisticated computational workflow of many software tools should be used. The input to this workflow is the list of NGS reads and the output is the list of significant annotated variants related to the disease. In this workflow, the programs run according to certain dependency plan and the results of one program is fed to the next one through intermediate files.

The output of an NGS machine is a large set of short reads (DNA fragments) [9]. The number of these reads depends on the technology and the model of the NGS instrument. For Ion technology, one expects around 80 million reads per run for the Ion Proton model. For Illumina technology, one expects up to 20 billion reads per run for the recent NovaSeq model. Processing such huge number of reads entails huge I/O operations, especially when a workflow of multiple independent programs is used. This causes two problems: First, a considerable fraction of the analysis time is spent in reading/writing of the data.

BDSIC 2023, October 20–22, 2023, Singapore, Singapore
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0892-3/23/10.
https://doi.org/10.1145/3633624.3633628

Second, such considerably amount of time spent by researchers to bring data in high-speed medium which is a manual operation and which interrupts the operation and increases the operational costs. To solve these problems, it is important to automate data management as much as possible.

Fortunately, the recent advancement in data discovery and meta data management coupled with modern automation techniques based on Machine learning makes this possible.

We found out that aforementioned is a data intensive problem, which requires multiple computations on same data sets [1] – It is not like high through put computing (HTC) in which Data can be chunked and distributed as D1, D2, D3 and Tasks T1, T2, T3 executed in their own set of chunks. NGS analysis requires different sets of tasks T1, T2 , T3 executing on same Data Dx . This requires different scheduling algorithms and execution engines than traditional HTC. Let us re-elaborate the process of NGS,

Step-1: Sample is taken from patient, and it is sent to wet-lab for DNA extraction,

Step -2: Extracted DNA processed in sequencing machines such as Ion Protons, Illumina that is called primary analysis, this primary analysis output is in a FASTQ format file.

Step-3: FASTQ file generated from step-2 is aligned with reference genome and converted into Sequence Aligned mapped or brief (SAM File). This is further converted into binary format and coined a new acronym BAM (Binary aligned Mapped file)- This BAM file is used for future analysis and archive for long term by various sequencing centers.

Generated from Step-3 above the BAM file that is usually of size 200-300 GB per run is used for tertiary analysis such as finding mutations, variants and etc. and on every instance, it is to be accessed from underlying storage platform. Finding single mutation such as Cancer would need to access BAM around 20-30 times and as we need to find more mutations and sequence more patients the I/O load increases exponentially.

2 PROBLEM

In massive parallel genomic sequences, the main objective is often to reduce the total wall clock execution time to release the findings, often patients are waiting before the diagnosis begins or even before the doctor prescribes the appropriate medicine specific to the patient's genome. this is often coined as personalized medicine. Rather than simply increasing CPU power or adding more fast storage do not increase the time-to-release. Because so many different factors are present, you cannot expect a linear improvement in performance just by adding more and more nodes or adding more and more storage.

One of the most important factors is the inherent parallelism present in the pipeline from an infrastructure point of view, many additional factors can also contribute to improved performance. Such as what storage systems should be better suited to that particular stage.

Usually, the decision to select and move the date between different genomics analysis stages need human intervention and is often semi-scripted which results in a delay of the final analysis. This is because the genomics analysis is a multistep process which is often called workflow also the storage system is not only classified as space available but rather there are multiple factors involved such as the network speed, the disk performance, and resiliency. Selection of appropriate storage medium on the specific stage of analysis could not be done manually as we see in most current situations, rather we suggest that it could be automated with the help of a rule engine. This is further explained in later chapters.

Since the logic to process data manually would require a substantial human effort, it is desirable to put an automated yet flexible system in place that can serve bioinformatics research. In particular, repetitive and error-prone steps that need manual attention from specialists should be avoided as much as possible by software means.

We are focusing to remove above barriers of selecting correct storage medium with velocity of data. Not bothering about the semantic of data and its format. This will give the strength of being not only data format agnostic but also storage layer agnostic, as the date aware engine itself would decide these.

2.1 Proposed solution

We are proposing a Data Aware pipeline, which would bring the data in advance into designated compute environment so that task can start immediately. This will reduce the NGS analysis time that is the core requirement for Genomics medicine.

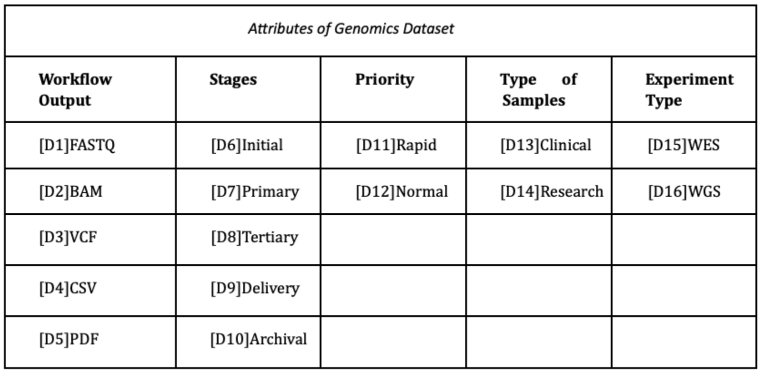
These systems need to be designed not only to support the analysis of data but to address additional aspects associated with operating a genomics operational pipeline. Examples include automatically starting data processing when a sequencing run has finished, do the quality analysis once the primary analysis finished, further do the secondary and tertiary analysis with appropriate algorithms according to the initial findings and require by the patient later move the data to archive to remote storage with selective data removal when needed. These operational aspects have not been thoroughly investigated in the scientific literature but are essential when taking a bird's-eye view of the complete process of refining raw genomics data to scientific results on a high-throughput scale. Tackling these issues involves examination of how higher-level orchestration, integration, and management of workflows can be done in an efficient yet flexible manner, while providing a clear enough understanding of the system so that changes can be implemented with minimal mental overhead and risk of breaking existing functionality.

We fill a niche by providing a systematic way of approaching the operational aspects of data management and analysis of genomics data.

One example of a system addressing the operational challenges outlined above in the context of a sequencing core facility is described by Cuccuru et al. [14]. They describe a system with a central Automator that handles orchestration of the processes in an event-based manner, using the Galaxy platform [15] as a separate workflow manager. The Galaxy platform provides a web-based interface, making bioinformatic analysis accessible to users who lack the training to use command line tools. The system's Automator is based on daemons monitoring a RabbitMQ [16] based event queue.

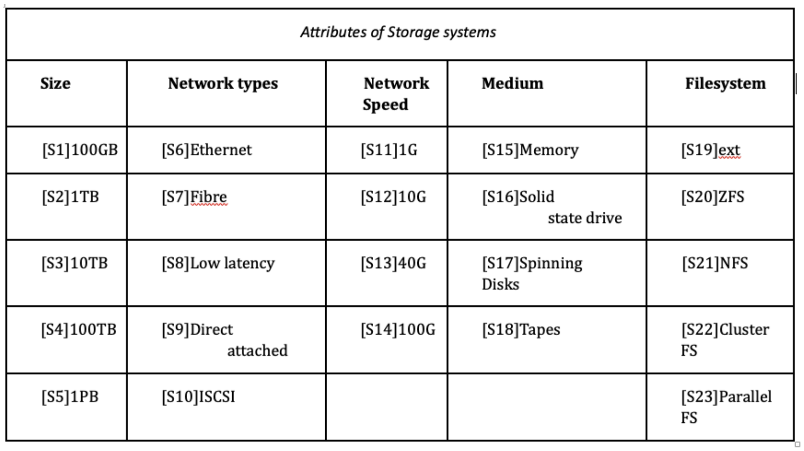
Other system shares ideas with the Arteria system [4], but it does not involve tagging of data with meta data and furthermore it lacks the classification of storage system which as we stated is one requirement for automating the data movement.

Table 1: Attributes of Genomics meta-data



Attributes of Genomics Dataset				
Workflow Output	**Stages**	**Priority**	**Type of Samples**	**Experiment Type**
[D1]FASTQ	[D6]Initial	[D11]Rapid	[D13]Clinical	[D15]WES
[D2]BAM	[D7]Primary	[D12]Normal	[D14]Research	[D16]WGS
[D3]VCF	[D8]Tertiary			
[D4]CSV	[D9]Delivery			
[D5]PDF	[D10]Archival			

Table 2: *Attributes of Storage systems*



Attributes of Storage systems				
Size	**Network types**	**Network Speed**	**Medium**	**Filesystem**
[S1]100GB	[S6]Ethernet	[S11]1G	[S15]Memory	[S19]ext
[S2]1TB	[S7]Fibre	[S12]10G	[S16]Solid state drive	[S20]ZFS
[S3]10TB	[S8]Low latency	[S13]40G	[S17]Spinning Disks	[S21]NFS
[S4]100TB	[S9]Direct attached	[S14]100G	[S18]Tapes	[S22]Cluster FS
[S5]1PB	[S10]ISCSI			[S23]Parallel FS

We basic workflow of genomic data movement are described as:

- To store the data attributes in meta-data system
- To record the storage attributes in meta-data system
- Match the data attributes with the best data location.
- Move the data in consideration of its attributes to appropriate storage.
- Automation of above steps to avoid human errors and scaling purposes.

Classification of storage systems in Genomics pipeline involves with different attributes and mechanisms. Genome sequence workflow typically consists of the following:

1.Sequence appliance store data in a centralize location (that storage should resilient enough to tackle the load)

2.The finished primary sequence data (called as run) consists of different samples of patients which range from 1 to 96 usually. This Run copy to primary analysis storage.

3.The tertiary Analysis of genomics needs to move the run either into a cluster with high speed storage or a dedicated FPGA/GPU.

4.Once the tertiary analysis finished the run will be moved to a long-term storage for archival and retrieval purpose.

In above steps the requirements of storage are different with following attributes such mentioned in Table 1

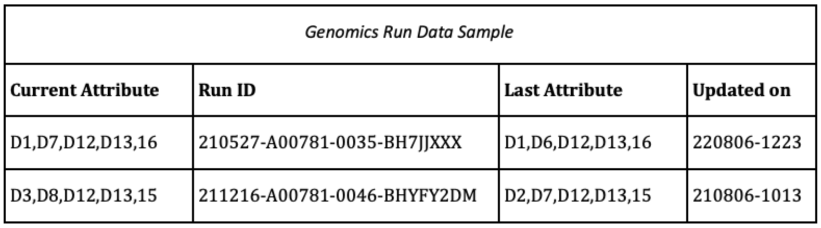
The decision to move the data according to the attributes to the desired storage are so far manual and requires human intervention. This cause delays and prone to human errors. As the sequence cost goes down the rate of doing sequences are growing exponentially. To effectively solve the delay and human error problem the date movements should be automated and intelligently decided in consider with both data and storage attributes.

There should be an automated mechanisms which decides if the data attribute is **X** than move it to Storage **Y** for e.g. Decision of both **X** and **Y** are not simple, instead it requires a sort of machine learning algorithms to make the optimize decision for data movement.

We are working with sensitive clinical data and reliably need to produce results that may have a significant effect on the treatment of patients. We cannot afford to waste time on repetitive manual tasks and the undocumented, unrepeatable changes that come with it.

The idea is to collect the limitation factors or attributes of storage systems as shown in Table 5 and take them in consideration when move the data according to its attributes as show in Table 1

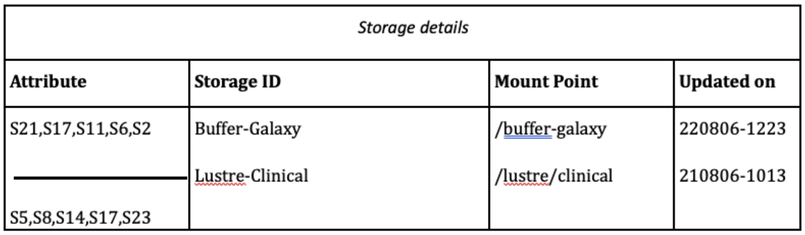
Table 3: Genomics Run Data Sample



Genomics Run Data Sample

Current Attribute	Run ID	Last Attribute	Updated on
D1,D7,D12,D13,16	210527-A00781-0035-BH7JJXXX	D1,D6,D12,D13,16	220806-1223
D3,D8,D12,D13,15	211216-A00781-0046-BHYFY2DM	D2,D7,D12,D13,15	210806-1013

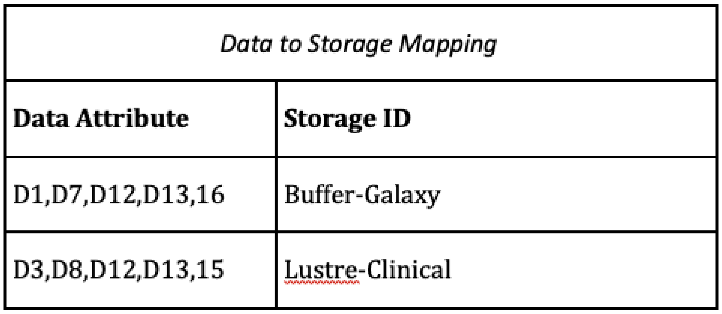
Table 4: Storage details



Storage details

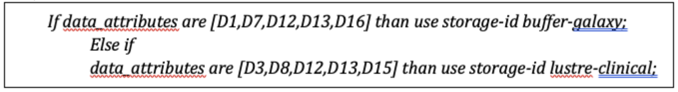
Attribute	Storage ID	Mount Point	Updated on
S21,S17,S11,S6,S2	Buffer-Galaxy	/buffer-galaxy	220806-1223
S5,S8,S14,S17,S23	Lustre-Clinical	/lustre/clinical	210806-1013

Table 5: Data to Storage Mapping



Data to Storage Mapping

Data Attribute	Storage ID
D1,D7,D12,D13,16	Buffer-Galaxy
D3,D8,D12,D13,15	Lustre-Clinical



```
If data_attributes are [D1,D7,D12,D13,D16] than use storage-id buffer-galaxy;
    Else if
    data_attributes are [D3,D8,D12,D13,D15] than use storage-id lustre-clinical;
```

Figure 1: Sample if-this-than-that statement

Ideally the data movement complexity should be hidden from human errors which would lead towards scale-able and a sort of autonomous system.

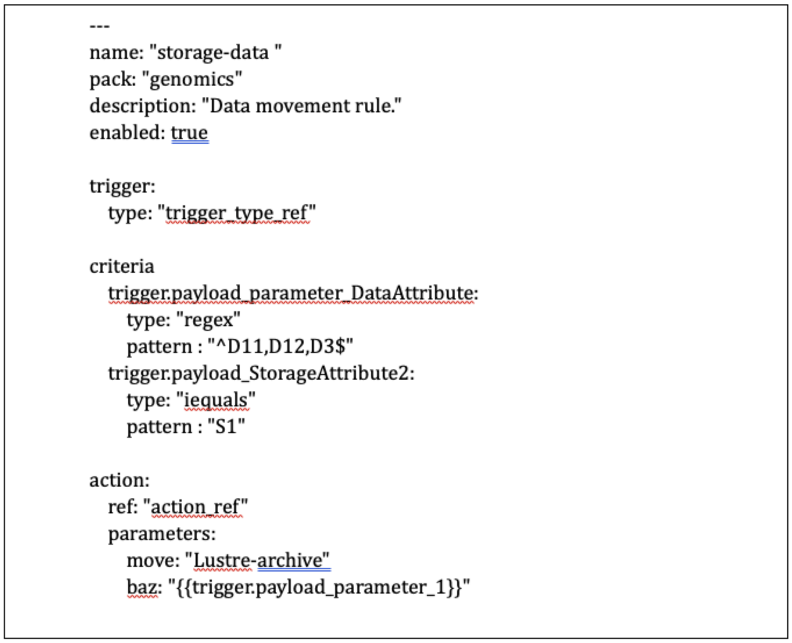
Furthermore, for automation the idea is to use StackStorm for automation but extends it with irods for meta data management.

Our central StackStorm orchestrator queries this service and raises an event every time new sequencing data is available. Based on this event, downstream data processing steps can be initiated for each new dataset.

The irods system perform the metadata tagging over data according to Table 1 and mentioned in previous research [12], the Stack storm utilizes the algorithm to move the data with combination of storage attributes Table [4] and mapping Table [5].

The interconnectivity between irods and stackstorm system is based on microservices which assist in customization. These movement of data happens autonomously, efficiently, reliably, and fully audited.

Above mentioned If-Else statements are not that simple as data attributes keep changing which requires complex rules engine with flexible rule language. Here comes the support of rule engine language from stackstorm which comprises of rules and action. Not just simple if -than-else statement.



```
---
name: "storage-data "
pack: "genomics"
description: "Data movement rule."
enabled: true

trigger:
  type: "trigger_type_ref"

criteria
  trigger.payload_parameter_DataAttribute:
    type: "regex"
    pattern : "^D11,D12,D3$"
  trigger.payload_StorageAttribute2:
    type: "iequals"
    pattern : "S1"

action:
  ref: "action_ref"
  parameters:
    move: "Lustre-archive"
    baz: "{{trigger.payload_parameter_1}}"
```

Figure 2: ***Sample rule in yaml format***

With above system there are no out-of-working-hours delays, no menial tasks for researchers to perform, no inconsistencies due to slight differences in manual steps.

3 RESULTS

Detecting new data utilizes host of useful tools and routines to help with the common tasks of a sequencing center. One of those tools is a folder monitor, which watches a defined directory for new sequencing data and keeps track of its state.

Our central orchestrator queries this service and raises an event every time new sequencing data is available. Based on this event, downstream data processing steps can be initiated for each new dataset..

The data from the sequencing machine first tagged in a meta data system that is recognized by the rule engine. Sample applies with underlying storage systems whose tagging also storage in central meta data management systems.

After a new dataset has been detected, a StackStorm rule calls the script to trigger the data movement. The script calls back to the StackStorm instance allowing further steps to happen without human intervention. In case of a conversion success for instance the data is then automatically transferred to the compute cluster for analysis. Analysis pipeline(s) There are several post processing steps to be taken to ensure quality, comparability, and usefulness of the data to researchers and clinicians beyond our team.

Data and results dissemination the data lifecycle does not stop at the results though. Data and results need to be available in a way that clinicians, collaborators, and other parties can easily access it and it needs to be safe and reliably stored for future reference. We plan to put data distribution modules in place that automate this process and make sure we comply with regulatory requirements as well as community demands.

REFERENCES

[1] Ailamaki, A., DeWitt, D.J., Hill, M.D., Skounakis, M.: Weaving relations for cacheperformance. In: VLDB. vol. 1, pp. 169–180 (2001)

[2] Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J.: A basic localalignment search tool. J. Molecular Biology **215**, 403–410 (1990)

[3] Bhardwaj, R., Sethi, A., Nambiar, R.: Big data in genomics: An overview. Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014 pp. 45–49 (2015). https://doi.org/10.1109/BigData.2014.7004392

[4] Dahlberg, J., Hermansson, J., Sturlaugsson, S., Lysenkova, M., Smeds, P., Ladenvall, C., Guimera, R.V., Reisinger, F., Hofmann, O., Larsson, P.: Arteria: An automation system for a sequencing core facility. GigaScience **8**(12) (12 2019). https://doi.org/10.1093/gigascience/giz135, https://doi.org/10.1093/gigascience/giz135, giz135

[5] D.H. Huson, D., A.F., A., Qi, J., Schuster, S.: MEGAN analysis of metagenomicdata. Genome Research **17**, 377–386 (2007)

[6] Gilbert, J., Dupont, C.: Microbial metagenomics: Beyond the genome. AnnualReview of Marine Science **3**, 347–371 (2010)

[7] Hirsch, M., Mateos, C., Rodriguez, J.M., Zunino, A., Gar´ı, Y., Monge, D.A.: Aperformance comparison of data-aware heuristics for scheduling jobs in mobile grids. 2017 43rd Latin American Computer Conference, CLEI 2017 **2017-Janua**, 1–8 (2017). https://doi.org/10.1109/CLEI.2017.8226474

[8] Kerk, D., Templeton, G., Moorhead, G.: Evolutionary radiation pattern of novelprotein phosphatases revealed by analysis of protein data from the completely sequenced genomes of humans, green algae, and higher plants. Plant Physiology **146**(2), 351–367 (2008)

[9] Koboldt, D., Ding, L., Mardis, E., Wilson, R.: Challenges of sequencing humangenomes. Briefings in Bioinformics **11**(5), 484–498 (2010)

[10] Qi, J., Zhao, F., Buboltz, A., Schuster, S.: inGAP: an integrated nextgenerationgenome analysis pipeline. Bioinformics **26**(1), 127–129 (2010)

[11] Shah, Z.A., El-kalioby, M., Faquih, T.: Exploiting in-memory Systems for Genomic Data Analysis. springerlink . https://doi.org/978-3-319-78723-735,

[12] *https* ://link.springer.com/chapter/10.1007/978 − 3 − 319 − 78723 − 7 35