Redbooks
ibm.com/redbooks

IBM Reference Architecture for Genomics

Speed, Scale, Smarts

Frank Lee, Ph.D.

Infrastructure Solutions

Redpaper

Genomic medicine promises to revolutionize biomedical research and clinical care. By investigating the human genome in the context of biological pathways, drug interaction, and environmental factors, it is now possible for genomic scientists and clinicians to identify individuals at risk of disease, provide early diagnoses based on biomarkers, and recommend effective treatments.

However, the field of genomics has been caught in a flood of data as huge amounts of information are generated by next-generation sequencers and rapidly evolving analytical platforms such as high-performance computing clusters.

This data must be quickly stored, analyzed, shared, and archived, but many genome, cancer and medical research institutions and pharmaceutical companies are now generating so much data that it can no longer be timely processed, properly stored or even transmitted over regular communication lines. Often they resort to disk drive and shipping companies to transfer raw data to external computing center for processing and storage, creating an obstacle for speedy access and analysis of data.

In addition to scale and speed, it is also important for all the genomics information to be linked based on data models and taxonomies, and to be annotated with machine or human knowledge. This smart data can then be factored into the equation when dealing with genomic, clinical, and environmental data, and be made available to a common analytical platform.

To address the challenging needs for speed, scale, and smarts for genomic medicine, an IBM® end-to-end reference architecture has been created that defines the most critical capabilities for genomics computing: Data management (Datahub), workload orchestration (Orchestrator), and enterprise access (AppCenter).

The IBM Reference Architecture for genomics can be deployed with various infrastructure and informatics technologies. IBM has also been working with a growing ecosystem of customers and partners to enrich the portfolio of solutions and products that can be mapped into the architecture.

This IBM Redpaper™ publication describes the following topics:

- Overview of IBM Reference Architecture for Genomics
- Datahub for data management
- Orchestrator for workload management
- AppCenter for managing user interface

For more information, see the following section:

- Additional information and resources

This paper is targeted toward technical professionals (scientists, consultants, IT architects, and IT specialists) responsible for creating and providing life sciences solutions.

Overview of IBM Reference Architecture for Genomics

This section provides an overview of the IBM reference architecture for genomics, and the opportunities and challenges with genomic medicine.

Opportunities and challenges for genomic medicine revolution

Genomic medicine promises to revolutionize biomedical research and clinical care. By investigating the human genome in the context of biological pathways, drug interaction, and environmental factors, it is now possible for genomic scientists and clinicians to identify individuals at risk of disease, provide early diagnoses based on biomarkers, and recommend effective treatments (Figure 1).

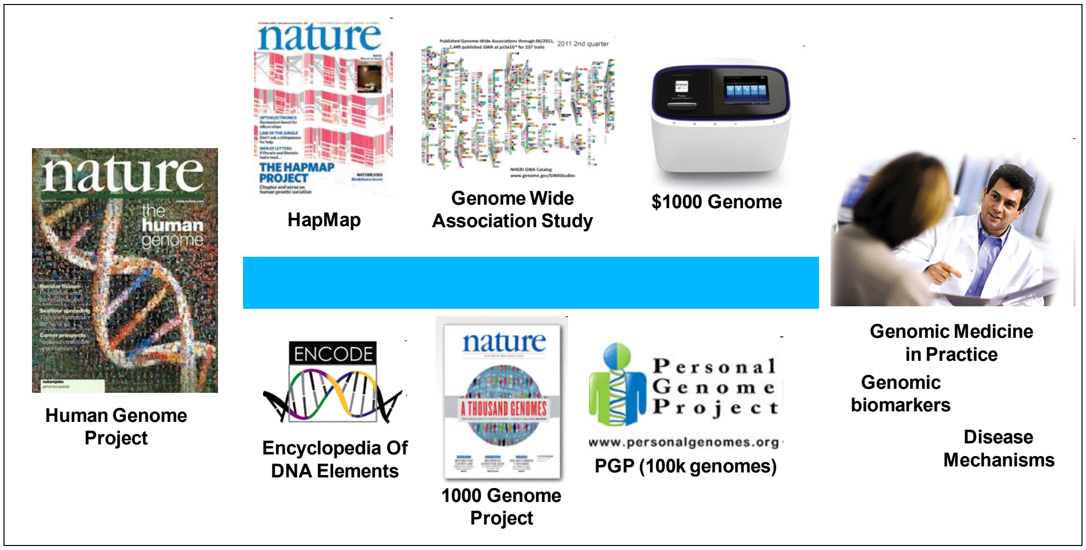

Figure 1 A decade of technological advancement for genomic medicine

Since the Human Genome Project, various projects have started to reveal the mysteries of human genomes and connection to disease mechanism as shown in Figure 1. As sequencing technology continues to advance, the $1000 Genome has recently become a reality.

The Human Genome Project is the first scientific research project to determine the genomic sequence of human being. The project took 13 years and nearly $3B to complete in 2003, and so far remains the largest collaborative biological project. Since then, a series of technological evolutions have taken place in the area of DNA sequencing and large-scale genomic data analysis (Figure 1). The time and cost needed to sequence a full human genome has dropped precipitously, even faster than Moore's law (Figure 2).

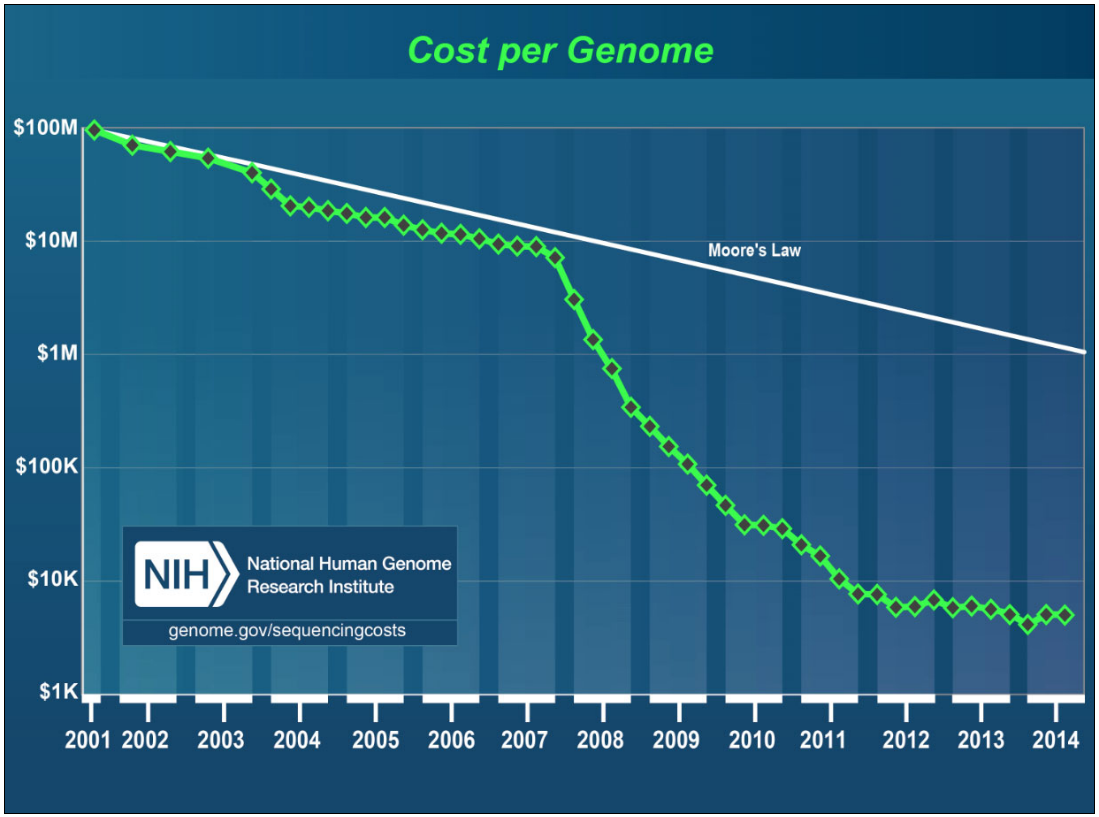

Figure 2 Fast drop of DNA sequencing cost[1]

As an example of advancement of sequencing technologies, Illumina Inc. announced in 2014 a next-generation sequencer, HiSeq X Ten, that will be capable of deciphering 18,000 whole human genome a year at a cost of $1000 per genome. This so-called *Thousand Dollar Genome* technology makes human whole-genome sequencing more affordable and accessible than ever before, and is expected to have immense impact on the health care and life science industry.

The success of new technology and research methods comes with a considerable cost. The field of genomics has been caught in a flood of data as huge amounts of information are generated by next-generation sequencers and rapidly evolving analytical platforms such as high-performance computing clusters:

- Genomic data has doubled every 5 month for the last 8 years.
- The completion of the ENCODE project found that 80% of genome has meaning, so it important to acquire the full genome sequence.

[1] Since 2001, the National Human Genome Research Institute (NHGRI) has tracked the costs associated with DNA sequencing performed at the sequencing centers funded by NIH. This information has served as an important benchmark for assessing improvements in DNA sequencing. The diagram here provides one view of the remarkable improvements in DNA sequencing technologies and data-production pipelines in recent years. (Source: NHGRI, http://www.genome.gov/sequencingcosts/)

- Cancer genomics has revealed a diverse set of genetic variation in cancer cells that need to be tracked and monitored by full genome sequencing, which creates about 1 TB of data per analysis.
- There are genomic sequencing projects started by a growing number of countries such as the US, the UK, China, and Qatar. These projects can easily generate hundreds of PB of sequencing data.

This data needs to be quickly stored, analyzed, shared, and archived, but many genome, cancer, and medical research institutions and pharmaceutical companies are now generating so much data that it can no longer be timely processed, properly stored, or even transmitted over regular communication lines. Often they resort to disk drives and shipping companies to transfer raw data to external computing center for processing and storage, creating an obstacle for speedy access and analysis of data.

In addition to scale and speed, it is also important for all the genomics information to be linked based on data models and taxonomies, and to be annotated with machine or human knowledge. This *smart* data can then be factored into the equation when dealing with genomic, clinical, and environmental data, and made available to a common analytical platform.

IBM Reference Architecture for Genomics

To address the needs for speed, scale, and smarts for genomic medicine, IBM has created an end-to-end reference architecture that defines the most critical capabilities for genomics computing: Data management (*Datahub*), workload orchestration (*Orchestrator*), and enterprise access (*AppCenter*).

To determine the inclusion and priorities for the building blocks of the reference architecture (capabilities and functions) and mapped solutions (hardware and software), follow these three main principles:

1. Software-defined: Defining the infrastructure and deployment model based on software-based abstraction layers for computation, storage, and cloud. This helps *future-proof* the genomic infrastructure for growth and expansion as data volume and computing loads continue to pile up.
2. Data-centric: Meeting the challenge of explosive growth of genomics, imaging, and clinical data with data management capabilities.
3. Application-ready: Integrating a multitude of applications into a consistent environment that provides support for data management, version control, workload management, workflow orchestration, and access for execution and monitoring.

Figure 3 illustrates the IBM Reference Architecture for Genomics (v7.6). The blue box depicts the genomics platform. The green box depicts the translational platform. The purple box depicts the personalized medicine platform. These three platforms for genomic medicine can share common enterprise capabilities: *Datahub* for data management, *Orchestrator* for workload management and *AppCenter* for access management.

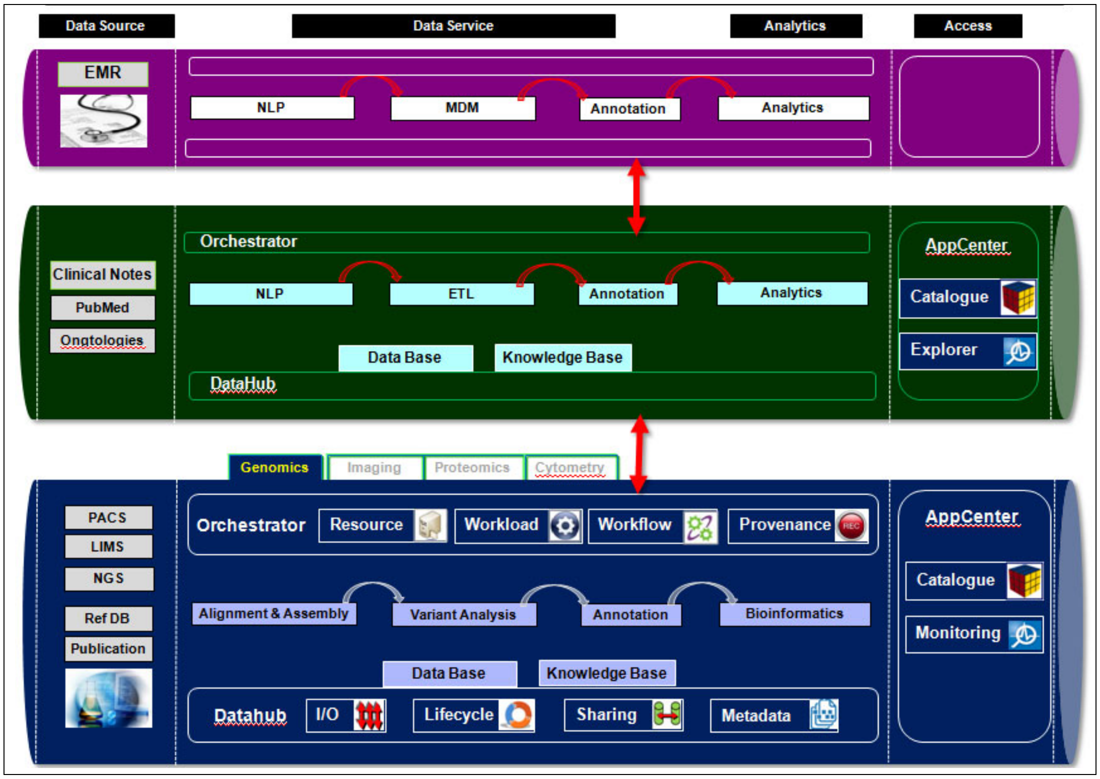

Figure 3 IBM Reference Architecture for Genomics

The IBM Reference Architecture for Genomics first serves as a master plan for composing capabilities for data, workload, and access management into system architecture for platforms such as genomics (*Blue Box*), translational research (*Green Box*) and personalized medicine (*Purple Box*). The system architecture then can be implemented as a genomics computing infrastructure. For the definitions of the acronyms, see “Additional information and resources” on page 18.

Reference Architecture as master plan for deployment

IBM Reference Architecture for genomics can be deployed with various infrastructure and informatics technologies. IBM has also been working with a growing ecosystem of customers and partners to enrich the portfolio of solutions and products that can be mapped into the architecture.

Figure 4 is an overview of the deployment model and some example technologies, solutions, and products that have already been mapped within *Datahub*, *Orchestrator*, and *AppCenter*.

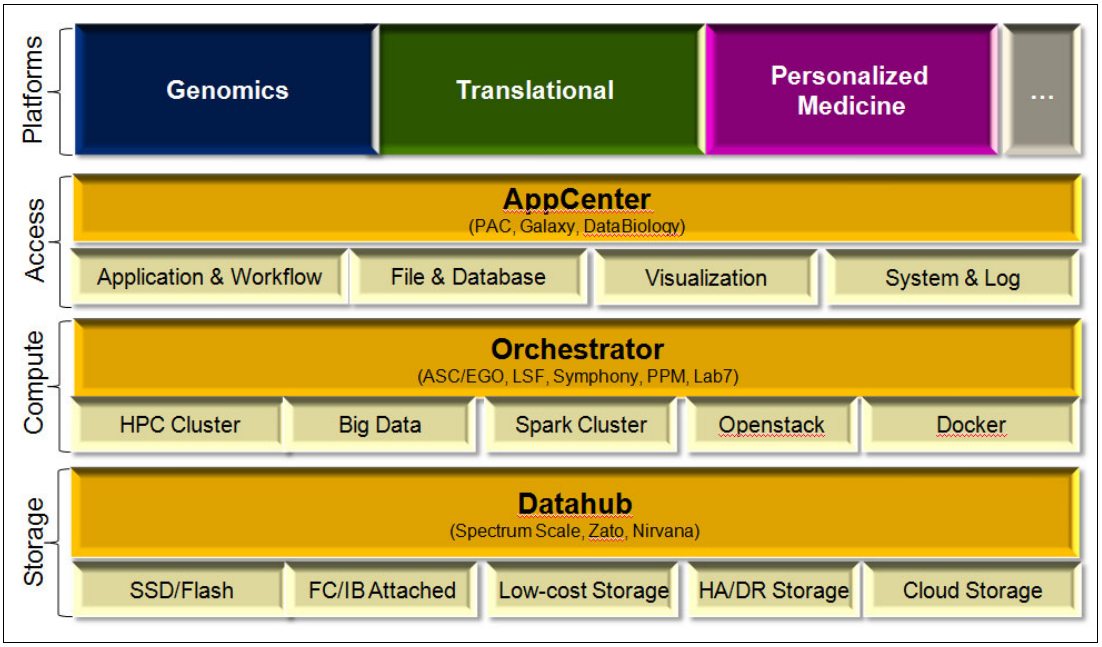

Figure 4 Deployment model for IBM Reference Architecture for Genomics

The infrastructure technologies for storage (SSD, Flash, Disk, Cloud), compute (HPC, Big Data, Spark, Openstack, Docker) and informatics technologies for user access (application/workflow, file protocol, database query, visualization, monitoring) are all managed by the three enterprise capabilities of *Datahub*, *Orchestrator*, and *AppCenter* as shown in Figure 4). Various commercial productions ranging from IBM Spectrum Scale (formally GPFS™) to open source solutions such as Galaxy are mapped to these capabilities. These solutions and products can then be composed into a deployable platform for genomics, translational, or personalized medicine.

Reference Architecture as blueprint for growth

The IBM Reference Architecture for Genomics also serves as blueprint to enable organic growth of platforms and infrastructure by implementing new or scaling existing building blocks that map to varying requirements. These building blocks can be of different types, models, sizing, and even system architecture. Examples include stand-alone servers, cloud-based virtual machines, high-performance computing (HPC) cluster, low-latency networks, scale-out storage systems, big data cluster, tape archive or metadata management system, and so on. For the building blocks to be able to integrate into the blueprint, they need to be compliant with the standards of the reference architecture: Industry-standard data formats, common software framework, and hardware interoperability.

With a master plan and blueprint, the implementation and expansion of genomic infrastructure can be carried out in many flexible ways:

- It can start small: As a software-defined infrastructure, if the key capabilities and function are in place, the sizing of the system, platform, and infrastructure can be quite small to accommodate a limited budget. For example, a clinical sequencing lab can deploy a small system consist of only 1-2 servers, providing a small amount of disk storage with key software for management.

- It can grow very fast and large. As demand for compute and storage grow, the existing infrastructure can be expanded quickly without operational disruption so that new capabilities can be added or scaled to very large size. As an example, Sidra Medical Research Center started building its genomics infrastructure in late 2013 with a small IBM PureSystems® Cluster and some storage. Following the blueprint of the reference architecture, a new building block (60-node HPC Cluster) was added and their existing storage building block was tripled in size by mid-2014. This robust capability enables Sidra to become the genomics computing infrastructure provider for Qatar-based Genome Arabia Project.
- It can be geographically distributed. This is a new capability that was recently introduced into the high-performance computing field and is now fully incorporated into the reference architecture. The sharing and federation are built into *Datahub* and *Orchestrator* seamlessly. Both data and compute resources can now be deployed in different locations while still being accessible to users and available to applications and workflow.

Dozens of world's leading health care and life science organizations have adopted the IBM reference architecture for genomics in support of their integrated research computing infrastructure. In the following sections, this paper describes the key components of the reference architecture and various best practices and lessons learned during many of these projects.

Datahub for data management

Data management is the most fundamental capability for genomics platforms because a huge amount of data needs to be processed at the correct time and place with feasible cost. The temporal factors can range from hours to analyze the data in an HPC system to years when data needs to be recalled from a storage archive for reanalysis. The spatial aspect can span a local infrastructure that provides near-line storage capability to a cloud-based remote cold archive.

Data management challenges

The four *Vs* that define big data also describe genomic data management challenges: Very large data size and capacity (Volume), demanding I/O speed and throughput (Velocity), fast-evolving data types and analytical methods (Variety), and the capability to share and explore large volume of data with context and confidence (Veracity). In this case, the challenges are exacerbated by extra requirements such as regulatory compliance (patient data privacy and protection), provenance management (full versioning and audit trail), and workflow orchestration.

Data volume

Genomic data volume is surging as the cost of sequencing drops precipitously. It is common for an academic medical research center (AMRC) equipped with next-generation sequencing technologies to double its data storage capacity every 6-12 months. Consider a leading AMRC in the New York City (NYC) that started 2013 with 300 TB of data storage. By the end of 2013, the storage volume surged passing 1 PB (1000 TB), more than tripling the amount from 12 months before. What made it even more astonishing was that the speed of growth has been accelerating and continues still today. For some of world's leading genomic medicine projects such as Genome England (UK), Genome Arabia (Qatar), Million Veteran Project (US), and China National GeneBank, the starting points or baseline for data volume are no longer measured in terabytes (TB) but tens and hundreds of petabytes (PB).

Data velocity

The data velocity in a genomic platform can be extremely demanding due to three divergent requirements:

1. Very large files: These files are used to store genomic information from study subject, which can be a single patient or group of patients. There are two main types of such files: Genomic sequence alignment (BAM or Binary Alignment/Map) and genetic variants (VCF or Variant Call File). They are often larger than 1 TB and can take up half of total storage volume for a typical genomic data repository. Additionally, these files are quickly growing larger, often the result of condensing more genomic information from higher resolution coverage (for example, from 30X to 100X for full genome) or a larger study size. As the genomic research evolves from Rare Variant study (variant calling from a single patient) to the Common Variant study, there is an emerging need to make joint variant calling from thousands of patient samples. Consider a hypothetical case provided by the Broad Institute: For 57,000 samples to be jointly called, the input BAM file will be 1.4 PB and the output VCF file will be 2.35 TB, both extremely large in today's standard but potentially commonly used soon.
2. Many small files: These files are used to store raw or temporary genomic information such as output from sequencers (for example, BCL file format from Illumina). They are often smaller than 64 KB and can take up half of total file objects for a typical genomic data repository. Because each file I/O requires two operations for data and metadata, workload that generates or requires access to large number of files creates a distinct challenge that is different from that of large files. In this case, the velocity can be measured in I/O operations per second (IOPS) and they typically reach millions of IOPS for the underlying storage system. Consider a storage infrastructure at a San Diego-based AMRC that was not optimized for massive small file operation. A workload such as BCL conversion (for example, CASAVA, from Illumina) will stall as compute servers are constrained with limited I/O capability, especially IOPS. A benchmark has confirmed that the CPU efficiency drops to single digits because the computing power is being wasted waiting for data to be served. To alleviate this computational bottleneck, IBM researchers developed a data caching method and tool to move I/O operation from disk into memory.
3. Parallel and workflow operation: To scale performance and speed up time to results, the genomics computing is often run as an orchestrated workflow in batch mode. This parallel operation is essential to deliver fast turnaround as more workloads evolve from small-scale targeted sequencing to large-scale full-genome sequencing. With hundreds to thousands of diverse workloads running concurrently in such a parallel computational environment, the requirement for storage velocity as measured in I/O bandwidth and IOPS will be aggregated and will increase explosively. Consider a bioinformatics application from the NYC AMRC. This application can be run in parallel on 2500 compute cores, each writing the output to the disk at a rate of 1 file per second and collectively creating millions of data objects, either 2500 folders each with 2500 files or 14 million files in one directory. This workload is one of many that contributed to a data repository with 600 million objects, including 9 million directories that each contain only one file. Due to the massive amount of metadata, the IOPS load was constraining the overall performance that even a file system command to list files (`ls` in Linux) took several minutes to complete. A parallel application such as the GATK Queue also suffered from poor performance. In early 2014, the file system was overhauled with emphasis on improving the metadata infrastructure. As a result, both bandwidth and IOPS performance were significantly improved and the benchmark showed a 10X speedup of a gene-disease application without any application tuning.

Data variety

There are also many types of data formats to be handled in terms of storage and access. The data formats range from intermediary files created during a multi-step workflow to output files that contain vital genomic information to reference data sets that need to be carefully versioned. The common approach today is to store all this data in online or near-line disk in one storage tier despite the expense of this approach. One practical constraint is the lack of lifecycle management capability for the massive amount of data. If it takes the genomic data repository a long time to scan the file system for candidate files for migration or backup, it becomes impossible to complete this task in a timely fashion. Consider a large US genome center that is struggling to manage its fast-growing data as it adopts Illumina X10 sequencer for full genome sequencing. To complete a scan of the entire file system, it currently takes up to four days, making daily or even longer backup windows impossible. As a result, data is piling up quickly in the single-tier storage and slowing down the metadata scan and performance even further, causing a vicious cycle for data management.

Another emerging challenge for data management is created by the spatial varieties of data. As inter-institutional collaboration becomes more common and a large amount of data needs to be shared or federated, the locality becomes an indispensable character of data. The same data set, especially reference data or output data, can exist in multiple copies in different locations, or in duplicates in the same location due to regulatory compliance requirement (for example, physically isolating a clinical sequencing platform from one for research). In this case, managing the metadata efficiently to reduce data movement or copying will not only reduce cost due to extra storage, but also minimize problems due to versioning and synchronization.

Data veracity

The multi-factorial nature of many complex disorders such as diabetes, obesity, heart disease, Alzheimer's, and Autism Spectrum Disorder (ASD) requires sophisticated computational capabilities to aggregate and analyze large stream of data (genomics, proteomics, imaging) and observation points (clinical, behavioral, environmental, real-world evidence) from a wide range of sources. The development of databases and file repositories that are interconnected based on global data sharing and federated networks bring the promise of innovative and smarter approaches to access and analyze data in unprecedented scale and dimensions. It is in this context that the veracity (trustworthiness) of data enters the equation as an essential element. For example, clinical data (genomics and imaging) needs to be properly and completely de-identified to protect confidentiality of study subject. Genomic data needs to have end-to-end provenance tracking from bench to bedside to provide full audit trail and reproducibility. The authorship and ownership of data needs to be properly represented in a multi-tenancy and collaborative infrastructure. With built-in capability to handle data veracity, a genomic computing infrastructure should enable the researchers and data scientists to share and explore large volume of data with context and confidence.

Datahub functions

To address the challenges of data management for genomics, define an enterprise capability that functions as a scalable and extensible layer for serving data and metadata to all workloads. Name this layer *Datahub* to reflect its critical role as a central hub for data - storing, moving, sharing, and indexing massive amount of genomic raw and processed data. The *Datahub* also manages the underlying heterogeneous storage infrastructure from SSD/Flash to disk to tape to Cloud (Figure 5).

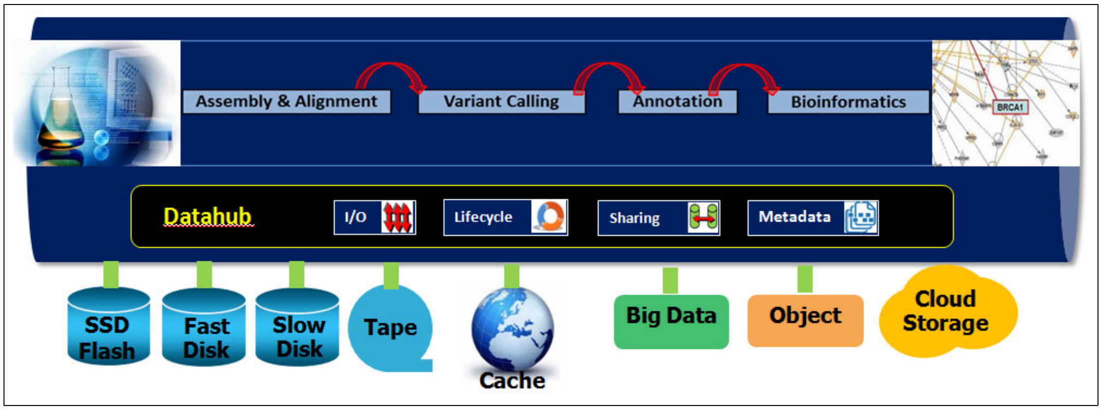

Figure 5 Overview of Datahub

The *Datahub* is the enterprise capability for serving data and metadata to all the workloads (Figure 5). It defines a scalable and extensible layer that virtualizes and globalizes all storage resources under a global name space. *Datahub* is designed to provide four key functions:

1. High-performance data input/output (I/O)
2. Policy-driven information lifecycle management (ILM)
3. Efficient data sharing through caching and necessary replication
4. Large-scale metadata management

For physical deployment, the *Datahub* should support an increasing number of storage technologies as modular building blocks, for example:

- Solid-state disk (SSD) and Flash storage system
- High-performance fast disks
- Large-capacity slow disks (4 TB per drive)
- High-density and low-cost tape library
- External storage cache that can be locally or globally distributed
- Big data storage based on Hadoop
- Cloud-based external storage

Four key functions are mapped to the *Datahub*:

1. I/O management: This *Datahub* function addresses the need for large and scalable I/O capability. There are two dimensions to the capability: I/O bandwidth for serving large-size files such as BAM, and IOPS for serving a large number of small files such as BCL and FASTQ. Due to these divergent needs, the traditional one-size-fit-all architecture struggles to deliver the performance and scalability. *Datahub* I/O management solves this challenge by introducing the pooling concept to separate the I/O operations for metadata/small files from those for the large files. These storage pools, while mapped to different underlying hardware to deliver optimal storage performance, are still unified at the file system level to provide a single global name space for all the data and metadata, and are transparent to users.

2. Lifecycle management: This *Datahub* function addresses the need for managing the lifecycle of data from creation to deletion or preservation. We use the analogy of *temperature* to describe the stages and timeliness that data needs to be captured, processed, moved, and archived. When raw data is captured from instruments such as high-throughput sequencers, they are the *hottest* in temperature and need to be processed by a high-performance computing cluster with robust I/O performance (so-called *scratch* storage). After initial processing, the raw and processed data becomes *warm* in temperature as it will take a policy-based process to determine the final destiny: Deletion, preservation in a long-term storage pool, or archived. The process takes into account file type, size, usage (for example, last accessed time by user), and system utilization information. Any files that meet the requirement for action will be either deleted or migrated from one storage pool to another, typically one that has a larger capacity, slower performance, and much lower cost. One such target tier can be a tape library. Coupled with storage pooling and low-cost media such as tape, this function enables the efficient usage of underlying storage hardware and drastically lower the total cost of ownership for *Datahub*-based solution.
3. Sharing management: This *Datahub* function addresses the need for data sharing within and across logical domains of storage infrastructure. As genomic sample and reference data sets grow larger (in some cases exceeding 1 PB per workload), it becomes increasingly difficult to move and duplicate data for sharing and collaboration purposes. To minimize the impact of data duplication while still enabling data sharing, *Datahub* introduces three elements under sharing management:
   a. Storage multi-clustering such that one compute cluster can access a remote system directly and only pull data/storage on demand.
   b. Cloud data caching such that metadata index and full data set for a specific data repository (host) can be selectively and asynchronously cached on a remote (client) system for fast local access.
   c. Database federation that enables secured federation among distributed databases. In all these functions, the data sharing and movement can occur over a private high-performance network or a wide area network such as the Internet. The technology accounts for the security and fault tolerance.
4. Metadata management: This *Datahub* function provides a foundation for the previous three. The ability to store, manage, and analyze billions of metadata objects is a must-have for any data repository that scales beyond petabytes of size, which is increasingly the case for genomics infrastructure. The metadata includes system metadata such as file name, path, size, pool name, creation time, modified or access time, and so on. It can also include custom metadata in the form of key value pairs that applications, workflow, or users can create to associate with the files of interest. This metadata should be efficiently used to accomplish these goals:
   a. Facilitate the I/O management by placing and moving file based on file size, type, or usage
   b. Enable policy-based lifecycle management of data based on information collected from lightning-fast metadata scan
   c. Enable data-caching so the distribution of metadata can be light-weight and have less dependency on networking

Datahub solution and use cases

One of the IBM solutions mapped to *Datahub* is based on the software solution called IBM Spectrum Scale, formerly GPFS. Spectrum Scale is high-performance, scalable, and extensible.

Initially developed and optimized as a high-performance computing parallel file system, Spectrum Scale manages to serve large volumes of data at a high bandwidth and in parallel to all the compute nodes in the computing system. As genomic pipelines can consist of hundreds of applications engaged in concurrent data processing on large number of files, this capability is critical in feeding data to the computational genomics workflow.

Because the genomics pipeline generates a huge amount of metadata and data, the file system, a system pool built upon SSD and Flash disk with high-IOPS capability, can be dedicated to store metadata for files/directories, and in some cases small-size files directly. This drastically improves file system performance and responsiveness to metadata-heavy operations such as listing all files in a directory.

As a file system with a connector to MapReduce, *Datahub* can also serve MapReduce/Big Data jobs on the same set as compute nodes, eliminating the need for and complexity of Hadoop Distributed File System (HDFS).

The policy-based data lifecycle management capability allows *Datahub* to move data from one storage pool to the others, maximizing I/O performance, storage utilization, and minimizing operational cost. These storage pools can range from the high-I/O flash disk to high-capacity storage appliance to low-cost tape media through integration with a tape management solution such as IBM Spectrum Protect (formerly IBM Tivoli® Storage Manager) and IBM Spectrum Archive (formerly IBM Linear Tape File System™).

The increasingly distributed nature of genomics infrastructure also requires data management on a much larger and global scale. Data not only needs to be moved or shared across different sites, their movement or sharing needs to be coordinated with computational workload and workflow. To achieve this, *Datahub* relies on a sharing function based on the Active File Management (AFM) feature of Spectrum Scale. AFM enables the *Datahub* to extend the global name space to multiple sites, allowing them to share a common metadata catalog and a cache copy of home data for a remote client site to access locally. For example, a genomic center can own, operate, and version-control all reference databases or data sets, while the affiliated or partnering sites or centers can access the reference data set through this sharing function. When the centralized copy of database gets updated, so will the cache copies of the other sites.

With *Datahub*, a system-wide metadata engine can also be built to index and search all the genomic and clinical data, enabling powerful downstream analytics and translational research.

Orchestrator for workload management

This section describes the challenges in workload management with genomics, and the use of the *Orchestrator* to help minimize workload management challenges.

Challenge of workload management in Genomics

The genomics workloads can be very complex. There are a growing number of genomic applications with varying degrees of maturity and types of programming models: Many are single-threaded (for example, R) or embarrassingly parallel (for example, BWA) while a few others are multi-threaded or MPI-enabled (MPI BLAST). However, all applications need to work in concert or tandem in a high-throughput and high-performance mode to generate final results.

Orchestrator functions

Through the *Orchestrator*, the IBM Reference Architecture for Genomics defines the capability to orchestrate resources, workload, and workflow as shown in Figure 6. A unique combination of the workload manager and workflow engine links and coordinates a spectrum of computational and analytical jobs into fully automated pipelines that can be easily built, customized, shared, and run on a common platform. This provides the necessary abstraction of applications from the underlying infrastructure such as a high-performance computing cluster with a graphical processor unit (GPU) or a big data cluster in the Cloud.

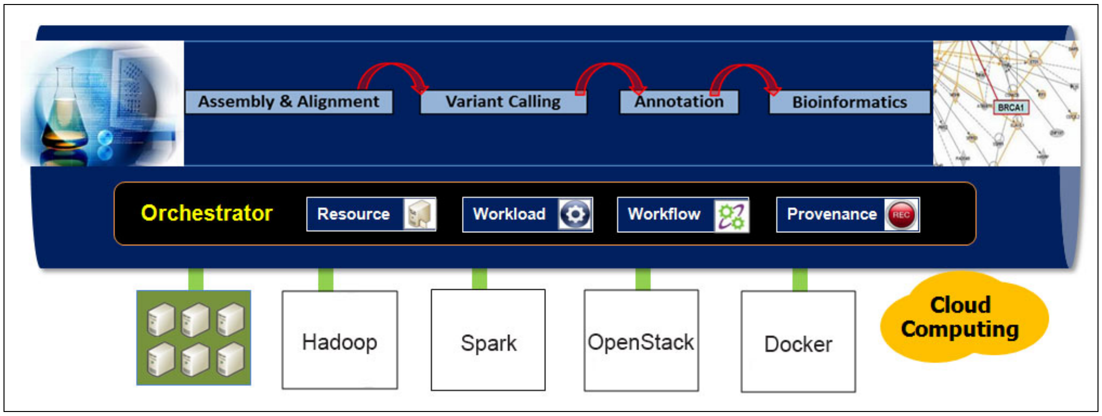

Figure 6 Orchestrator overview

The *Orchestrator* is the enterprise capability for orchestrating resource, workloads, and managing provenance as shown in Figure 6. It is designed to provide four key functions:

1. Resource management by allocating infrastructure towards computational requirement dynamically and elastically
2. Workload management by efficient placement of jobs onto various computational resources such as local or remote clusters
3. Workflow management by linking applications into logical and automated pipelines
4. Provenance management by recording and saving metadata associated with the workload and workflow

By mapping and distributing workloads to elastic and heterogeneous resources (HPC, Hadoop, Spark, Openstack/Docker, Cloud, and so on) based on workflow logic and application requirements (for example, architecture, CPU, memory, I/O), the *Orchestrator* defines the abstraction layer between the diverse computing infrastructure and the fast-growing array of genomic workloads.

Resource Manager

The Resource Manager in the *Orchestrator* functions to allocate computational resources in a policy-driven way to meet the computational needs of genomic workloads. The most commonly used resource is a high-performance computing bare-metal cluster (HPC). The Resource Manager can either one-time provision the resource, or dynamically shift and allocate resources. Just as *Datahub* I/O Management provides a layer of storage services, the Resource Manager provides a fabric of computing services. Additionally, a new type of infrastructure can be added into the resource pool. These include big data Hadoop cluster, Spark cluster, Openstack virtual machine cluster, and Docker cluster.

The ability to manage and shift resources based on the load information from the workload is a requirement for the Resource Manager. As an example, for a genomic infrastructure that is shared between a batch alignment job and Spark machine learning job, as the workload fluctuates during the run time, the Resource Manager should be able to sense the utilization level and shift resources, in the form of computing slots or container, from supporting one job to the other.

Workload Manager

The Workload Manager function in the *Orchestrator* enables the genomic computing resources, as made available by the Resource Manager, to efficiently share, use, and deliver optimal performance to the genomic applications. The Workload Manager should be able to handle demanding, distributed, and mission-critical applications such as Illumina ISSAC CASAVA, bcltofastq, BWA, Samtools, SOAP (Short Oligonucleotide Analysis Package), and GATK. The Workload Manager also needs to be highly scalable and reliable to manage large number of jobs submitted in batches, a common requirement for mid-to-large genomic computing infrastructure. As an example, a genomic computing cluster at a Medical School in New York typically handles 250,000 jobs in the queuing system without crashing or stalling. At some of the world's largest genome centers, the Workload Manager queue can sometimes reach 1 million jobs. For ever-growing numbers of genomics applications with different level of maturity and architectural requirements (for example, CPU, GPU, large-memory, MPI and so on), Workload Manager provides the necessary resource abstraction so that jobs can be transparently handled for submission, placement, monitoring, and logging.

Workflow Engine

This *Orchestrator* function addresses the need for genomic workflow management. The Workflow Engine works to connect jobs into a logical network. This network can be a linear progression of computational pipeline in multiple steps such as sequence alignment, assembly, and variant calling. It can also be a much more complex network with conditional branches or loops based on user-defined criteria and requirements for completion.

The Orchestrator Workflow Engine distinguishes itself from traditional workflow tools with its ability to handle complex workflow dynamically and quickly. Individual workloads or jobs can be defined through a easy to use interface, incorporating variables, parameters, and data definition into a standard workflow template. Many workload types can be easily integrated into the workflow engine: Parallel HPC applications, big data applications, or R scripts for analytics workload. After it is defined and validated, the template can be used by the users to start workflow directly from their workstations or be published in an enterprise portal (*AppCenter*, see “AppCenter for managing user interface” on page 16) to be used by other users.

Orchestrator Workflow Engine (Figure 7 on page 15) can deliver the following additional values:

- Job arrays: To maximize the throughput of the workflow for genomics sequencing analysis, a special type of workload can be defined by using job arrays so data can be split and processed by many jobs in parallel.
- Subflow: In another innovative use case for genomics processing, multiple subflows can be defined as a parallel pipeline for variant-analysis following the alignment of the genome. The results from each subflow can then be merged into a single output and provide analysts with a comparative view of multiple tools or settings.
- Reusable module: The workflow can also be designed as a module and embedded into larger workflows as a dynamic building block. Not only will this approach enable efficient building and reuse of the pipelines, it will also encourage collaborative sharing of genomic pipelines among a group of users or within larger scientific communities.

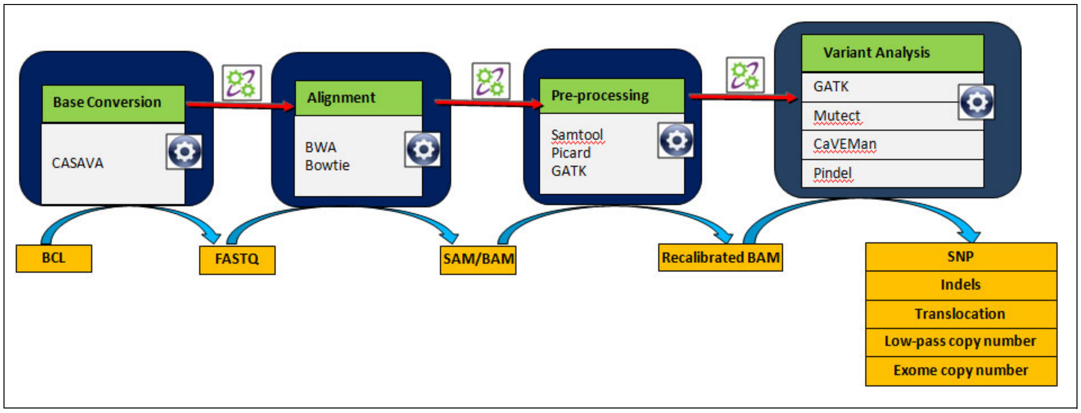

Figure 7 Genomic pipeline implemented using Orchestrator

Figure 7 shows the following components, starting from the left in the pipeline:

- Box 1: The arrival of data such as BCL files automatically triggers CASAVA as the first step of the workflow.
- Box 2: A dynamic subflow uses BWA for sequence alignment.
- Box 3: Samtool performs post-processing in a job array.
- Box 4: Different variant analysis subflows can be triggered in parallel.

Figure 7 shows a conceptual view of the IBM Genomic Pipeline in which an end-to-end workflow was created to process raw sequence data (BCL) into variants (VCF) using a combination of applications and tools. Each box represents a functional module of the workflow that consists of genomic applications that map to functions such as base conversion, sequence alignment, pre-processing, and variant calling/analysis. These modules can be implemented as stand-alone workflow itself and can be connected logically and conditionally into a larger flow such as the one displayed in Figure 7.

As more institutions are deploying hybrid cloud solutions with distributed resources, the *Orchestrator* can coordinate the distribution of workloads based on data localities, predefined policies, thresholds, and real-time input of resource availabilities. For example, a workflow can be designed for processing genomic raw data closer to sequencers, and followed by sequence alignment and assembly using the MapReduce framework on a remote big data cluster. In another use case, a workflow can be designed to start a proxy event of moving data from a satellite system to the central HPC cluster when the genomic processing reaches 50% completion rate. The computation and data movement can happen concurrently to save time and costs.

Another emerging use case for the *Orchestrator* is the publishing and sharing of genomic pipelines by one research institution with others. Because the software tools enable the saving and distribution of workflow templates, some leading cancer and medical research institutions in the US and Qatar have started to exchange genomic pipelines to facilitate collaboration.

Provenance Manager

Many computational methods and applications can be applied to assemble, analyze, and annotate the genomic sequences. The choice of applications, reference data, and runtime variables are critical provenance information that can have a significant impact on interpretation and preservation of genomic analysis. Currently, there is little to no public standards or practice to capture the provenance information, which potentially amounts to missing or losing critical data for computational analysis. This problem is potentially compounded by other factors such as complexity of data, workflow, or pipeline as a high-level analytical process and frequent release or updates of applications.

Therefore, provenance management is called out as an important function within *Orchestrator*, which is analogous to the importance of metadata management to *Datahub*. One can also think of provenance data as the *metadata of workloads*. The functional requirement for the Provenance Manager is to capture, store, and index user-defined provenance data in a transparent and nondisruptive way to any existing computational workloads or workflow.

Multiple technologies and solutions are being developed or have already been made available. Some of them are commercial solution such as ESP Platform from Lab7 or Nirvana from General Atomics. IBM has also been developing a technology for large-scale and near real-time metadata management system that can work in concert with *Datahub* and *Orchestrator*.

AppCenter for managing user interface

This section provides an overview of the *AppCenter*.

Overview of AppCenter

In the IBM Reference Architecture for genomics, the third enterprise capability called *AppCenter* is the user interface for accessing both *Datahub* and *Orchestrator*. The *AppCenter* provides an enterprise portal with role-based access and security control while enabling researchers, data scientists, and clinicians to easily access data, tool, applications, and workflows. Its goal is to make complex genomics platforms accessible to research and data scientists who do not have computer programming experience.

AppCenter can be added as an integral part of genomics, translational, and personalized medicine platforms, taking advantage of its reusability.

Figure 8 shows an overview of the *AppCenter*.

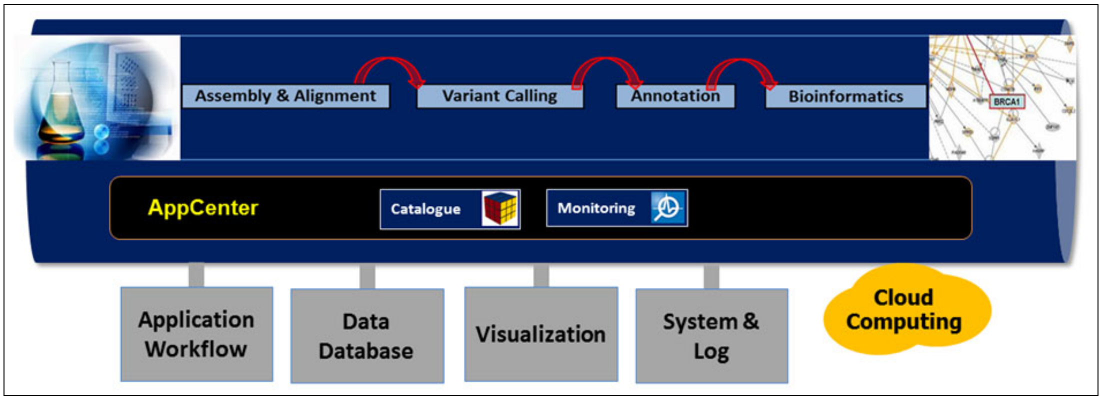

Figure 8 Overview of the AppCenter

The *AppCenter*, as shown in Figure 8, is the enterprise capability for accessing the genomics platform to start and monitor workloads, query and explore data, visualize analytical output, and track system log and utilization information. It defines the abstraction layer between users (researchers, clinicians, and analysts) and the *Datahub/Orchestrator*.

AppCenter functions

There are two functions in the *AppCenter*:

1. A portal-based catalog that provides access to applications, workflow, data set, and capability for visualization.
2. A monitoring capability that allows application-specific information to be monitored, tracked, reported, and managed.

Catalog

To minimize or eliminate the barriers between complex genomic analysis and data scientists who want intuitive access to genomic workflow and data set, there is a functional component called *AppCenter catalog*. This serves as a catalog of pre-built and pre-validated application template and workflow definitions so that users can easily start the job or pipelines directly from the portal.

Figure 9 shows an end-to-end genomic pipeline (BWA-GATK) that is started and visualized through the *AppCenter catalog*.

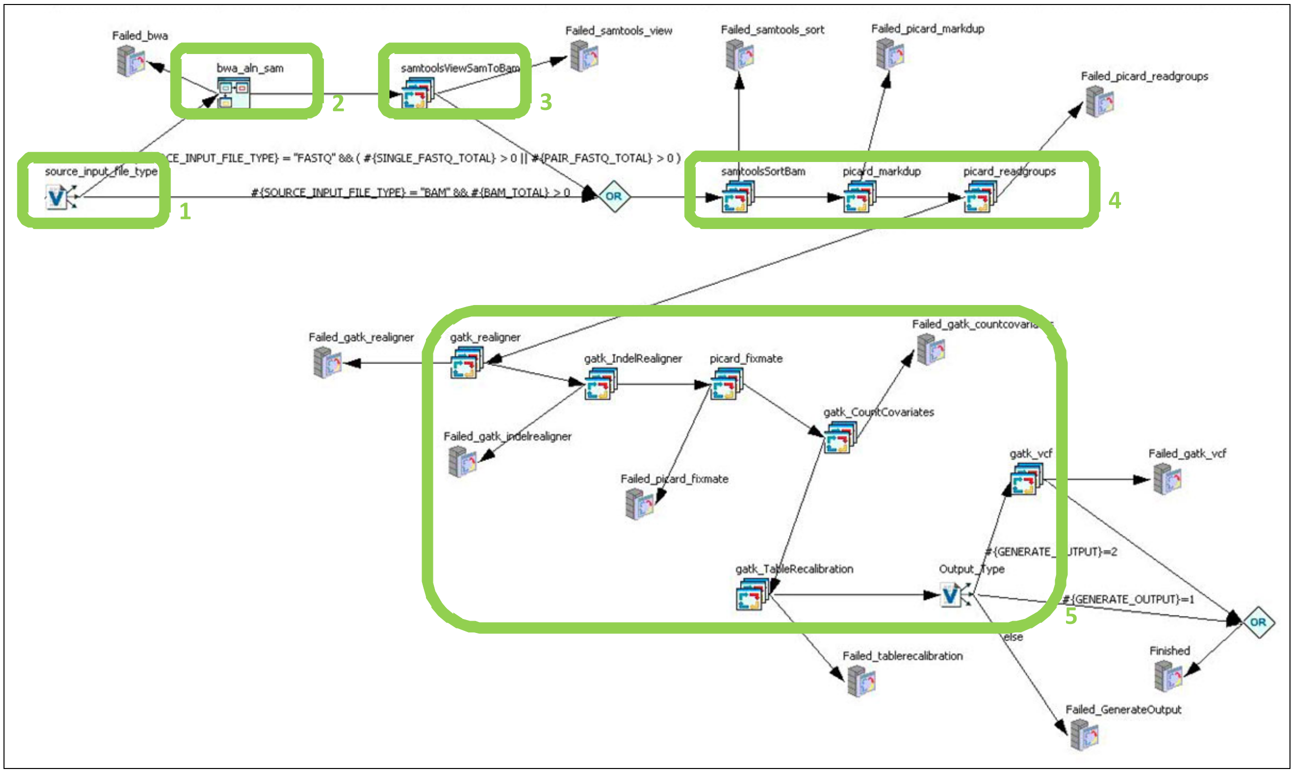

Figure 9 Genomic pipeline in the AppCenter

An end-to-end genomic pipeline can be started and visualized from the *AppCenter catalog* portal. Figure 9 shows these components, starting from the left in the flow:

- Box 1: Arrival of data automatically triggers the pipeline into action.
- Box 2: A dynamic subflow uses BWA for sequence alignment.
- Box 3: Samtool performs post-processing in a job array.
- Box 4: BAM file recalibration.
- Box 5 (large one in the middle): GATK performs variant calling.

The *AppCenter catalog* can be configured with a cloud data browser for user to manage the data needed for genomic computing. Within this portal-based browser, users can locate genomic data by browsing or searching all files/directories on remote or local storage servers (*Datahub*). Wherever the files are located, only two more clicks are needed to append the files for starting the job. With the data browser, users can mark/unmark a file directory as favorite, so it can be found easily and quickly. A useful directory to mark as favorite for a genomics computing user can be the one storing a commonly accessed reference data set.

Finally, the data browser can also facilitate data transfer because users can upload multiple files at the same time by dragging them from the browser desktop to the current remote directory.

Monitoring

The *AppCenter Monitoring* provides a portal-based dashboard that provides comprehensive workload monitoring, reporting, and management. Unlike other monitoring tools that focus on just one facet of system monitoring, this *AppCenter* function provides a complete, integrated monitoring facility for workloads. With the diverse profiles of genomic applications (for example, large memory, parallel, or single-threaded), the ability to track and report usage information at a job/application level can help improve application efficiency in terms of using computer CPU, memory, and storage I/O.

Additional information and resources

The following is a list of acronyms used throughout this publication:

Symphony®	IBM Platform Symphony
PPM	IBM Platform Process Manager
IBM Spectrum Scale	Formerly IBM GPFS
HPC	High Performance Computing
AMRC	Academic Medical Research Center
BAM	Binary Alignment Map (a type of genomic data format)
VCF	Variant Call File (a type of genomic data format)
BCL	A type of genomic data format
SSD	Solid-state drive (a type of storage disk)
IOPS	I/O operations per second (a measurement of storage performance)
IBM Spectrum Protect	Formerly IBM Tivoli Storage Manager
IBM Spectrum Archive	Formerly IBM Linear Tape File System
AFM	Active File Management (a feature built into Spectrum Scale)
SOAP	Short Oligonucleotide Analysis Package (a genomic software package by BGI)
BGI	Beijing Genomics Institute
MPI	Message Passing Interface (a communication protocol for HPC)
GPU	Graphical Processing Unit
SNP	Single Nucleotide Polymorphism (a type of genetic/variant mutation)

The following information resources for the IBM Reference Architecture for Genomics have been developed:

- Solution Brief for IBM Reference Architecture for Genomics

  http://ibm.co/1bufFla

- Review of the IBM Reference Architecture - IBM System Journal

  http://bit.ly/1ks3HvG

- Public website for the IBM Reference Architecture for Genomics

  http://www.powergene.net

Authors

This paper was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

Frank Lee, Ph.D. is the lead architect for IBM Genomics Solution and Software Defined Infrastructure. He has 15 years of experience in scientific computing and is a thought leader on High Performance Computing and massive-scale data management. As an advocate for shared research infrastructure, Frank created an end-to-end reference architecture and designed underlying high-performance computing, cloud, and analytics solutions for the health care, life science, and education industries. Currently, Frank is responsible for the implementation of this reference architecture in genomic medicine (PowerGene). To facilitate the transformation of the field facing huge IT challenges, Frank has spoken in dozens of public events, published papers, and written review articles to promote enterprise architecture for genomics computing. When there was technological gap, Frank invented a metadata-driven workflow and data management system (US patent applied). While working with these technologies, Frank also brings in the subject matter expertise on genomics, an experience that includes participation in the Human Genome Project as a research associate at Washington University Genome Center and currently being the technical advisor for Genome Arabia project. Trained as a molecular geneticist, he also conducted research with model organisms and discovered a novel cellular signaling pathway implicated in cancer gene regulation. In his blog site “Journey to Frontier”, Frank chronicles his journey for scientific discovery infused with technological innovation.

Thanks to the following people for their contributions to this project:

Daniel de Souza Casali
Marcelo Correia Lima
IBM Brazil

Joanna Wong
IBM Solution Center

David Cremese
Janis Landry-lane
Jane Yu
IBM Software Defined Infrastructure

Kathy Tzeng
IBM Systems Solution Enablement

Linda Cham
IBM Systems Development

Dino Quintero
International Technical Support Organization, Poughkeepsie Center

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:

  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new IBM Redbooks® publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm
- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-5210-00 was created or updated on May 20, 2015.

Send us your comments in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:
  ibm.com/redbooks
- Send your comments in an email to:
  redbooks@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

Trademarks