White Paper

8 Guiding Principles to Kickstart Your Healthcare Big Data Project

December 2018

CitiusTech

OVERVIEW OF BIG DATA IN HEALTHCARE

Big Data technologies have seen widespread adoption across different industries over the past 3-5 years, but the healthcare is just starting to realize the benefits. This is mainly due to the exponential growth of unstructured and semi-structured healthcare information.

With sensors and wearables becoming a part of our daily lives, people and organizations now have access to enormous amounts of data, e.g., step tracking, heartbeat / blood pressure monitoring, calorie tracking, sleep pattern analysis, etc.

The explosion in healthcare data, while posing massive storage and processing challenges, also has the potential to transform the way we use data to improve outcomes, for example:

- Predicting future care needs for specific populations
- Minimizing health risks by predicting specific events well in advance
- Identifying / expediting process of identifying new patterns in disease detection, etc.

Our experience with a large number of healthcare Big Data projects has shown that most customers face significant hurdles in kick-starting their Big Data initiatives.

With limited or no experience, customers often realize last-minute that their Big Data project implementations don't have the architectural robustness to address future needs.

This white paper illustrates our experiences and learnings across multiple Big Data implementation projects. It contains a broad set of guidelines and best practices around:

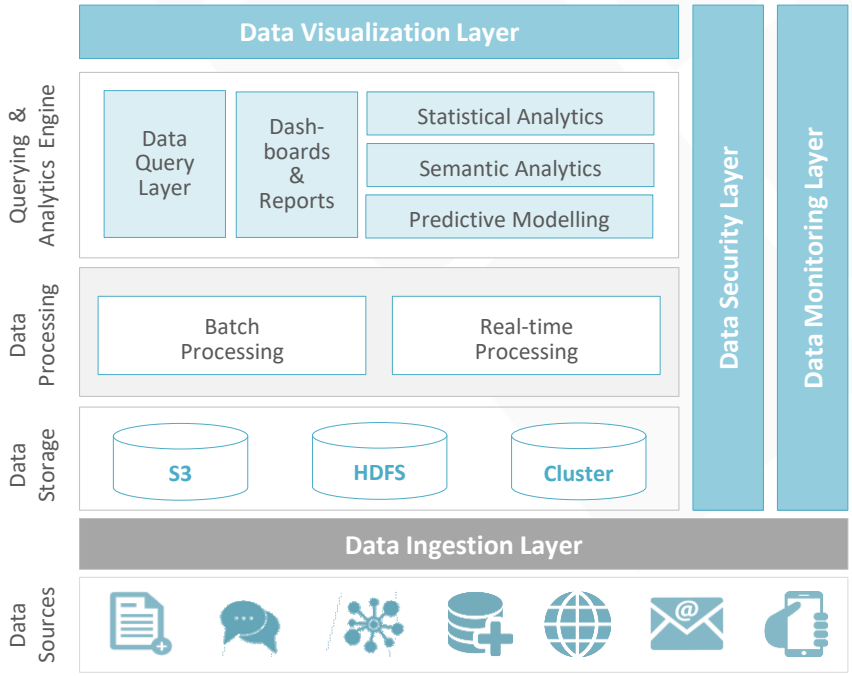
- Building highly secure Big Data lakes
- Efficiently processing vast amounts of data
- Providing access to downstream systems
- Best practices to mitigate project risks
- Technical hurdles and approaches to overcome them

GUIDING PRINCIPLES FOR BIG DATA IMPLEMENTATION

1. Use a Comprehensive Data Ingestion Framework

While working with a Big Data lake, you need to integrate numerous source systems with multiple feed types. Your Big Data solution should have the ability to handle different feed types, and cater to future source system integration needs. Design a data ingestion framework that addresses:

- All types of data: relational, semi-structured and unstructured
- Standard feed protocols: HTTPS, SFTP, etc.
- Different types of loading scenarios: including initial load and incremental load
- ELT (Extract, Load, Transform) approach as compared to traditional ETL
- Various ingestion frequencies: batch, real-time
- Relevant data ingestion mechanisms (push or pull). Pulling data may not be preferable when data lake is on the cloud and sources reside on-premise



Big Data Layers: Typical Architecture

2. Choose the Right Storage Type for Each Feed

Since Big Data ecosystems provide multiple storage components, it gives the opportunity to use relevant and optimal storage type for a specific feed. The following points need to be considered while choosing a storage type:

- **Feed attributes:** e.g., total size of data, size of individual files, velocity at which data arrives, etc.
- **Data ingestion system:** Ability to identify whether the data ingested is small or big in size
- **Database architecture:** Based on size, data can be stored in distributed file systems, cloud storage or in NoSQL / columnar data bases. For example:
    - Files of 128MB and above (default Hadoop block size), can be stored in HDFS. Small files (in KBs) can be stored in Hadoop sequence files, or in HBase
    - JSON data can be stored in document database

3. Create Separate Storage Layers

Organizations starting their Big Data implementations often ask, "How do we arrange data in a Data Lake?" and "How many layers should we create?". The answers depend on the type of data being pulled and processed in the Data Lake. In a standard scenario, customers want to correlate data from relational systems, IoT devices, social media and unstructured data sources, e.g. notes, images, documents etc. In such scenarios, a three layer approach can be used.

Raw Layer

Although not mandatory , it is always advisable to store data in its native form in the Data Lake. This forms the raw layer or raw zone of the Data Lake. The raw layer is generally referred by data scientists or analysts to perform analysis instead of waiting for operational data.

Curated Layer

While the raw layer is important from a raw analytics and reprocessing perspective, it isn't the most

optimal way to store data, as it may contain duplicate, incorrect or incomplete records. It is always advisable to create a curated data layer that has cleansed and standardized data. Analytics performed on the curated layer provides much more accurate results than the raw layer.

Operational Layer

Data stored in the curated layer isn't reconciled and continues to have the context of the source system. This poses analytics challenges and also has the possibility of duplicate records being sourced. The operational layer solves this problem by reconciling and transforming incoming data from different sources into a single, canonical model.

4. Use the Right Data Processing Frameworks & Tools

Identifying the right data processing framework can be difficult as there are multiple processing frameworks in the Big Data ecosystem. Common data processing tasks like data cleansing, quality reporting, aggregation, transformation and reconciliation can be performed by standard ETL tools. However, for Big Data processing, most standard ETL tools use Apache Spark. While these ETL tools provide drag-drop UI and out-of-the-box adapters, the internal working is abstracted, making them difficult to operate in certain scenarios.

Commonly used ETL tools are Talend Enterprise, Pentaho, Informatica, DataStage and Attunity. For simple data processing needs, IT teams can create a custom ETL utility using Apache Spark and its in-built transformation functions.

The following best practices need to be kept in mind while working on data processing frameworks:

- Big Data processing happens in a distributed manner. It is necessary to arrange data to minimize shuffling and optimizing performance. Use compression to speed up data transfer over network and reduce shuffling time

- Joins are expensive in Big Data and should be thoughtfully implemented.You can also improve performance by de-normalizing records
- Use parameters like batch id, date range or specific set to overcome bad / corrupt data issues
- Keep track of events (meta-data, audits) during data processing, e.g., who triggered the process, which dataset was used for processing, size of the dataset, count of records processed, status of processing, start and finish time, etc.
- Be practical with partitioning. Distributed processing often fails to take full advantage of the nodes due to small or numerous partitions
- For stream processing, create enough partitions on a Kafka Topic to trigger parallel processing in Apache Spark. Provide checkpoints at regular intervals to minimize stream processing failure

5. Think of Data Management Right at the Beginning

With business environments changing rapidly, organizations need to consider data management as a critical component of their business strategy. The organization's data strategy is affected by multiple scenarios, including:

- Changes in organization or technology
- Process and people changes due to mergers and acquisitions
- Changes in regulatory compliance or contractual arrangements
- Issues with quality / availability / timelines of data that affect decision making
- Massive investments in time and resources required to get data in correct shape

To overcome these challenges, organizations must start thinking of data management solutions right from project inception.

Few frameworks provide data management capabilities for Big Data, e.g., Apache Atlas with Apache Falcon for Hortonworks, Cloudera Navigator has partial functionality, MapR uses a custom framework.

7 Pillars of Data Management

1. **Data Architecture:** Data analysis, enterprise data architecture, integration with applications
2. **Content Management:** Organizing, consolidating and optimizing content
3. **Data Development:** Requirement analysis, data modelling, database design, implementation and maintenance
4. **Master Data and Metadata Management:** Master patient index, master provider index, master facility index, ICD 9/10, CPT, SNOMED, LOINC, DRG and standards, common codes, integration metadata, control metadata, quality metadata
5. **Data Quality:** Measurement, assessment and improvement in data quality
6. **Operations Management:** Acquisition, recovery, tuning, retention and purging
7. **Data security:** Classification, administration, privacy and confidentiality, authentication and auditing

6. Provide a Sophisticated Search Capability

The search feature becomes essential to Big Data systems due to high volumes of data. Searching for specific attribute values is like finding a needle in a haystack. As entities are added / updated / removed from the Data Lake, there must be a way to quickly search and get a view of the entities present and quickly search for specific attribute values.

Its always beneficial to index your data and provide a search UI for quick discovery. Consider providing a facility to tag attributes to make it searchable and allow users to group attributes using tags.

7. Simplify Data Access Using APIs and Data Virtualization

All data warehousing / Data Lake projects need to provide data extracts to downstream / external systems, and allow users to search data and enable analytics systems to connect and analyze data using

standard interfaces. Most of these requirements can be fulfilled by a thin API access layer that provides unified access to the underlying data. The API layer implementation should support standard based interfaces like REST, SQL or a combination of both.

Data extraction processes are scheduled jobs that extract data from specific tables and store it in a shared location (e.g., SFTP). A low priority processing queue can be used for data extraction during peak hour to ensure the extraction query does not consume all processing resources. Additionally, data virtualization software (e.g., Denodo) or custom data virtualization layer (using Apache Ignite and Spark) can be used to create a common interface for Data Lakes and other source systems.

8. Provide an Analytics Workspace for Advanced Users

With the evolution of Big Data and Data Lakes, more organizations are adopting advanced analytics tools and technologies – e.g., Predictive Analytics, Machine Learning, Deep Learning, Natural Language Processing and AI algorithms. These technologies require extensive piloting, model operationalization and custom dashboarding before they can be applied in real-world scenarios.

Data scientists and analysts need a dedicated workspace and desired toolsets to pull, process, analyze raw, curated and aggregated data, and share their findings. They should be able to perform activities like preliminary analysis, identifying new trends and quick dashboarding, without affecting the Data Lake.

An analytics workspace can be implemented in one of the following ways:

A. Use Existing Data Lake Infrastructure to Carve Out Space for Individual Data Scientists

This option uses the existing Data Lake infrastructure to create slots for individual data scientists where they can to play with a copy of the data using various tools e.g. Apache Spark based note books.

B. Use a Separate Cluster for Each Data Scientist

This option creates separate infrastructure for individual users and pulls data from the Data Lake. This option may prove costlier but provides a true multitenant architecture and ensures that the system performance is always optimal.

H-SCALE ADDRESSES KEY HEALTHCARE BIG DATA NEEDS

CitiusTech's H-Scale platform for healthcare data management has been specifically designed to address healthcare Big Data challenges such as data acquisition, real-time processing, Master Data Management, data security and advanced analytics. Here is how H-Scale supports the Big Data requirements discussed in this paper.

Data Ingestion	Highly configurable data ingestion pipeline that caters to structured, unstructured and semi-structured data ingestion, using Big Data ecosystem components like Sqoop, Flume, etc. Also provides real-time data ingestion-streaming using Apache Kafka and Storm based scalable ingestion cum processing pipeline.
Storage Types	Configurable data ingestion pipeline - dynamically chooses storage (HDFS or HBase) based on data attributes.
Storage Layers	Ability to configure and execute data transformation and reconciliation rules using a self-service UI. CitiusTech's healthcare data model can be used to create canonical data model in operational layer.
Data Processing	Highly configurable and easy-to-use data processing pipeline built on top of Apache Spark to perform data validation, curation, transformation and reconciliation. Data processing pipeline improves time-to-market for customers by quickly integrating data from various sources.

H-SCALE ADDRESSES KEY HEALTHCARE BIG DATA NEEDS

Data Management	Data governance adapters to capture data lineage and auditing information. H-Scale data governance adapters can be used while working with Apache Atlas on Hortonworks Data Platform (HDP) and Cloudera Navigator when working with Cloudera Hadoop Distribution (CDH).
Search	Apache Solr indexing framework to index specific tables for fast search. It also provides tag-based logical grouping facility for searching all occurrences of specific groups.
Data Access	Apache Spark and Ignite based data virtualization platform which can connect to different sources without replicating data. Data virtualization processes use source catalogue to join data at runtime without replication.
Analytics Workspace	Big Data analytics workspace that provides self-service UI, Zeppelin based notebook and tools for creating data processing pipeline.

CONCLUSION

As healthcare organizations worldwide begin to roll out their Big Data strategies, they will face a number of challenges along the way. With the right initial approach, organizations can create more robust strategies which enable them to leverage their Big Data assets more effectively.

Our experience with Big Data implementations puts us in a strong position to define and articulate best practices for healthcare Big Data implementation. CitiusTech's H-Scale platform for healthcare data management has been aligned to fit seamlessly with the healthcare industry's Big Data implementation needs.

REFERENCES

- https://atlas.apache.org/
- https://www.redoxengine.com/blog/how-to-do-microservice-chassis-and-microservice-scaffolding-on-a-budget-2/

ABOUT THE AUTHORS

Pawan Mathur
Senior Technical Specialist – Data Management Proficiency, CitiusTech
Pawan.mathur@citiustech.com

Pawan has 20+ years of experience in the IT industry. He has extensive experience in software development using Big Data Flink-Spark-Hadoop and Analytics. He has played the role of Senior Architect in the development and implementation of CitiusTech's H-Scale platform. He holds a degree in Software Enterprise Management from the Indian Institute of Management, Bangalore.

Swanand Prabhutendolkar
Vice President – Data Science Proficiency, CitiusTech
Swanand.Prabhutendolkar@citiustech.com

Swanand leads the Data Management Proficiency at CitiusTech which includes the Healthcare Interoperability, BI-DW and Big Data practices. He has 20+ years of experience in the IT industry., of which 11+ years are in healthcare analytics and data management. Prior to CitiusTech Swanand served leading technology organizations such as EPIC Corporation, Polaris and 3i Infotech. He holds a Master of Science degree in Information Technology and Applied Statistics from the Indian Institute of Technology (IIT), Bombay.

CitiusTech

CitiusTech is a specialist provider of healthcare technology services and solutions to healthcare technology companies, providers, payers and life sciences organizations. With over 3,200 professionals worldwide, CitiusTech enables healthcare organizations to drive clinical value chain excellence - across integration & interoperability, data management (EDW, Big Data), performance management (BI / analytics), predictive analytics & data science and digital engagement (mobile, IoT).

CitiusTech helps customers accelerate innovation in healthcare through specialized solutions, healthcare technology platforms, proficiencies and accelerators. With cutting-edge technology expertise, world-class service quality and a global resource base, CitiusTech consistently delivers best-in-class solutions and an unmatched cost advantage to healthcare organizations worldwide.

For queries contact thoughtleaders@citiustech.com