Big Data Analytics with Microsoft

Scalable Intelligence Using Azure Synapse, Fabric, and Power BI

Swarup Panda

DeepScience

Big Data Analytics with Microsoft: Scalable Intelligence Using Azure Synapse, Fabric, and Power BI

Swarup Panda

SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

DeepScience

Published, marketed, and distributed by:

Deep Science Publishing, 2025
USA | UK | India | Turkey
Reg. No. MH-33-0523625
www.deepscienceresearch.com
editor@deepscienceresearch.com
WhatsApp: +91 7977171947

ISBN: 978-93-7185-524-2

E-ISBN: 978-93-7185-151-0

https://doi.org/10.70593/978-93-7185-151-0

Citation: Panda, S. (2025). *Big Data Analytics with Microsoft: Scalable Intelligence Using Azure Synapse, Fabric, and Power BI*. Deep Science Publishing. https://doi.org/10.70593/978-93-7185-151-0

Preface

This book is a complete guide for professionals and data enthusiasts who want to make the most of Microsoft's cloud-native ecosystem for big data analytics. It covers essential services like Azure Synapse Analytics, Microsoft Fabric, and Power BI. The book provides a full framework for scalable data processing and smart decision-making. Readers will learn best practices for data ingestion, transformation, storage, modeling, and visualization. They will also see how to combine data engineering, data science, and business intelligence workflows within a single Microsoft environment. With practical examples and architectural designs, this book helps readers build secure, effective, and cost-efficient analytics solutions that meet the needs of today's enterprises.

Swarup Panda

Table of Contents

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 1: Big Data Technologies and Their Integration with Microsoft Tools

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Understanding Big Data

Big Data is a term used to describe the large volume of both structured and unstructured data. Big Data is unpredictable. The speed of accumulation is high and coming in from a range of sources including social media and sensors. The characteristics of Big Data are usually referred to as the 3Vs of Big Data [1-3]. They are volume, variety, and velocity. The characteristics of Volume and Velocity are what really set Big Data apart from traditional Data Analytics. To this book, we consider Data Lake and Data Warehouse Analytics Traditional Data Analytics.

For more than 20 years now, the emphasis of Data Analytics technologies has been on operational transaction processing. With the analytically driven interests of organizations, the rapid growth and availability of Social Network data, unstructured and semi-structured data volumes are expanding significantly before Big Data tools were made available [2,4]. What accelerates this Data burst has been the introduction and adoption of group knowledge-sharing data and a variety of new sensors as evidenced by the introduction of the Internet of Things. Then these Data Sets, largely uncovered or previously unsampled, are now being stored to make better new or improved decisions [5-8]. Big Data goes beyond the traditional Data Warehouse Data due to its large Volume; higher Velocity, Variety, and Variability; and increasing Value of Data. Big Data is Dirty Data as the sources may be unverified. Digital Data is changing in real-time and presents newer opportunities for analysis and decision-making, currently through Machine Learning and Artificial Intelligence actuarial algorithms. Business and

healthcare analytics are being transformed by the application of new data sources and the associated algorithmic processing leveraging Deep Learning algorithms. The difference is that whereas traditional Data is supported by the Data Warehouse, Big Data is unstructured and disparate; it is supported by the rapidly evolving Data Lakes and other less structured Big Data technology [6,9].

1.1. Definition of Big Data

To understand the concept of Big Data a good starting point is to define data and the several types of data we deal with in our daily life. Our communications on a day-to-day basis almost exclusively rely on the existence of data. What is data after all? It is the collection of objects, events, transactions, or happenings of a defined subject that takes place in a definite period that is desirable to analyse. We deal with data in various forms such as qualitative, quantitative, subjective, objective, numerical, or categorical in their raw format or summarized.

You may have also heard of the term Information. What is the relationship between Information and Data? Data at one moment in time may not be termed Information. Information is Data that has been involved or processed in a particular way such that it has an added goal or purpose. Data has a quality to it that is valuable for business or research activities [10-12]. Data, whether it is in its raw format or summarized, is the foundation or building blocks of Information. Today, the data, information, and knowledge pyramid, which delivers the hierarchy of data in the domain of business intelligence, data science, and analytics, has assumed a new definition. From a cost perspective to the expense of collecting and storing data, it was tremendously reduced due to cloud storage technologies. The various facets of Big Data which we will be discussing, velocity, variety, volume, variability, visualization, and veracity have made Data the foundation for development on which countries should build their character to become knowledge-based economies, where Intelligent Nations empower their citizens to effectively use the Data to create, innovate and generate business value [7,13-16].

Now, over the past few years, the various academic and technology research-based institutes and companies have been attempting to define Big Data. In our discussions, we will focus on Big Data, Data Science, and Analytics, each of the three representing a different level of the Information architectural structure. The focus of this chapter is to define the term Big Data, and its facets, as in today's world of business intelligence, big data has situated itself in the digital matrix which has revolutionized the way countries conduct business and how organizations are able to generate, create, visualize, and develop business analytics reports to make decisions.

1.2. Characteristics of Big Data

Big Data technology has evolved to a stage where most organizations visibly acknowledge its value yet find it difficult to estimate its deployment on their portfolio. The term Big Data defines a large volume of data that comes with complex structures at a high velocity, but devoid of widespread and comprehensive explanation [2,17-19]. This section focuses on defining Big Data in terms of its characteristics, examples of storage solutions and use cases, and its importance and relevance to organizations.

Big Data is characterized with the 5Vs - Volume, Variety, Velocity, Veracity and Value - that when expressed in various degrees defines the property of the data. The characteristics are a progressive lug-in from the small size equivalent data to simpler systems leading to guiding dimensions that need to be evaluated while focusing on Big Data systems. The various characteristics of Big Data are discussed below.

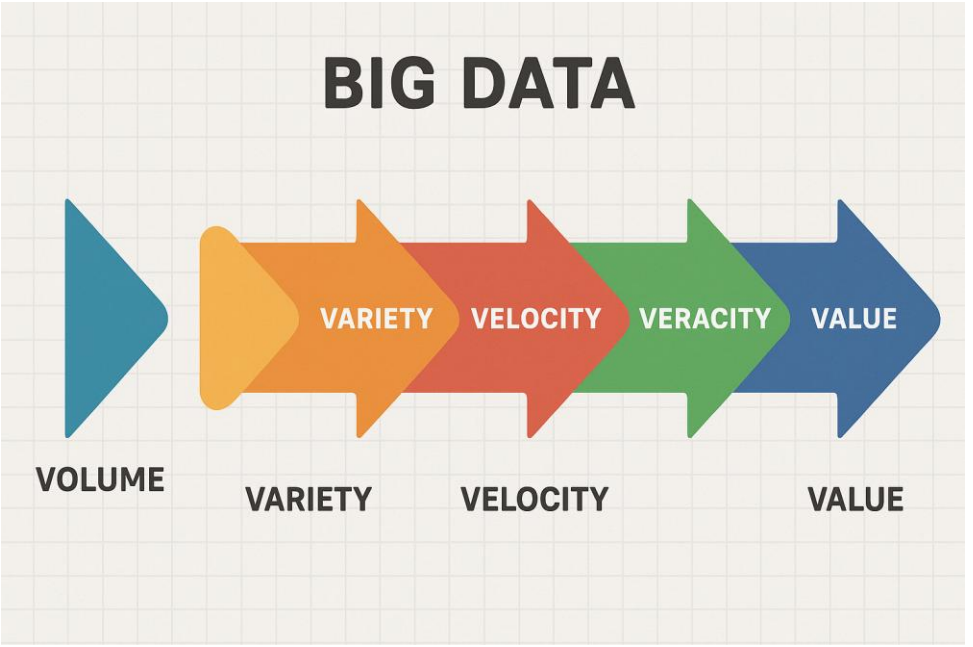


Fig 1. Bigdata

Volume refers to the size of data generated on modern computing systems. The growth in storage space enabled by advancements in technology and the advent of cheap storage have enabled businesses to store terabytes and even petabytes of data. A commonly cited data point is that more than 90 percent of all data today has been created in the last two years, with modern systems generating and using more than 2 zettabytes of data in 2010. Store data doubles every four years,

a rate faster than computer processing improvements [3,20-23]. Data such as clickstream, multimedia (images, audio, video), GIS (satellite imagery), medical records and genomic equivalents, and email become potential candidates in this category. Enterprise data warehouses and RDBMS-based systems have relatively well-defined limits to data sizes before performance issues are encountered. Unfortunately, much of this expectation is based on traditional architectural designs and possibly older technology.

1.3. Importance of Big Data

Organizations have been collecting and processing data for ages. Initially, data was a means of keeping track of goods and services. The ancient Egyptians recorded harvest yields and inventory items to aid trade and commerce [9,24-26]. As the modern economy developed, records became more complex, covering sales, supply routes, transportation, and consumers. The Industrial Age brought the use of modern technology for data collection. Every sale was recorded as a product was scanned by a clerk at checkout using a barcode reader. Inventory control systems kept a live update of stock available at the back of stores. Consumers swiped their credit cards at the point of sale, creating a digital trail. Yet, all these data points were processed in a sequential manner by a central server.

The dawn of the Internet Age saw the meteoric rise of the volume of digital data generated every day. Websites created exponential digital footprints for consumers. Digital advertising provided clicks and conversions, contributing to the volume. Social media dynamics saw new users join simultaneously [27-29]. Massive amounts of UGC were created in the form of photos, videos, comments, likes, and shares. Financial transactions underwent an epoch of rapid modernization as digital payment systems emerged in the Twenty-first Century. These systems enable those selling goods and services to process payments in seconds and wiring money to family from anywhere became virtually free. These economic activities, coupled with the rich and varied consumer interactions on social networks, create and continue to create more data than the world had ever seen.

Fig 2. Importance of Big data

2. The Microsoft Ecosystem Overview

The growing mass of data in organizations over the past several years is leading to the so-called big data technology revolution. Among the challenges presented by big data is ensuring that organizations can leverage the data assets to increase revenue, to lower risks of doing business, and to improve and streamline business operations. Data intelligence has gained prominence in technology circles as the next-generation solution to use analytics and reporting to transform data into useful information for the organization [30-32]. Recognizing the importance of the popularization of data intelligence as a lead technology, software companies are developing and deploying technologies to help organizations wrangle, manage, analyze, and visualize corporate and external data of all varieties.

Microsoft created a core data technology ecosystem to appeal to the needs of organizations for solutions to derive insight from data. The Microsoft ecosystem consists of software tools and services that assist organizations throughout the various steps of the intelligence cycle [9,33-35]. Each tool has a strong feature set that appeals to different audiences, allowing organizations to build a strong ensemble of tools that can scale along with their increasing business intelligence and analytic needs.

Over the past several decades, Microsoft has become one of the world's most important technological players. Their strategy has been to create a large ecosystem of software solutions that span multiple areas of technology and address the needs of a wide variety of customers – from the single-device consumer to the enterprise deployments found in corporate, government, and academic institutions.

2.1. Introduction to Microsoft Technologies

The Microsoft Ecosystem offers an environment where many components function and collaborate seamlessly. This involves thousands of developers around the globe investing in their ineffable talent and zeal to make technologies that seamlessly integrate and present their creations to the world using a Microsoft platform. The array of Microsoft technologies and products has sort of a charm about them. They are so integrated and lend themselves to so many easy-to-use methods for carrying out hundreds of complex tasks in different domains making the enterprise development world appealing both to the developer and the end user that one cannot but stop and wonder how this is possible [36-38].

What is more interesting is the enterprise world is obsessed with security. Most businesses are shying away from exploring and deploying systems from smaller, lesser-known technology vendors with the fear that they will leave their customers high and dry in case support or services are needed later. Likewise, enterprise shoppers are going for solutions that are stable, proven, adopted and tested in the field. Microsoft delivers all this. In fact, this was their approach to build their market for Windows and other enterprise products. They offered what is often called an "end-to-end" solution [3,39-41]. By that, they meant integrating the operating system, the database, the mail server, and so on into one seamless environment which worked efficiently and securely and easy for the developers to create solutions for vendors and enterprises as well.

This approach to enterprise development helped Microsoft capture more than 90% of the enterprise applications development and deployment market. In fact, their share of the market is still near that number even in today’s world of big

data and cloud computing, where open-source technologies are sort of taking over.

2.2. Key Components of the Microsoft Ecosystem

On a high level, the ecosystem that you'll be working in is not just "Microsoft." There are other partners that have built great solutions that complement Microsoft's products, and there are a lot of technologies from Microsoft that are important as part of the ecosystem. This chapter goes through the key components of that ecosystem.

2.2.1. Azure SQL Data Warehouse

Azure SQL Data Warehouse is a cloud-based enterprise-scale analytics service. It is a distributed SQL query engine and Microsoft’s Cloud Data Warehouse solution for analytics. It combines the reliability and convenience of Microsoft Azure with the advanced capabilities of SQL Server and Microsoft Business Intelligence. You typically use Azure SQL Data Warehouse to support business intelligence (BI) reporting or data analytical applications for forecast, budgeting, month-end data consolidation, and ad-hoc query on historical business data. Azure SQL Data Warehouse integrates with various ETL software that supports bulk data movements from a variety of data sources.

2.2.2. Microsoft Power BI

Microsoft Power BI is a suite of business analytics tools for the cloud to analyze data and share insights. Power BI can be connected to SQL Data Warehouse and various data sources both on-premise and cloud based. It helps create visualizations of those data with interactive dashboards and beautiful reports that are easy to create and easy to share. To summarize, it collects the data, performs data transformation, and stores it in a Data Warehouse, and then uses Power BI to analyze the data and provide insights into the business. It is also a good Business Intelligence tool to perform analysis on data from multiple sources to provide ad-hoc analysis.

3. Big Data Solutions in the Microsoft Ecosystem

This chapter showed the motivation for Big Data and how from its origins until the present day has evolved into today’s Big Data technologies and solutions. These solutions mobilize a significant part of the world’s economy, thus the knowledge of very diverse professionals who need to understand today’s data

and the available technologies to store, process, and analyze it, become increasingly important. This chapter also gave an overview of the Hadoop ecosystem and its main components, which are also part of the Microsoft ecosystem, where Big Data technologies have evolved to today's unique Microsoft ecosystem. From the previous sections, Microsoft Azure plays a central role in storing, processing and analyzing Big Data. In addition to the Azure ecosystem, business professionals need to understand how to create dashboards and reports to browse the data and use tools to create charts [36,42-44]. Central service publishes dashboards and reports and facilitate the distribution and team collaboration. Data professionals use Desktop to create dashboards and reports. can connect to the various data sources from the Microsoft ecosystem to ingest static or streaming data to publish dashboards and reports on the Central service, as well as connect to many other data sources available in the market. has been gaining popularity not only with business users, but also with data specialists, mainly due to its easy-to-use dashboards and reports and connectors with many other external data sources that team users, data engineers, data scientists, and data analysts from various areas have used to make decisions based on external data. Finally, today's data stored in hinged tables or tables and views in relational databases accessed through famous are just one of the various forms of data today's organizations manage and use for decision making.

3.1. Microsoft Azure for Big Data

Microsoft Azure is the cloud platform from Microsoft, where you will find a set of integrated services for Big Data. Azure provides infrastructure services that allow the integration of different platforms and technologies for the collection, storage, processing, analysis, and visualization of Big Data solutions. These services can be deployed either in the cloud, on-premises, or in hybrid topologies. Microsoft Azure implements open-source technologies, as well as APIs and connectors that ease their integration with other cloud providers.

Big Data Solutions typically use numerous resources for their processing tasks, and therefore, the performance in a Big Data solution relies on an optimized use of storage and processing services [40,45-46]. Consequently, the Big Data architecture needs to address challenges inherent to Big Data: the storage architecture needs to be designed for each specific workload of the solution, while the processing operations deployment also needs to be performed in an efficient manner. Azure provides what is called "Provisioned capacity", which allows the customer to specify and contract specific resources from Azure, and

"Serverless capacity", which uses Azure's technology to scale for the customer only what they need, when they need it.

Different solutions for the storage and processing of Big Data can be found in Azure: Data Lakes, Data Warehouses, Batch Processing services, Stream Processing services, Machine Learning services, and Artificial Intelligence services. As for the Data Lake Storage, Azure Storage provides a specific set of APIs that allow you to interact with the data in the storage as a Hadoop-compatible file system. In addition, Azure provides high-performance and low-cost Blob operations to perform operations on data files stored in the Data Lake. These operations can be done either using the proprietary protocol or by using the open-source connector and libraries.

3.2. Power BI for Data Visualization

Power BI is a business intelligence and data visualization tool that Microsoft released as a software-as-a-service offering in 2015. The development history of Power BI, however, extends back to Microsoft's earlier products like Excel and SQL Server Reporting Services. Power BI allows users to unify data from many sources to create interactive dashboards and reports. Although Excel is still an excellent choice for creating charts and small reports, Power BI is a more suitable choice for business intelligence suites that require reports for big data sources and dashboards that provide a view of vital business metrics.

Power BI is an attractive data visualization solution because it is easy to learn and, most importantly, has a free-tier version which allows you to develop Power BI reports on your Windows PC which can be ingested by Power BI Pro at your organization. Compared to other business intelligence tools, Power BI provides a good range of visualization tools to cater to most business needs. Power BI also supports both on-premises and cloud hosting of your reports.

Organizations that are in the Microsoft ecosystem and use Azure as their cloud provider commonly leverage the integration capabilities that Power BI provides with other Microsoft products. Together with dedicated connectors to other vendors, organizations typically find data sources of value to their business in their organizations and in the outside world. Data scientists and advanced analytics teams use Power BI to create custom visualizations to model the data and tell stories about the overall process that occurred.

3.3. SQL Server and Big Data

Relational database management systems (RDBMS) played a predominant role in the data ecosystem during the last decade by providing the transaction

processing and analytics backbone for enterprise applications and services. However, the challenges posed by Big Data in terms of high-volume and high-velocity have encouraged organizations to explore a new generation of data analytics tools that are amenable to distributed and parallel processing on commoditized servers. This has caused a decline in the popularity and usage of SQL Server over the last five years. Nevertheless, Microsoft continues to polish SQL Server. With every new major release, SQL Server gains new features and functions that allow organizations to effectively tackle different types of workloads, including Big Data. With SQL Server 2019, Microsoft is clearly positioning SQL Server as an ideal data hub for organizations.

SQL Server has a long history of strong support for structured and semi-structured data. However, SQL Server had limited or no support for a majority of unstructured data such as images, audio, and video, and social media data in its initial versions. SQL Server 2019 enriches SQL Server with several new features that enable organizations to ingest, store, process, and analyze multiple data types from a wide variety of sources. Empowered with these features, organizations can now use SQL Server to manage diverse workloads, from transactional processing to analytics on different types of data and perform enterprise-wide analytics using a wide variety of services on a one-stop platform.

4. Data Storage Options

Data storage is an important part of any IT solution. Data storage options can be broken down into three areas: storage in blob format without any inherent structure, data structures that are schematized and enforce schema on write, enabling structure for easy querying, and schematized options that permit more flexibility in schema storage and management. Depending on the type of data, the retention period for data, the frequency of use of data, and the costs of storage, you may use one or more of the following areas to store your data.

Data Lake Storage is a service that is built specifically for working with big data. It stores data in a file format like blob storage, though there are accompanying format technologies that help present the files as tables. Data Lake Storage is intended for long-term storage of raw (or near-raw) data [2,3]. Due to the storage costs associated with it, you wouldn't be putting in data that you need to access often, like the data used for operational processes. Instead, data stored in the Data Lake would be for analytics or machine learning needs and probably be accessed through various services.

If your data has a structured schema or requires strict schema on write, you could use a database solution. One option started out as a cloud-based complement to traditional databases but now supports large sets of semi-structured datasets. You could use it for documents, graphs, and key-value data types. There are no restrictions on the amount or size of the documents that you can store, as its table format lets you use different document expansion rules, scaling its storage automatically to fit your needs. It is globally distributed, has transactional support, and allows you to scale your configurations independently for throughput and storage size, offering flexibility in both costs and usage. Its connections with various functions also make it useful for information and operational functions, supporting event-driven operations.

4.1. Azure Data Lake Storage

Many times, scientific analysis requires data that is not structured in a relational format. These include documents, video, image, and audio files, as well as log files, clickstream data, message contents, etc. Also, although SQL Server can also store some non-relational data, it was originally designed to be a database management system with the main purpose of collecting, organizing, and retrieving data efficiently. This relational model, designed for complex transactional operations, was combined with indexes, triggers, data integrity, and many other requirements needed to support the consistency and reliability needed for operational systems.

When looking for storage options for fast data, and especially for data science, we should be looking for systems optimized for cost and performance when asked to collect massive amounts of diverse data, required for our analysis but that were never used in transactional operations, since they belonged to a company's operational storage systems, and would usually be kept under some data retention policy. That is the specific purpose of Azure Data Lake Storage. ADLS is a Big Data storage service capable of housing virtually unlimited amounts of data and virtually unlimited diversity of formats. It was designed from the ground up with a focus on the specific needs of data scientists and analysts by providing high throughput, low latency, security, and integrated monitoring while also simplified access and usage and working seamlessly with tools. ADLS functionality is also offered by systems within the Hadoop ecosystem, such as HDFS and services.

4.2. Cosmos DB

Cosmos DB is a key-value, document, and graph database as a service, or DBaaS, which is a fully managed service on the Azure platform. With its multi-model

approach, it provides support for both non-relational SQL, like relational, columnar, document, and key-value model, and non-SQL data access APIs. With its automatic NoSQL data computing with low latency and high availability service, you can provide your business data anytime and anywhere around the world. Because it allows regular data structure-less schema for application development, it has a rapid custom development cycle and lowers application deployment time and cost. It is particularly good at enterprise business systems and enterprise e-commerce or retail websites.

Apart from providing a full-fledged NoSQL solution on the Azure platform, there is also an SDK for Cosmos DB so you can deploy it on your own environment, further extending the reach of Cosmos DB as an enterprise application infrastructure as a service solution.

4.3. SQL Database Solutions

Most computer users are familiar with the concept of tables, rows and columns of data. In addition, we tend to think of this type of structure as ideal to represent sets or collections of entities of similar characteristics, or of relationships among those entities. Consequently, SQL database solutions have been prevalent for decades, as data experts build dynamic applications in varied computing environments, to address an immense number of common business needs. As such, SQL database solutions are also a key component of the ecosystem.

We can find SQL databases in on-premises data centers, data centers, or even at third party cloud service providers. Depending on the various needs of applications and those working with data, including volume, variety, velocity, and veracity of the input digital information, and of derived datasets, the various SQL database solutions usually work together to address the diverse workload requirements of organizations. Databases might also be deployed in clusters or mirrored for performance, scalability, and resilience as data-driven applications and operations scale up – an ever-growing technical need at the level of enterprises.

In addition to SQL Server, deployed on-premises, or in Virtual Machines in the cloud, there are three SQL database services available: Database for MariaDB, Database for MySQL, and Database for PostgreSQL. These popular open-source database engines are available in Managed IT service mode in the cloud, taking care of the backend engine-related duties as a full-service service level agreement, or again at the level of the enterprise, or in develop-test mode as a dev/test service level agreement, depending on varied factors like workloads and data.

5. Data Processing Frameworks

Data processing frameworks are services that help us process big data efficiently. There are a number of popular open-source data processing frameworks, including Hadoop Map-Reduce, Apache Spark, Apache Storm, Apache Flink, and Apache Samza. In addition, several other data processing frameworks have been developed by companies and are now considered de-facto standards. Azure Databricks is a data processing solution provided by Microsoft. It combines the latest Data Bricks product with the Azure cloud ecosystem. Azure Databricks allows developers to process big data and build data analytics and machine learning products based on the data, with much higher developer productivity. Azure Databricks provides a collaborative environment, with tools for using notebooks and an interactive workspace for collaboration. Azure Databricks uses a cloud version of Apache Spark, which allows for easier deployment and faster operational efficiency and scales automatically depending on the workload. Azure Databricks is also integrated with Azure Data Lake Storage Gen2, making it easy to access data stored in ADLS Gen2, which is the recommended storage solution for data lakes after the introduction of the latest version.

5.1. Azure Databricks

Desiring to offer a collaborative analytics solution that permits both shortcuts and creativity in data processing tasks, Microsoft incorporated a managed Databrick service called Azure Databricks into their cloud platform. In Azure Databricks, users can access connected Blob or ADLS Data Lake Service storage containers from a notebook interface that enables the remoting of the parallelized Spark job coordination, debugging, execution, and visualization of job output with the granular progress information inferred from residing in the notebook itself.

While notebooks reduce entry cost for workloads that require a lot of extra libraries not included with vanilla Databricks and exploratory data analysis before the data processing orchestration pipeline is written in a more Pythonic fashion, data engineers, or scientists that want to deliver production-grade pipelines in an orchestrated, reusable, user-friendly form still must process data frames and create tables with Spark Data Frame API calls in a notebook [6,9]. This is less of a concern at the present time for Databricks on Azure, which appears to maintain the code jumping visualizations with adjustable scopes and linked editor functionality present in Databricks-independent notebooks, as the Data Frame pipeline-building, temporary table inspection, and exploration procedure at the heart of Databricks still require visualization and exploration

techniques that are more cumbersome to employ than in more purely notebook-centered tools for certain workloads.

Despite the costs and challenges of maintaining a truly editor-technology unplugged platform, Azure Databricks still presents a seamless, notebook-centric approach towards accelerated exploration of many analytics problems, particularly those involving external cloud APIs and auxiliary library ecosystems, at least within the available library constraints defined by the user's session. Collaboratively, easily tinkered and debugged pipeline-oriented workflows are also a great asset for the currently adopted Agile product creation paradigms.

5.2. HDInsight

HDInsight is Microsoft's distribution of Apache Hadoop, entirely implemented as a cloud service. Typically, the process of running Hadoop is more complicated than downloading and installing a few machine images. Hadoop is a complex and somewhat anarchically designed piece of software. It typically consists of more than a dozen different components, almost all of which need some level of configuration. Those components need to communicate with each other over a variety of protocols, and require some combination of Java, C++, and Python files to be installed on each machine. Adding a new node to an existing cluster requires joining that node to the cluster, which usually involves significant time and effort. Generally, running a Hadoop cluster requires constant attention.

HDInsight performs all the necessary configuration and setup work for you, and allows you to concentrate on the actual task of running your analytics jobs using Hadoop. You can choose the number and type of machines that you want in your cluster, as well as which subset of the complete set of components you want installed. Once your cluster is allocated, HDInsight handles the difficult part of managing the communication and coordination between the machines. It handles the initial setup and configuration of new clusters, the reconfiguration of existing clusters, the dynamic allocation of resources via the YARN scheduler, and the removal of decommissioned nodes from use in clusters that have been running for some time.

The fact that HDInsight is a fully managed service means that you have to give up some local control. However, the benefits often outweigh the costs: you can rapidly build huge clusters by calling an API or filling in a form, and they can be just as rapidly deleted when you are finished using them.

5.3. Azure Stream Analytics

A component housed in the Analytics Platform System, the Stream Analytics Cloud service was initially created, and is today managed, by the Business Intelligence division. Stream Analytics is designed to work in virtually the same way as SQL-like and what is sometimes referred to as a real-time data warehouse, providing dynamic query capabilities over streaming data entering into a privately hosted instance of the Analytics Platform System, or Data Warehouse.

Creating a Stream Analytics job is a three-step process. First, a streaming input, and an optional reference input, must be defined. Each input is implemented through a plug-in that allows the service to connect to the input data stream and, optionally, persist input data into Blob storage. Once the inputs defined, a SQL-like query must be defined over the inputs, describing the computations that should be applied to the input streams, and how the output streams will be derived from the input streams. Finally, the outputs must be defined, which might be a final output sent back to a third-party service, as a persistent, optional intermediate result written to a set of Blob files, or as a final, persistent result written to a SQL Database or Cosmos DB.

The first task in creating a Stream Analytics job is defining the inputs. As of this writing, Stream Analytics supports four input sources. Alerts from Azure and Syslog alerts from the Alert Systems Management and the syslog protocols, default alerts. Custom alert and state data that is also generated by an Analytics Platform System manager and Call Home support engineer, and optional sensor data. Data for alerts that are also generated by an Analytics Platform System manager and Call Home support engineer.

6. Analytics and Machine Learning

While data is at the core of the big data ecosystem, the real value is derived by gaining insights from the data. The value is unlocked with the help of analytics, which is broadly defined as the systematic computational analysis of data or statistics. The two traditional categories of analytics are descriptive analytics, which answer the questions of what is happening, and predictive analytics, which answer what is likely to happen in the future. Advancements in machine learning, statistical modeling, and computing power have enabled the emergence of a third category—prescriptive analytics—which addresses the question of what is the best that can happen?

Prescriptive analytics uses data, and often results from predictive analytics, to recommend actions that can provide the optimum outcome. Certain platforms are prescriptive in nature, as they analyze user behavior patterns and then suggest the ideal ad or update. Over the past several years, we have seen a dramatic rise in the adoption of various forms of analytics, mainly driven by the big data explosion. The availability of a wealth of data across industries, combined with powerful open-source analytics software libraries, increasingly affordable cloud infrastructure, and enterprise focus on gaining insights from data to improve business performance, has led to the rapid adoption of analytics and prescriptive analytics—data-driven decision-making—across different domains, including life sciences, retail, marketing, finance, and business operations. The AI-first generation solutions are centered around providing an optimal outcome by emulating human intelligence, and use ML at their core. These solutions are increasingly being adopted in select domains such as self-driving cars, healthcare diagnostics, and recommendation systems.

The big data ecosystem has a comprehensive set of services that support different aspects of analytics that include data exploration and preparation, business intelligence, descriptive and predictive analytics, analysis pipelines, and machine learning. In summary, these services can be grouped into two distinct categories—one focused primarily on predictive and prescriptive capabilities, and the other focused mainly on descriptive analytics—but still provide some degree of predictive capabilities. However, based on its modular architecture, customers can use multiple services that best address their needs.

6.1. Azure Machine Learning

Azure Machine Learning is a cloud-based platform designed to streamline end-to-end machine learning workflows. It assists data scientists in various crucial activities, beginning from data preparation tasks to data preparation code management and version control. It also helps in model building by giving exploration options and a drag-and-drop interface and offering GPU computing in the cloud for training and inference tasks. Azure Machine Learning handles model management, allowing data scientists to look up older versions of models. It also helps with creating reproducible inference endpoints.

Azure Machine Learning helps with data preparation through a set of capabilities. It provides several function-based data preparation SDK libraries that are optimized for use with dataframes in either local or remote compute configurations. It has a data versioning and tracking capability called Data Store. Azure Machine Learning provides an environment and an integrated Notebook experience for running Jupyter-based code. In addition to that, it has several

model gallery entries that you can modify, test, and run on your data. Azure Machine Learning also provides reproducibility for long-running experiments through its built-in experiment tracking. You can also monitor your existing notebook jobs through its web interface for monitoring job inputs and outputs.

Azure Machine Learning has a drag-and-drop interface called Designer for creating pipelines that can be seen in both a pipeline or a flowchart style. Azure Machine Learning also exposes several steps through the Portal, and other services through the capability. You can combine these capabilities to help create Intelligent Analytics Reports for Microsoft Power BI.

6.2. Integrating AI with Big Data

The relationship between AI and big data analytics goes both ways. For machine learning and deep learning to find success, the amount of data available needs to be huge, and the data needs to be high quality. Lately, we've seen large datasets being made available by various organizations and services. And there is much more data available about people's choices and behavior that organizations collect during their daily operations. There are many companies that try to be the first movers in AI to lay their hands on this data to extract insights and predictive capabilities. At the same time, the technological infrastructure needed to support the computer requirements for model training does not come cheap. Clouds give businesses renting capabilities that allow for flexibility in their scale-up-and-down approach to their IT requirements and high-performance computing power.

Large organizations have innumerable servers in data centers around the world. Startups can also take advantage of these large-scale networks by hiring the services of these organizations as their partners. Proven frameworks and platforms allow the enterprise and the user to focus on the business pain point, data, and algorithms while the technical details of hardware and set-up, available services, distributed programming and processing, and environment setup are taken care of. This is why the use of big data and cloud computing analytics can help integrate the AI model training and deployment in the enterprise intelligence ecosystem offering the convenience and speed of deployment needed in the design of intelligent applications. The cloud provides the storage and compute elements needed to collect, process, and analyze the vast amounts of data available for model training, and then either use the cloud for inference or deploy the services locally in a smart edge device.

6.3. Use Cases for Analytics

Analytics can be defined as the exploration and use of large amounts of data to reach conclusions. It represents the second aspect of a data-driven organization,

after making data available. Whereas data availability involves asking what data is available and how can I use it, analytics is concerned with asking what can I do with the data available. Analytical projects may help answer questions such as those listed.

The insights gained from exploring data are varied. One case for analytics is descriptive analytics. Descriptive analytics is the simplest form of analytics, helping to answer questions like what had happened to known entities in the past? A characteristic of descriptive analytics is that it can account for aggregates or groups of entities in a time interval or over a longer period, but it does not allow tracking the same entity over time. Reports that count distinct types of events that happened in a predefined time interval are examples of descriptive analytics. Descriptive analytics is also the backbone of dashboards.

Accessible machine learning is about making machine learning available to people irrespective of their skillset. Businesspeople can build models without being dependent on data scientists, while data scientists are freed to invest their time in building sophisticated models using the whole data science toolbox. Here we briefly present the layers of abstraction available for analytics, starting with the most accessible, continuing with machine learning, and concluding with models. What model add-ins do is allow limited areas of the user interface to be changed to expose simple model building capabilities, replacing intelligent heuristics to build the model.

7. Security and Compliance

Security is a foundational concept for cloud services and has many aspects and models. Although, as users, we shouldn't have to think about physical security aspects, including how the company making cloud services ensures that you don't steal the hardware that the provider has built to run its services, as service consumers, we need to evaluate how the services ensure our secure use. This includes how they ensure the integrity, confidentiality, and availability of our data, how those services are architected, how we interact with them, and how we are authenticated and authorized to access what.

The shared responsibility model is a concept in cloud security that defines the division of security and compliance responsibilities between a cloud provider and its customers. With this model, the provider is responsible for securing the infrastructure that runs all of the services offered in the cloud, with the customer assuming responsibility for security of the data they put in the services, as well

as security of some services themselves. This includes configuration of security rules and settings specific to those services, as well as identity and access management responsibilities. This allows the provider to offer economies of scale and operational efficiency that allow customers to offload the security of underlying infrastructure they cannot control, while enabling customers to use the services to build more secure solutions.

With security built into everything we do, from our infrastructure to our applications and into how we manage your service. We consider security throughout the lifecycle of every product we build and implement industry-leading assurance programs, access controls, physical and online security, and operations monitoring for all services. Our offerings are anchored in an enterprise-grade security foundation that provides real-time visibility and control, unparalleled protection, automated intelligence, and management simplicity.

7.1. Data Security in Azure

Azure includes many features to help users secure their data. Since Azure runs in a multi-tenant architecture, this multiple-tenant approach offers some unique security and data privacy challenges. These include shared access security issues, questions of the adequacy of data security measures, and geographical issues such as data ownership and authority.

To help customers with these challenges, information is made available on the security standards, procedures, and technologies that can be used to secure Azure data. Resolute security teams and processes are put in place to help ensure all data is in a secure environment, with physical security controls in data centers. Data encryption is applied to restrict data access. Connectivity to the Azure services is secure with protection from unauthorized access through VPN or Express Route connections. Network security features are provided at various levels of service using Azure infrastructure. A distribution of data instructions is provided by Document DB to different servers. Data is automatically sharded by creation of sets of data. These different segments are securely shared across servers to create multiple copies, and then seamlessly reassembled by Azure at the time of query.

Data access is restricted. All servers responding to requests for data are needed to authenticate. Inside these servers, a second level of data access is enforced by a security token. Data that is resident in Document DB can only be trusted if packets are encrypted. A fully managed data service, Document DB, implements automatic backups of the document collections. Data is tracked and rolled back

if any unusual activity is detected. Servers are patched automatically to keep security, performance, and stability at the desired levels. Untrusted requests over an extended period receive a “deny all” response. The Document DB service runs on a 99.95 percentage availability guarantee.

7.2. Compliance Standards

Technology vendors from cloud providers to SaaS companies often claim to offer safe and compliant solutions. However, maintaining strong compliance with the legal and regulatory safeguards related to security is as important to the cloud vendor as it is to the customers that purchase cloud technology services. Without this commitment to compliance, customers are left to their own devices to ensure the security of the environment in which their data and systems are operating. Even then, customers may not have the necessary tools and visibility into the underlying physical or managed services to ensure that the proper security measures are in place at either layer of the offering.

Compliance should not be confused with security nor simply considered a box to check off. Compliance is the establishment of a series of safety and security controls to mitigate prior breaches and security errors; and then verify those controls continually through regular assessments. Compliance typically has a start date which can be when a breach occurs. Putting in the framework of controls can be long, tedious, and expensive. However, once in place, it must be actively monitored and its application verified by third party external assessments. After the assessment, a report shows the compliance posture at that time to mitigate risk. A compliant vendor can show that vast resources are put into the continuous assessment of controls in order to show their clients how well they are doing and potentially to improve compliance further.

8. Real-World Applications for Big Data

The previous section mentioned that several companies are employing big data analytics to derive useful information and make informed decisions. But you may be hungry to know more and ask: What do these companies do with their data? In this section, we provide several real-world case studies and examples to demonstrate the power and importance of big data analytics in various industries. Armed with such know-how, you can embark on a big data journey with your company to transform your data better to achieve your business goals.

More than 80 of the top 100 companies in the Fortune Global 500 list have a big data initiative. Some experts also claim that big data creates more value in non-traditional data analytics industries. The impact of big data and its analytics goes well beyond the boundaries of established enterprise data analytics – sales and marketing organizations – to finance and risk, logistics and supply chain, and product development, engineering and manufacturing. Traditionally conservative areas are discovering new insights that enable transformational business impacts. Big data uses include product and service innovation, customer experience, strategic and tactical decision making. Some specific examples of using big data analytics include the following. The big data set used includes data from sensors, machine logs, data from supply chain links, and maintenance data. Specific algorithms including predictive algorithms are used on the data for predictive modeling and pattern recognition. Data visualization tools are used for presenting predictions so that business operations can leverage the predictive models based on the big data analytics.

One company analyzes spending patterns of thousands of users in real-time to predict fraud. Another has a department dedicated to using big data analytics to make decisions in multiple areas including risk management, technology and operations, marketing, and so on. A director at the company says that big data helps them better understand customers and translate that data into better service. The company employs techniques including regression models to analyze and predict customer and market behavior.

8.1. Case Studies in Various Industries

With better analytics capabilities and sharper solutions, many organizations now use Big Data tools and solutions to optimize their business processes. An increasing number of 'big data stories' from various industries highlight this trend. Here are some examples: advanced analytics are enabling innovative companies to capitalize on their vast amounts of data. Insurance companies are seeking new measures of creditworthiness in order to help millions of people without history. The latest charge-card customers in need of a car loan, who would otherwise pay high interest rates for borrowing, cost the industry about $100 million a year in delinquencies. Sharing data lowers advertising costs for retailers, increasing, therefore, cooperation between stores and their suppliers.

Predictive "science" has become important in technology companies, perhaps more than in other industries. Old-school software companies and user-centric hardware companies are forming predictive analytics teams, racing to apply Big Data techniques to better manage the life cycle of everything from products to promotions. Companies are constantly fine-tuning what they sell, how they sell

it and how much they sell it for. The software buzz sometimes ignores the fact that there are superb technology-intensive organizations in finance, transportation, telecommunications, energy and other industries. These companies are already deeply involved in integrating predictive analytics into their day-to-day business. Sustainable Companies Using Predictive Analytics Are the Majority of Companies.

8.2. Impact on Business Decision Making

Business is all about data, be it financial, operational, geographic, social, customer or some other type of data. Presently organizations are swamped with data from various sources; transaction systems, social media, GPS based mobile technology, third party service providers, indirect and complementary service providers, etc. Given all these data sources, extraction of information with insights becomes a daunting task. Therefore, business decision makers have the need for tools and service providers who can manage these enormous data assets by filtering the important pieces of information required.

Focus on decision making applies not only to Finding Answers but also on Specialization and Monitoring areas of focus elaborated later in this chapter. One large organization can have multiple business lines with specific niche focus areas and over a period develop high specialization in a particular service. Such an organization may have the control and ability to monitor real time data in multiple niche areas of business of various other organizations. Such a service organization may employ statisticians, mathematicians, natural scientists, social scientists, computer scientists, and cognitive scientists who develop solutions and/or products providing a big data analytics infrastructure to manage filtering of data for various organizations or specialize in niche areas that provides big insights with respect to various business processes.

9. Challenges in Big Data Management

Big Data is omnipresent; however, collecting and storing massive amounts of data is not sufficient. Organizations are also looking at methods to extract actionable insights from this vast and varied data. Apart from the sheer volume of Big Data, there are also other challenges that organizations face in the management of Big Data analytics. Some of the critical challenges are based on data quality issues, scalability to manage the volume, speed, and variety of data, and the general lack of skill in the workforce who are responsible for producing results using Big Data tools and technologies.

Big Data is often collected from a vast number of sources, with variable quality controls. Social media is one example of a source where any user can input unverified information that is not always factually true but can still go viral. Tweeting trivial facts about an important sporting event, for example, does not present any reliable knowledge nuggets, yet it may still get a lot of traction and thus may be included when performing frequent term searches. Querying social media and using that information for Big Data analysis without any quality filtering can be very misleading and result in poor insights of the Big Data. Another problem that organizations observe is that often the expected output is inaccurate due to poor quality data being the starting point. Users of Big Data need to have great tools to perform adequate filtering of Big Data to ensure higher accuracy of results.

Scalability issues are common in Big Data as organizations do not have the right environments in place, or the right tools, to handle the size of data they expect to cover, and they feel that Big Data is not scalable. Architectures are flexible and scalable but many organizations have the misconception that they are only using these tools to produce reports when in fact they are doing a lot more than simple reporting analysis. Data from different organizations have different formats and structures, so converting the same data from different organizations to a common format is a challenge by itself. Data sources can be internal to the organization such as customer information or external data from different sources; either of these can have data in irrelevant formats.

9.1. Data Quality Issues

1. Introduction Big Data is one of the most popular buzzwords in recent times. This phrase refers to the huge investments made for research, development, and technology around the massive amount of data generated by organizations. There is a never-ending increase in the number of things that have to be monitored. Emerging trends such as artificial intelligence, the Internet of Things, and machine learning have led to a huge demand and quest for the massive volume of data. The cloud technology advances have made it possible to derive useful information and wisdom from this data explosion supporting organizations in its major business decisions. However, organizations face especially important challenges managing their "Big Data". These issues range from the quality and scalability of data storage, preparation of this data, inference, to the skill gap in their workforce. This chapter discusses some of the major big data management challenges faced by organizations today. These challenges revolve around the concept of big data veracity and workforce gaps, which if mitigated could support organizations and assist them have better knowledge in the big data space,

improve their decision-making processes, and create overall strategic advantage over their competitors. These identified challenges are of paramount importance for organizations, and a concise understanding of these challenges could encourage further research studies in these areas thereby addressing them. 2. Data Quality Issues To analyze data, a lot of processing happens during the earlier stages of data preparation and organization. Most of this data is collected from various resources including unstructured. As the saying goes, "garbage in, garbage out", the saying holds well with data as well, thus making data quality and trustworthiness of data one of the important requirements. The usefulness of data is determined by factors such as accuracy, completeness, reliability, validity, what relation it has to the task, and uniqueness. Big data veracity challenges could include unreliable or excessive various sources of data or random sampling in data collection.

9.2. Scalability Challenges

Scalability means that the solution can not only support increasing workloads but can do so cost-effectively. A big data technology needs to be able to deliver both consistency and scalability in microseconds and milliseconds. Scalability can be achieved using two models: replication and partitioning. Replication is the easiest way to implement scalability. Data is replicated across a number of nodes in a traditional relational database and any read request hitting any of these nodes should get the same answer. When the database grows, you simply add nodes and ensure that the data is replicated in each of these nodes. The downside of this method is cost. With each node having the same amount of data, cost can grow exponentially. Users also encounter challenges in managing reads and writes across multiple copies of the same data. In the partitioning model, when the database grows to a certain limit, it is partitioned across nodes. Each node has a piece of the whole data set. Each node is accessed based on the part of the dataset it stores. Partitioning is cost-effective but also raises serious consistency issues. It is a very thin line that separates consistency and scalability. Typically, large enterprises that have invested millions in relational technology for their transactional systems continue to invest in replication systems for their data warehouse solutions. They have invested heavily in enterprise processes and they want to ensure the same level of consistency around enterprise reporting as they do around transactional systems.

9.3. Skill Gap in Workforce

The big data revolution is not about technologically transforming our data management processes but economically unlocking the value of our data and using it to make decisions. The management and analysis of big data is a new

discipline. There is a shortage of skills needed to enter this new field. Big Data has received considerable attention for its potential to expose new prospects from the data. Leading organizations are starting to invest in big data and develop capabilities. While a wide range of data management and data analysis products are available to assist organizations in enhancing data-driven decision making, few individuals in the global workforce possess the expertise to make these systems work.

The field of data analysis is new. Many data scientists, data modelers, data architects, may have come from a statistical background. There is currently a shortage of qualified data science professionals. Its study is also not restricted to traditional engineering roles. The skills gap is experienced not just by technical colleges and reputable institutions offering data science programs but also by industries venturing into data science within their existing workforce. Enhanced predictions of experience levels will enable improved data science education programs, which in turn should lessen the skills gap. The skills gap poses a serious challenge for big data adoption in industries and enterprises.

10. Future Trends in Big Data

Though present in organizations of all kinds for the last two decades, Big Data is more a tool than a phase, meaning that it is here to stay and evolve as a technology as well as a business strategy. With that, new technologies are being developed to cope with new challenges and demands at the corporate, social, and market level. Additionally, the evolution of those new technologies is also a trend and together they are shaping the future of Big Data, as we understand today [10-13]. In this section we will discuss with the shining stars as well as the giants in the backstage of Big Data.

10.1. Emerging Technologies In the chronicle, it points out to 10 technology and shares some positive account with them from the Big Data Perspective. First, it highlights that the past decade was inconclusive in what regards AI predictions while many practical applications with great business impact are gaining prominence. Additionally, with the arrival of 5G, along with the expansion of cellular networks, satellites and mobility data access become more accessible.

Sharing the future stage with AI and 5G, health and life sciences and transport for autonomous vehicles are also technologies that will take the lead in Big Data applications. In health and life sciences, healthcare happens at home, bringing follow-on service opportunities in insurance and pharmaceuticals. For transport,

long demanded both in the consumer and in services arena, self-driving cars will put on the road new concepts about logistics, shaking common wisdom on local shops among others.

The other expected technologies evolve either on stage or on backstage, supporting the lead actors in their performances. Empowering a myriad of cyber marketplaces to secure and encrypt brand loyalty, transactions and contracts. Allowing easier and faster security and data privacy build-up, affordable access to resources, as well as quantum computing. Dragging the lines between physical and digital identities. Crucial to protect the digital infrastructure and consumers, ease and increase Big Data applications. Finally, the virtual workforce, demanding innovation in up-skilling and re-skilling practices but also enabling scaling new digital capabilities to the entire organization and business operation.

10.1. Emerging Technologies

The term “emerging technologies” refers to those technologies newly arriving at the marketplace and beginning to gain a foothold in the economy. New applications of existing technologies are considered emerging if they raise societal and commercial interest. Big Data technologies support a diverse set of new applications made possible by the big data phenomenon. For instance, new transaction engines help organizations make effective use of data in motion; new management frameworks allow organizations to build big data applications that parallelize large data sets across functional clusters; new memory-based software engines allow organizations to speed ad hoc big data queries; new systems software stacks augment existing database engines and relational data tools by enabling new analytic techniques including machine learning and secondary indexing over NoSQL databases, and add big data document retrieval and natural language processing; new business intelligence tools extend dashboards to the present focus on big data, providing a select range of new filters and advanced analytic capabilities such as optimization; and new data science techniques include unusual pattern and anomaly detection, predictive modelling, and trend extrapolation.

In addition, new technologies are reinventing the way that some business functions are managed. These disruptive new applications often deliver more than just savings. They can radically change company business models. Companies are benefactors of data coming from early examples of the data ecosystem such as smart devices and with the emergence of machines that are connected to the network. As the machines generate data, organizations are using big data analytics to glean business insights that can improve business productivity and customer experiences or generate entirely new products and

services. Today's organizations are constantly looking for data-experienced professionals who are familiar with creative thinking and effective collaboration. Thus, analytics technology is altering the way talent is hired, evaluated, and developed.

10.2. The Role of Cloud Computing

The first ten years of the breakthrough that big data is were foreshadowed by cloud computing. In fact, both concepts are strongly related. If it is debatable whether big data existed without the cloud, there is consensus that the cloud has provided the essential foundations for research and experimentation in big data. Providing a framework for painless experimentation is what allows researchers to explore new questions and ideas. The other aspect of cloud computing that is often referenced by researchers and developers is that of accessible scale. Cloud computing changes the problem of scaling up systems from one of financial feasibility to a question of timing and who pays. The existence of cloud computing services removes any non-technical barriers to implementing scalable versions of large-scale data processing algorithms. The two key defining characteristics of big data are thus satisfied: cloud computing enables access to vast amounts of data, and it allows that data to be processed quickly instead of taking years to complete.

The explosive growth of various fields within computer science is in turn due to the widespread availability of data, computer power, and information resources offered by the service model of cloud computing. An externality that benefits all cloud users is access to continuously improving foundational services and component technologies. The structure of fundamental technologies is such that there are many more users of the one-to-few core foundational technologies - such as information storage, security, licensing, databases, compression - than there are developers and experts in those technologies. Service technologies - that offer use of components rather than use of access to the expertise needed to create and customize the components - released those who did both create custom components for a market and design and maintain component-based systems for particular customers from the huge overhead of marketing, selling, and supporting component-based services to their customers.

11. Conclusion

As we have seen, big data is the term used to describe the volume, velocity, and variety of data that flows around us every day, and we are only beginning to

understand what we can gain from this abundance of information. In this book, we have looked at the main challenges in dealing with this massive quantity of information, as well as the opportunities that they afford. Technology is helping us to deal with the challenges, but at the same time it is easing our ability to learn from the data. Organizations that are initiative-taking in their approach to managing data and the way they use the insights the data can provide will be the future leaders. The ecosystem provides ample choice to everyone, no matter the type or size of organization. From simple analysis, using other products as part of the data pipeline to the massive cloud-based data services, it is up to you to decide the specific tools you will use to help you deal with the opportunities and challenges of the big data world. It is widely recognized as a leader in innovation, and the continuous update and release of these tools is proof of this. Yet they need to be in your arsenal to be adequately explored and used. Your biggest challenge now is to keep up with the pace of change and innovation this area is experiencing so that you and your organization can leverage the power these tools offer to help you improve your results and insights, prevent issues, and help you come out on top in your respective area, whatever it is.

References

[1] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.

[2] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.

[3] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.

[4] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.

[5] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.

[6] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.

[7] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.

[8] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.

[9] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in

Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[10] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[11] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[12] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[13] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[14] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[15] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.
[16] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[17] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[18] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[19] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[20] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[21] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[22] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[23] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[24] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.
[25] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using

scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).
[26] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[27] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[28] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[29] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[30] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.
[31] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[32] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.
[33] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.
[34] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[35] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[36] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[37] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[38] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[39] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[40] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.
[41] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[42] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.

[43] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.

[44] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.

[45] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.

[46] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 2: Azure Data Architecture and Modern Data Warehousing

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction to Azure Data Architecture

Over the years, organizations have gathered data for various purposes and used a combination of systems for these purposes, namely, business applications, relational databases, data lakes, legacy systems, etc. Data is growing exponentially with the expansion of technology toward cloud, IoT devices, and digitalization. Organizations have explored, leveraged, and used this data to meet their objectives and be successful [1-2]. The most successful organizations in the world have used their data to derive business insights, customer engagement, product enhancement, and for many other objectives, to enable them to stay steps ahead of their competition. All organizations are looking to do the same, so this has led to a surge in demand for data-related experts and technology.

A variety of data technologies and platforms are available for data engineers, data scientists, and BI analysts to derive valuable insights that affect the organization as a whole, increase revenue, and decrease costs [2-4]. These technologies enable the extraction, transformation, and loading of data into a scalable data platform that services both operational reporting and advanced analytics requirements. This is where the Azure Data Architecture comes into play [5-6]. Microsoft Azure has extensive capabilities in the data technology landscape and is continuously expanding its services and features to cater to this market. With its variety of services, Microsoft Azure solves most of the organizational requirements for data and analytics. This allows organizations to focus more on business objectives rather than on data challenges.

Today’s Azure Data Architecture is based on the principles of analytical data engineering, which enable solution architects to design and build scalable, accessible, usable, and efficient analytical data solutions [7,8]. This chapter introduces the key concepts of Azure Data Architecture, the principles of analytical data engineering and the Azure annotation model and concludes with a discussion of the benefits and value of analytics for the business.

Fig 1. Azure data architecture

2. Fundamentals of Data Warehousing

Data are processed in an organisation to provide information for business operation, short-term reporting, strategic oversight and long-term planning. Different kinds of processes may be required to satisfy these objectives, and different repositories may be used for data that fed into these processing activities. Data warehousing creates a special repository and process flow for a specific kind of processing for a specific management objective [9-12]. Data are stored in a warehouse primarily for access by the business analyst to satisfy the need for information rather than for transaction processing. These data are extracted from different sources, integrated and stored with the significant structural and functional characteristics remaining accessible over a long time period. Data warehousing is generally concerned with the storage of historical data that contain important nuggets of information about customers, products and markets [7,13-15]. These nuggets are required over a long time span, such as average product quality over fifteen years, for decision making relating to quality maintenance, product recall, warranty, etc.

Data warehouse architectures developed in the early 1990s involving the creation of relational databases, initially with an OLTP orientation focusing on data retrieval for transaction processing. While there has been a historical distinction between operational and decision support processing, it is important to realize that both are fundamental for the efficient and effective management of any economic organisation [9,16-18]. Data warehousing is one of the important innovations in the area of information systems due to the capabilities it provides for superior decision making. Information, rather than products or services, is increasingly being viewed as the major output of organisation processes, and computer storage has become a high volume journal that contains much of the historical record of the organisation and is also one of its most valuable assets. Data warehousing applies new processing technologies to the integration of this historical record for decision support.

3. Key Components of Azure Data Architecture

Migrating from an on-premises data architecture to an Azure data architecture has intrinsic complexities, and without doing the right upfront planning, we may not only end up dealing with legacy solutions but also even legacy problems and limitations. The advantage of building fresh and new is to deliver innovative solutions, which is made more possible using cloud capabilities and services.

Azure provides us with those capabilities, and Azure Data Architecture is rich in its design and informed capabilities.

A carefully designed and planned Azure Data Architecture provides us with a solid evergreen foundation that is future-proof as it is also enterprise-grade that can scale, and is flexible enough to adapt and incorporate new services as innovations are released [2,19-20]. It is also possible to architect a data architecture that can suit small enterprises or businesses to realize the many benefits Azure provides while being cost-effective. Ensuring that analytics workloads perform efficiently, regardless of where they are executed - that is, in Azure, on-premises, or hybrid, is central to Modern Data Architecture. We will see how to achieve this for Lake-centric and Warehouse-centric Modern Data Architectures with the engines in the Azure Services.

The question then is what services are available, and how are they used in Azure Data Architecture? Identity Management is the primary concern and starting point for any enterprise, and Azure Active Directory is the fundamental service. The other needed foundational services are Azure Data Lake Storage, Azure Synapse Analytics, Azure SQL Database, and Azure Data Factory. These four components are core to any Azure Data Architecture and enable us to successfully migrate our existing data solution to Azure and realize its potential as a data-driven organization.

3.1. Azure Data Lake Storage

Data lakes and data warehouses have a proud collaborative history that predates big data, machine learning, and artificial intelligence. Data lakes are used largely for data curation, as a set of building blocks for activities such as machine learning, and as storage for unstructured data. Data warehouses are used for enterprise reporting, sharing information across systems, and enterprise governance and secure access. They contain known-knowns, with data stored in a highly structured way to answer specific questions. The two systems cannot replace each other but can fill in each other’s missing facets. Making a conscious, informed decision about which technology to use is key.

Data Lake Storage is a no-cost option to add semantics and APIs to Storage Accounts featuring hierarchical namespace. It is a superset of Blob Storage with some extra capabilities. It uses the same underlying object stores used for Blob Storage. Pipelines and other big data services leverage Blob Storage and Data Lake Storage via compliant APIs. Recently, the size limit for a single blob increased to 5 terabytes. Logs, video, audio, static images, and binary signed documents above a few hundred megabytes in size should be stored in Data Lake

Storage [9,21-23]. Data lakes are usually updated by bulk processes, by people using storage tools, and by change data capture processes. Data lakes contain copies of enterprise data used for activities such as running enterprise reports or building enterprise models.

3.2. Azure Synapse Analytics

Azure Synapse Analytics is the analytics workhorse in the Azure ecosystem. Synapse integrates data ingestion, data preparation, data management, data warehousing, data integration, analytics, business intelligence, and visualization into a single unified platform. Synapse allows data engineers, data scientists, data analysts, and business users to explore their massive amounts of data using limited code. Data engineers can use Synapse workspaces to build data preparation and data transformation pipelines that can be executed on-demand or on a schedule. Data engineers can leverage technology to simplify the data preparation and transformation processes.

Data scientists can use notebooks with various code to explore and analyze data in different formats and build and train machine learning models. Data analysts can use Synapse workspaces for data exploration and deeper analysis for ingestion into workspaces or directly to end-user dashboards using connections. Data scientists and data engineers can use the integrated workspace for collaboration and to deploy models into the enterprise for consumption. Business users can use workspaces for quick analytics and data visualization.

As a combined service, Synapse reduces the need for many discrete services and the complexities involved in incorporating those services into workloads. Data from different sources and in different formats can be discovered, prepared, and made available to business users in one integrated solution. The concept of the reusable semantic model called the star schema is resident in the Business Intelligence and data visualization communities.

3.3. Azure SQL Database

As previously discussed, structured data is not obsolete by any means. Relational databases are still the best storage system for storing and using dynamic structured data. Consider a banking system containing customer information such as names, addresses, and account information; all of this data is structured, meaning that it fits neatly into rows and columns. Furthermore, this data is constantly being changed, with customers closing or opening accounts and changing address information as needed. The most important aspect is that it is a managed service that reduces the burden of management. Choosing the appropriate hardware, setting it up, and keeping it available is already a challenge

in itself. With this service, customers avoid all of this and can pretty much concentrate on their applications only. Unlike other storage options, however, the underlying service is not free, as it charges for the storage consumed by the database as well as the resources used to run it.

This service is a database as a service (DBaaS) implementation of a very mature relational database engine. With this service, you get great flexibility [24-26]. You can easily scale the size of your database up or down, enable or disable geo-replication with just a few clicks, and even enable Intelligent Insights that monitor your database and provide recommendations on how to improve its performance. Additionally, this service is set up as a super high-available service by default; there is no need to set it up using clustering, like you would have to do with a traditional platform.

3.4. Azure Data Factory

Among the core components of the Azure Data Architecture, Azure Data Factory is the service biased to less technical users. Orchestrating tasks is becoming more and more important in data engineering activities, as obtaining valuable information from data usually comprises several steps, such as ingesting the data, cleaning it, transforming it, and merging it with other datasets. Azure Data Factory allows data engineers to define several tasks as pipelines through a user-friendly interface, without requiring programming skills, and to schedule them to be executed on a given schedule or triggered by an external event.

A point to highlight is that Azure Data Factory is not executor of the data pipelines by itself. The services that actually run the different tasks pipelines are Azure Databricks, Azure Batch, Azure HDInsight, Azure Machine Learning, Azure Logic Apps, Azure Batch, Azure Functions, and Azure Synapse Analytics. After Azure Data Factory redistributes the workload to any of these services to carry out the operation, users can directly connect to that service and trigger other batch events. For example, in a multi-tenancy solution where there are multiple tenants for an organization, the Azure Data Factory pipeline can trigger an Azure Blob Storage event, which subsequently triggers the Azure Functions to process the new blob data generated from the different tenants.

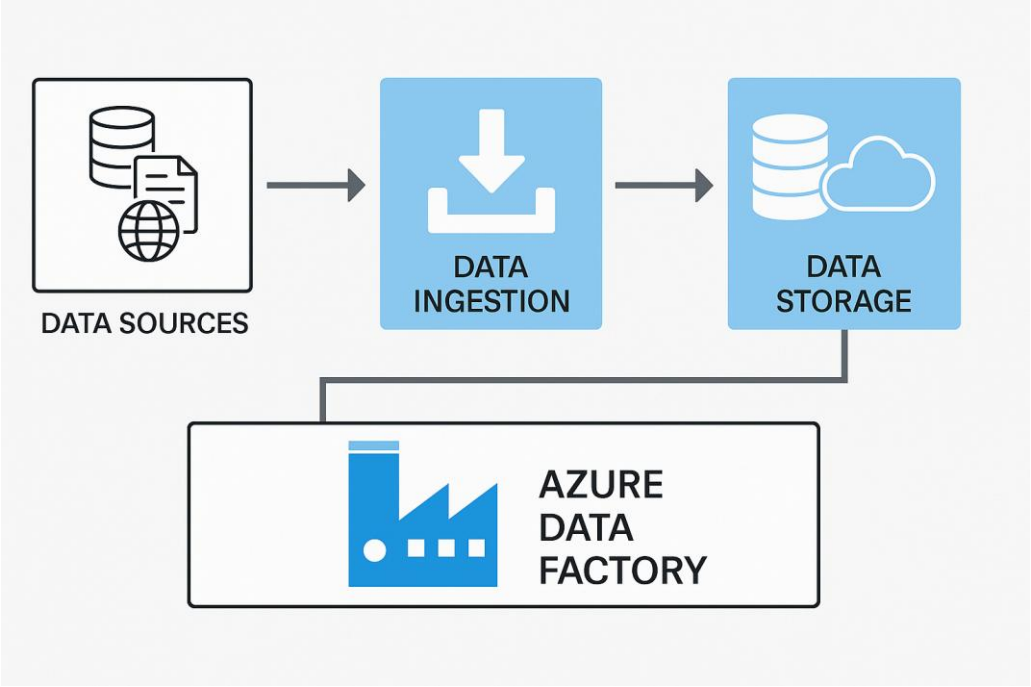


Fig 2.Azure data factory

4. Data Ingestion Strategies

Data ingestion is a critical step in the analytical pipeline where data is copied, moved, and transformed for subsequent analysis. During this stage, data sources are connected to the analytical platform, and the data is ingested to create a trusted and governed layer known as the data lake or data warehouse [8,27-30]. The ingestion layer is often referred to as the data lake or the data warehouse, but this is technically inaccurate [9,31-33]. The elements of the pipeline, such as orchestration and monitoring, data movement, and data transformation, usually live in an intermediate space between the source and the data lake or warehouse. These platforms are where the data is moved into an optimized state and metamodel, ready for reporting and analytics.

Inquire about the ingestion layer with any data architect or data engineer, and you will likely hear differing opinions about which ingestion strategy is best. Data ingestion encompasses about one-third of the entire analytical pipeline, and there will usually be multiple streams that represent a portion of the pipeline, which focus on customized ingestion pipelines for a specific need. The overall ingestion strategy is part of a broader optimization effort to architect the end-to-end

analytics strategy for a business. All businesses ingest events, logs, transactions, and activities about something; and if this is not true, they are likely not around for very long. For businesses storing vital information to help drive future business objectives, it is imperative to ingest the most business-driven data efficiently while enabling a faster time to insight for analytics, business intelligence, and data science.

4.1. Batch Processing

Data that you collect and process does not need to be all in the same manner. A common strategy is to collect and or process certain types of data in batch mode and to deal with different types of data in real time or streaming manner. Batch processing is when your code runs periodically or less frequently and processes one or more files or command line arguments and operates on data sets based on the ingestion time [34-36]. If you collect your stock trades every week or your server logs based on the date, running your processing job periodically makes sense, especially if the volume is not too large. If you want, you can create a data pipeline that picks up a specific week's log file every Tuesday at midnight and runs any machine learning jobs you want.

Batch processing of the data tends to be cheaper than real-time processing. It may be significantly more efficient on a workflow or such microbatching processes. However batch processing of many types of data may also be impractical. For example, have you ever gotten a flight delay notification via email while on the way to the airport for a scheduled flight? Or, have you sent an email to the support staff of an important site that is down because their updates were most recently made a week or two earlier? If the volume of new data is sparse and processing jobs are not on the critical path of your operations, batch processing works wonderfully with large stability and consistency at low costs [3,37-39].

Batch processing is how most of life has functioned for most part over the last century or so. People lived their lives oblivious to continuous changes. They would stop, look forward those weeks on end and decide whether changes were favourable before taking any action. Companies, governments and other organizations also worked with a lot of batch processing. Updates to finance databases were periodic, stock prices were determined at the end of the day based on periodic volumes, and clouds were seeded for the monsoon rains and tourists began flocking to the beaches again.

4.2. Real-Time Data Streaming

In most real-life scenarios, arriving new data is expected to be utilized by the receiving system as fast as possible. For instance, customers expect to see real-

time updates of their account balance or app transactions on their user interface screens. In such scenarios, one can think about a bank transaction as an event that should be transited from the transaction event source through various stages of the data processing pipeline until the final data presentation system where the user interface is running [36,40-42]. Depending on how soon the event is processed along that pipeline path, we can classify the bank event transaction into three different transaction types: Batch transaction processing, a semi-real-time transaction, or just a real-time transaction. All three types of data transactions are commonly supported by the data processing solutions provided by various cloud providers.

If you consider the following steps in the processing of events: The publisher publishes an event and enqueue it in one of the messaging services. Scripts implemented within a database append to a table that the user interface reads. The system design will generate dissimilar service costs for dissimilar transaction types. However, one might think about a trade-off between the system cost and the update frequency for the user interface. In this section, we will focus only on real-time data streaming. The terminology usually used for that type of data ingestion is Real-time Processing or Stream Analytics. To consume streams, there are a few alternative connectors that can be used [40,43-44]. The most common solutions to achieve real-time data ingestion comparatively to the batch are through pushing the data into message queue services from where the receiving microservice will retrieve and process the messages.

5. Data Transformation Techniques

A modern data architecture usually involves moving large volumes of data around and processing it in different ways. For example, the source data does not usually match the destination schema design, so the data must be transformed to match the target data format before being ingested into the data warehouse. The data must also be refined to fit the business logic and reporting needs of the company. ETL tools are important components of a modern data architecture to such an extent that data integration is one of the five main pillars or components of a modern data platform.

Data transformation is the process of converting data from one format to another or for another purpose. Why do we need to transform data? There are many valid reasons for transforming data. The data source and target systems may use different formats to represent similar information. Data in the source system may

not have the same level of detail or granularity as required by the target system. Data may also need to be cleansed before processing as errors are common. The structure of data may also change from one period to another or from one location to another, for example, conversion of flat-file family records to XML or relational database representation while consolidating [3,45-48]. Data in the source and target system may also need domain-specific processing, such as the taxes for different countries, account type classification, or the profit/loss % to be classified as gain, etc. The target system may also need pre-calculated or aggregated data rather than operational detail.

5.1. ETL vs. ELT

In this chapter, we explore data transformation, a same-day process of preparing data for analysis, and how computing innovations in cloud, data processing, and storage shifted the existing paradigm of ETL to ELT, which has made data preparation as database agnostic, disruptive, well-accepted, and easy as dragging and dropping. Despite the popularity of the ELT approach, and more broadly, data processing in cloud data warehouses, the ETL approach is still relevant. We discuss challenges and considerations of leveraging ETL approaches when working with data.

Traditionally, the data transformation process was often conceived closely bound to the actual data movement, with data moving from source to target being responsible for transforming the data. The embraced approach of data warehousing was "Extract-Transform-Load" and referred to as ETL for short to process data. In the ETL hierarchical model, various data management operational tasks were defined, whose feasibility and execution parameters were tightly coupled with their placement and execution onto a data processing engine, such as disk-based databases and ETL engines. Cloud data warehousing typically embraces "Extract-Load-Transform" and referred to as ELT for short. In the ELT hierarchical model, various data preparation tasks are defined, whose execution plans are loosely coupled from their placement onto a data processing engine, and easily parallelized and distributed across a heterogeneous computing grid, by the data processing engine. The storage hierarchy in ELT is also degrees of magnitudes different, supporting massive scale files backed by non-disruptive storage engine clusters that are substrate to cloud.

5.2. Data Transformation Tools in Azure

Numerous data transformation tools are provided in Microsoft Azure, which is organized into the following logical groupings: cloud-native, cloud-centric, and third-party solutions. Cloud-native solutions include the fully managed Kotlin-

based Azure Data Factory Data Flow service, the T-SQL-based Azure Synapse Analytics serverless SQL pool, and the open-source SQL transformation capability inside Azure Databricks notebooks. In addition to tightly integrated, cloud-native capabilities, Azure provides cloud-centric transformation solutions that are compatible with hybrid and on-premises environments as well. For example, SQL Server Integration Services is a popular on-premises ETL tool. The availability of the Azure-SSIS managed instance service makes it easy to lift and shift SSIS workloads with minimal changes to Azure.

While these tools are ideal for many common use cases, specialized transformation tools from third-party vendors may meet your needs better. Azure integrates with tools such as Talend Data Fabric, Informatica Intelligent Cloud Services, Matillion, and Alteryx. For some projects, it makes sense to choose one of these third-party data preparation tools initially and eventually transition to cloud-native tools after the business requirements are more clearly defined and concrete budget constraints are in place. For example, Informatica can help organizations quickly build cloud data pipelines and orchestrate data migrations to Azure before they decide to embrace a more autonomous cloud-native data pipeline orchestration experience based on Azure Data Factory. Furthermore, Informatica's AI-powered tools can help organizations automatically recognize data entities and eliminate the back-and-forth iteration often necessary to meet evolving business requirements.

6. Data Storage Solutions

Data storage is key in data architecture and data warehousing, from both a metadata management and a data management perspective. From a metadata management perspective, it provides the means to store all the data used and generated by the services of the ecosystem; this is key because the metadata defines the actual data models, pipelines, and all the assets available in the data architecture. From a data management perspective, it enables the different services to actually persist the data, which is done in different ways, depending on the type of the data and on the requirements of the services for which the data was generated or transformed. This topic explores the most common options and provides some best practices.

Structured Data Storage

The storage of structured data is often done in data warehouses or in relational databases. Data warehouses are big database engines that allow for multi-tenant

architectures and SQL access for multiple users simultaneously. They can optimize for reads and do some specific work to guarantee this optimization for complex SQLs, such as aggregations of big quantities of data and joins on big fact tables and relatively small dimension tables. Data warehouses can provide the fastest SQL access to data for analytics, business intelligence, and reporting purposes. They can generate a lot of overhead and latency when doing real-time or near-real-time analytics and business intelligence, especially when refreshing the data. Real-time or near-real-time analytics and business intelligence typically require database engines to have the data being queried, including indexes and aggregates, already loaded in memory.

Relational databases are smaller and include all the typical relational engines provided by cloud offerings. These engines can do query optimizations to deal with relatively small amounts of data but about all the data that requires transactions to guarantee the data being processed are real-time accurate.

6.1. Structured Data Storage

The Azure platform has a rich set of options for storing data. Designed to work in tandem with each other as well as third-party solutions, they provide a robust environment for gathering, moving, and governing data that needs to be analyzed. In the context of modern data warehousing, structured data often represents core business data gathered from operational systems, transaction data siloed in application-specific stores, or summary information generated through upstream processing. This data is often modeled in relational structures and is governed to ensure quality through common definitions and business ownership before loading into the data warehouse. Additional data may also be extracted from third-party or partner applications for enrichment or operational analysis.

Data modeling is a key method for designing a data structure that organizes and describes relational data elements and their relationships. The schema used to implement a data model outlines how data is stored, organized, and accessed. Traditionally, enterprise data has been modeled using an Entity Relationship approach, and schemas implemented using a normalized structure. This is especially true for data marts used to support enterprise reporting and analytics, with structure models created using stars and galaxies for ease of access and optimized for speed of query execution. Discovered data may follow a different modeling approach, especially if sourced from cloud-based transactional systems, SaaS applications, or for analysis of external third-party data. In these cases, a flexible and extended schema approach may be employed, as well as NoSQL storage options that allow schema-less storage. While these alternatives

certainly exist, the majority of data stored in Azure data stores is structured and relational in nature.

6.2. Unstructured Data Storage

For Data Storage Solutions for Data Analytics and Data Consolidation, there are different Solutions Architectures for Structured Data Storage and Storage for Unstructured Data. In the Microsoft Azure Data Technology stack, these Data Storage Technologies work smoothly to be part of a Modern Data Architecture.

Apart from Structured Data Warehouses, Microsoft has multiple solutions to build Unstructured Data Storage systems in Azure: Azure Data Lake Store; Azure HDInsight Data Lakes; or the Simple Solution of Azure Blob Storage. In this Data Storage chapter content, we will consider Unstructured Data Storage all Storage Solutions that don't include Structured Analytics Services. For choosing the appropriate Microsoft Azure Unstructured Data Storage, we will consider the type of Unstructured Data: Ingestion Patterns and Decisioning Patterns.

Microsoft Azure has a couple of Solutions for Unstructured Data Storage, with both Real-Time Access and Batch Processing, both with Data Lakes with Data Masking Encryption, without Data Processing or Azure Services. The Azure Simple Unstructured Storage solution is Azure Blob Storage. Microsoft Azure also has a full Managed Service Solution, Data Lake Storage with Azure Data Fabric solutions. Azure Data Lake Storage solutions are designed to meet the most complex Unstructured Data Storage requirements.

When a Company has a simple Unstructured Data Storage needs like Backup, a good approach is to use Azure Blob Storage. Blob Storage is the simplest solution to Build on Azure Cloud for Backup, with Block Storage Performance for large files of medium size data. Azure Blob Storage also supports File Sharing use case: Exposing and Sharing Files through SMB protocol using Azure Files. Additionally, Azure Blob Storage has also an Azure Blob Level Tiering System, with an End User Azure Pricing Model by the Amount of Data Stored in Different Tiers.

7. Data Modeling in Azure

Data modeling is an essential process for any data architecture solution and there are many such specifications required throughout any Azure Data Architecture in order to fully realize a successful solution. These range from the specification of the data structure required in the source systems for optimal data extraction,

through operational data models to support the business operations, to the enterprise data model which drives the requirements for a data-led architecture, and of course the logical and physical models that describe the structure of the data warehouse and potentially any additional layers of the architecture or any data marts. The focus in this section will be the specifications of the models that describe the data warehouse and its supporting layers.

The data model describes the actual structure of the data as it is physically implemented in the database, as opposed to a logical data model that describes the structure of the data that is required by the users and of the transactional processes that populate it. In the Delta implementation of the star schema, the data model will include fact and dimension tables that are specified in sufficient detail to enable optimal physical implementation for performance and storage cost.

The most common data modeling techniques have used these two approaches for many years, deriving fact and dimension table specifications from a dimensional data model; and making use of the Data Vault modeling technique to model the raw data warehouse layer. Dimensional data modeling was first pioneered in the early 1990s. The Data Vault modeling technique was first proposed in 1996. The essential aim of both these techniques is to create physical models that enable rapid data access by the business requirements.

7.1. Dimensional Modeling

Dimensional models are widely used in data analytics to support end-user analysis, often through user-friendly interfaces, such as those present in business intelligence tools and dashboards. They provide an intuitive framework that enables non-technical users to analyze metrics easily by criteria or in groupings that are meaningful to them, while also concealing the complexities of the underlying schema and data movement. For example, an airline facing decreasing revenues wants to understand whether this is linked to changes in passenger demographics and the reasons for decreased spending. With little knowledge of the schema, they can generate compelling, yet straightforward, reports to aid their decision-making.

Supporting frequent changes in analysis requirements is a core principle of dimensional modeling. Business questions best define dimensional models, rather than a generic well-formed relational schema design. Star and snowflake schemas are easy to understand, navigable, and suited to data mart implementations. In a star schema, a central fact table, consisting of transactional metrics together with foreign keys to dimensional tables, is associated with

surrounding dimension tables that provide the descriptive attributes for the associated metric transactions. In a snowflake schema, the surrounding dimension tables are further normalized to be multi-layered, broadly partitioned hierarchies of attributes linked by foreign key relationships. Snowflake schemas minimize redundancy at the expense of hindering performance. With semantic optimization, dimensional joins and aggregates can be partitioned and pre-optimized for analytic workloads. Thus, they can achieve performance on a par with, and often better than, OLTP systems. Consequently, OLAP systems with hyperspecific schema designs dedicated to specific queries are no longer required.

7.2. Data Vault Modeling

Data modeling is an essential part of data architecture that enables the description of business concepts at different levels of abstraction. Data architecture resides at the high-level architectural view. Data modeling extends this by linking the overall architecture to the specific data structures. Data modeling tells the story of data through the various layers and stages. Data modeling provides a foundation needed to govern the data interfaces.

There are many styles of data modeling, from highly abstract conceptual models to fully physical implementation models. The most well-known are entity relationship models, which are offered in a plethora of tools and still used heavily for operational modeling. For analytical solutions, which Data Vault is a type of solution on the physical level, the field is dominated by dimensional models and Data Vault models. Data Vault models are provided on a physical level for implementing a versioned, historical, resilient architecture. Dimensional models are provided on a logical level for providing a blueprint for analytical workloads. They are explained in the previous section. Data Vault models help us answer the following questions: What data needs to be collected? Where do I put it? How do I move it? What else do I need to consider? How do I monitor the environment? How do I document the environment? How do I integrate into the business? The Data Vault model is about the technique of storing your data in a point-in-time perspective.

Data modeling is the practice of designing and implementing a data model, a tool that describes the data objects, associations between these data objects, and the rules. Categorizing business and IT processes in relation to the data is essential in implementing a cost-effective data warehouse. The design process is an iterative loop between determining data requirements and representing those requirements in a data model. The final outcome of the iteration is a physical or technical model that will serve as a reference for the creation of the database.

8. Data Governance and Security

Data governance is the exercise of decision-making and authority for data-related issues and is essential to ensuring the availability, usability, integrity, and security of data in the data architecture. We consider data governance in terms of overall data quality and accuracy, with sensitivity to private information, for example in the areas of personally identifiable information and biometric information, financial services, increased operational support costs, health and safety risks, and youth protection.

As an asset of the organization, data must be created and governed according to a set of business and regulatory rules that are reflective of how the organization wants to leverage the asset. Doing so ensures that data assets are trustworthy, usable, valuable, and portable. Data governance defines and implements the data stewardship policies that preserve the data's integrity, security, and privacy, and enable effective use by business stakeholders and compliance with existing regulatory requirements. Other aspects of data governance include establishing and communicating rules for effective data management, creating a data environment in which stakeholders can report errors in data quality, establishing a metadata maintenance and publishing policy, and creating a data governance education program [7,8]. Data is also needed to build and deploy core artificial intelligence solutions, which raises an additional area of governance: bias. Those AI solutions often rely on datasets that may not be representative of the various groups of people who would use, or be affected by, the software, such as people of different ethnicities or demographics.

8.1. Data Privacy Regulations

Empowered access to abundant data is the cornerstone of a data-driven organization. It helps data science teams and operations ensure the delivery of vital data-driven experiences within the organization and for its customers without delays. However, the same abundance of data, when not properly governed, creates opportunities for malicious leaks or even ignorance-based misuse of sensitive personal data. By definition, any data that can identify or potentially identify the individual it relates to is sensitive by nature. The primary data protection regulations focus on regulating what organizations can do with the sensitive personal data they collect and process, what data to avoid or minimize as much as possible, what data retention policies to apply, as well as users' rights to access and rectify their data or to request its deletion.

The European Union adopted the regulation in 2016, and it was enforced starting from 2018. This regulation covers all data and services provided within the European Union, but also the data collected by organizations located outside Europe about European citizens. The regulation established several data collecting and storing constraints. These constraints include the use of anonymization and pseudonymization on user data, the establishment of a Data Protection Officer responsible for ensuring compliance, the need for organizations to explain why they need the data they are using, and the users' rights to be aware they have their data, access it, restrict its processing, and enforce its deletion if needed. Data processors and data controllers from outside Europe also have to establish a specific contractual agreement with a legal entity in Europe granting the data protection compliance responsibilities, as they are subject to turmoil over data sovereignty.

8.2. Access Control Mechanisms

In a Data Lake house, many roles need to access the data. Users can have one of the classic roles of a Data Warehouse: Analyst, Backend Developer, Data Engineer, Data Scientist, but also non-classic ones, like Business User, Data Directory Curator, or Data Lake operator. The data need to be stored following the Data Governance laws of the enterprise (the right data, in the right format, at the right time) and having the least sensitive data encrypted and high-consumption data easily accessible.

For authorization, we can use simple Identity and Access Management (IAM) permissions to control reading access to data, or Advanced Data Security mechanisms. The first mechanism is very basic: it relies exclusively on the IAM desktop portal to control access. In our Data Lake house, if using Data Lake Gen2, we will provide direct access to the file and folder levels, allowing Access Control Lists (ACLs) to work and possibly creating the underlying folders in the namespace. If using Data and Analytics Service, we will create a Lake House which will allow us to write data in a parquet format, converting it if necessary.

On the other hand, the Advanced Security layer allows defining, querying, and protecting the data, which allows building a business model on it. Therefore users will get access only to data that are not sensitive according to business rules or information classification rules defined in the policies. The Advanced Security layer relies on Dedicated Pools, in which we can define business models, restrictions, manage all users accessing the data, define what parts of columns and rows of a table a user or a class can manage, from which frequency these accesses should be applied, and many other things.

9. Analytics and Business Intelligence

Data is only useful when it is consumed and acted upon. It is therefore critical that the data we are moving into Microsoft Azure generates actionable insights. Power BI is by far the fastest adopting analytics solution. It has been designed for rapid development, deployment, and consumption of dashboards and reports. Many organizations already run business intelligence using Power BI. Azure also provides for advanced analytics tools.

Power BI Integration

Power BI provides a range of options to help incorporate Microsoft Azure data into Power BI dashboards and reports, and to allow Azure solutions to leverage Power BI as a native dashboard and reporting capability. Power BI has native connectors available to connect to nearly all data services within Microsoft Azure. You can push aggregated data typically from Azure Data Factory, Azure Stream Analytics, or Azure Analysis Services. Power BI APIs are available to push data to Power BI premium datasets from any service running in Microsoft Azure.

Power BI can be invoked to display dashboards from within any of the Microsoft Azure services that have the Azure Marketplace API to display a URL. Azure Synapse Analytics integrates tightly with Power BI enabling rapid exploration and transformation of data.

Azure provides for a native reporting and dashboarding capability in Azure Synapse Analytics workspaces via the integrated Power BI workspaces. Azure Data Factory enables publishing Power BI dataflows for collaborative data preparation. The data preparation is then available as a service for Azure Synapse Analytics and Azure Databricks to use. Data news can be set up using Azure Datashare, which can create a Power BI dashboard for placed data to keep the consortium of business partners up to date.

9.1. Power BI Integration

The integration of Power BI with Azure Data Architecture is a powerful solution for organizations that want to provide trusted, governed access to the data people need. Power BI allows business stakeholders to explore data freely, gaining insights and improving decision making. Built-in natural language querying features and embedded AI capabilities allow business users to analyse their data without help from IT. Data producers build datasets that bring together the relevant data from different data sources and publish the datasets to a centralized

data model in Power BI Service. A single source of the truth helps drive additional insights, reducing redundant calculations that create a burden on the underlying data sources. The popular Power BI desktop tool empowers business analysts to create compelling paginated and visual reports. Seamless collaboration is provided for business stakeholders by building insights on top of the Power BI Service capabilities.

Direct SQL Query mode allows Power BI to query data from Azure SQL Data Warehouse. Live connection allows organizations to connect to logs stored in Azure Log Analytics or Azure Monitor without ETL. The Paginated Reports feature is what most organizations will leverage at the beginning of their Power BI adoption. This feature has a service that directly connects to Azure Data Lake Storage, Azure Blob Storage, and Azure SQL Database, Azure Cosmos DB, or RESTful APIs. Azure SQL Data Warehouse provides integration with Power BI for ad-hoc querying so that business users can analyze data in the warehouse without having to rely on IT or an analyst.

9.2. Advanced Analytics with Azure

The preceding chapter showed how the majority of data analytics tasks can be covered in Azure via Power BI. Power BI empowers business stakeholders with the ability to brainstorm new ideas, and create and redesign any visualizations based on the most up-to-date data. Power BI also brings everyone in the organization on the same page via the creation of custom dashboards sharing key metrics that require constant monitoring. The key role of Power BI is to empower multiple business users while alleviating pressure from IT departments. IT departments remain responsible for making sure that the data warehouse is scaled, optimized, and secure, while the Power BI Premium version allows the organization to share and analyze data residing in the data warehouse without any limitations or restrictions.

However, advanced analytics tasks sometimes go beyond the capabilities of Power BI. For example, advanced analytics algorithms like neural networks or specialized libraries are needed to solve certain types of problems such as image or text recognition. These advanced tasks require data to be trained on using modeling tools. The trained models need to be monitored and retrained based on new incoming data. The trained models then need to be deployed either in the Azure service to be made accessible by other applications or they need to be embedded back in the data warehouse via a stored procedure to be called with each corresponding new incoming row. In this chapter, we will describe and explore these advanced tasks in Azure and related services.

Azure Analysis Services is essentially an online service running on Microsoft servers for OLAP services, but with a few differences. It allows an organization to segment users into groups with different access permissions, trade the cash flow and ownership of the engine with Microsoft instead of managing a physical engine, and leverage the capabilities of Azure by integrating it with other Azure services. Azure AS allows the organization to create datasets out of raw data from the Azure data warehouse, and expose them as tabular models.

10. Performance Optimization Techniques

Data Warehouses require cloud storage services to deliver sufficient performance, stability, and scalability throughout their lifecycles. Without these guarantees, critical analyses designed to supply meaningful metrics for measuring business performance are unusable. Common performance problems include slow query execution, queries returning nobody, concurrent connection limits applied, non-optimized CPU and memory usage, unbounded query results, and long data loads under normal load or wait requests disk.

This chapter provides a quick overview of the problems and their solutions. To start, I will cover two broad principles before launching into specific examples. First, remember that performance optimization at the warehousing level is not the same as solving performance problems at the database query level. One way to think about this distinction is the difference between a road system that breaks down under demand and an individual traffic accident that backs up rush hour traffic. It is easy to mix up the two kinds of issues because they can look the same from the perspective of business users and application developers. However, in reality, different teams are generally responsible for fixing the two classes of problems. In a service-based architecture, we may not even be able to diagnose whether we are looking at a raw resource contention problem or dealing with a problem deeper in the structure of the service chain that the warehouse is serving requests for. Therefore, it is essential to cooperate closely with the developers of the services that rely on the warehouse.

10.1. Query Optimization

Query performance optimization is the area of performance optimization which focuses on the performance of analytical queries, in particular SQL queries. The demand for high-performing analytics queries and low latencies is one of the major driving forces behind the adoption of specialized query engines, on-premises, and cloud-based analytic databases. Over the years, many techniques

and tricks have been proposed and developed by query engine and database engineers and developers to improve on the performance of analytical queries, and database turns into a production way of delivering business insights. These techniques also form the building blocks of specialized database engine architectures as well as the cloud-based data warehouse service.

The query optimizer is concerned with determining the most efficient way to execute a given query by considering the possible actions and selecting the most efficient one for execution. The most common part of a query that can be optimized is the join operation between two or more stream sources or relations, but there are hundreds of other functions and operations that can also be optimized using different techniques. Since the optimization of a database query depends on both the database engine design as well as how data is stored in the database, there are several different ways to optimize SQL query performance, starting with the simplest and most commonly used optimizations followed by engines that use caches, selectivity estimation techniques, as well as custom-built optimizers. Query tuning or optimization occurs at a variety of levels. Users of the query interface can optimize their queries by using certain sequences. To serve many users efficiently, the system can also precompile queries and store them in an executable state. The optimizer responds to any translational ad hoc queries by producing, for the first time, a query plan but can also dynamically optimize queries for performance.

10.2. Indexing Strategies

When executing a SELECT operation on a Data Warehouse table, the query optimizer will pick the most efficient data access path using available SELECT execution strategies. This is commonly done using a B+ Tree traversing the data. Therefore, the efficiency of SELECT operations is much higher when the target columns are indexed. An index is a redundant copy of a portion of the data table embedded with additional information at various levels for faster data access. An index reduces the number of I/O accesses that must occur when a query joins with other related tables.

As data in data marts starts growing, using indexes to improve query response times has been a recommendation in the industry for a long time. Data Warehousing tables – especially the dimension tables – have numerous columns, but are sparsely populated with data. Hence, indexing on such tables assists with speeding up query performance. Also, the columns with the largest variation are the best candidates for index optimization. Because of their sparsely populated nature, Data Warehouse tables need to be monitored on a timely basis to analyze, rebuild, and remove indexes as necessary to cater to changing business needs.

With the advent of modern Data Warehousing solutions, columnar storage is natively available, where the data within a column is stored in contiguous blocks. This results in both improvements in data access, query performance, and inference for analytics. Query criteria can be more restrictive using the Disk-Based Graphical Interface, and the Data Warehouse reduces the logical I/O and paging that occurs, removing unnecessary elements from the page itself.

11. Cost Management in Azure Data Solutions

Managing costs throughout the stages of your data solution lifecycle is an essential part of any project plan. Not only does the Azure Cloud have APIs to report on your actual spend, but there are also tools to estimate costs. There are several areas where proper planning can get you a working solution that is cost-effective. Beyond the infrastructure, deploying good code and using the various services correctly can play a huge part in keeping your costs down.

11.1. Cost Estimation Tools

Azure enables businesses and organizations to take advantage of the cost-saving benefits associated with the cloud. However, without the proper planning and management, Azure costs can quickly spiral out of control. One of the best ways to prepare for management of Azure costs is to develop or gain access to the tools that monitor your Azure consumption accurately and help estimate your costs. These estimates can help inform your decisions regarding proper resource allocation and billing.

Various cost estimators can help you decide what resources to leverage for your desired solution. Azure provides a pricing calculator and the Azure Pricing API, which allows you to create applications that access Azure pricing information in real time to populate your application’s user interface or allow for some decision logic. The Microsoft Cost Management Tool helps users estimate costs for Azure subscription or resource group activity. This tool, which is present in all Azure subscriptions in the Azure Portal, is available. Another way to estimate costs is to make use of the open-source Create-Azure-Resource-Estimate tool.

11.1. Cost Estimation Tools

Due to the widespread use of cloud services, many customers find that their monthly usage is higher than expected, especially during or shortly after the end of a month. There are many reasons for this, mainly related to the Underwear Rule: "At any time of day, the services are offered by Azure to the neighbors of

a sophisticated data solution that do it in an amateur way, while the data solution still partially asleep, are on the cloud; and however, is within a reasonable area of time." Therefore, it is good policy to estimate the costs of Azure services as accurately as possible before starting to create any solution. Azure offers a set of tools that can help with this task.

The first and most important of these tools is the Pricing Calculator, which allows users to estimate the cost of the services they plan to use for different periods and with different options. The calculator interfaces with most of the services in the Azure ecosystem, from those that are the basis of data solutions, such as virtual machines and Azure Synapse Analytics, to those that support them, such as Azure Bastion, Azure Monitor, or Azure Machine Learning. The Pricing Calculator allows you to set multiple configurations, so you can better understand how much uses the configured services for each configuration, which might be a key factor in the decision-making about the architecture and parameters of the services.

The Pricing Calculator is available and, although it is worth creating a free Azure account to adjust some of the parameters for the estimations, such as the region where the services would be deployed, it has plenty of options available and is completely free and unrestricted. In addition to being the best practice for making estimations before creating any Azure solution, be as accurate as possible selecting all the parameters available in the Pricing Calculator, estimations can be saved as a JSON file, loaded to review or modify them at another moment, and also shared with other users.

11.2. Best Practices for Cost Optimization

Azure offers multiple services for Data Solutions, such as Azure Data Lake, Azure Databricks, Azure Synapse Analytics, Azure Data Factory or Power BI. The pricing structure of those services may vary pretty much depending on the dependencies and combinations of those services used for the Data Solution and the use of the Data Solution throughout its lifecycle. Not only the monthly costs associated with the Data Solutions but also the consulting services used to build and deploy the Data Solutions may vary dramatically depending on the project definition and the imposed constraints. The budget can either increase dramatically over the first months of a project, depending on how much time the consultants spend building dashboards or how quickly the client is to test or approve the results, or can decrease quite a lot depending on whether a consumption-based strategy is followed, which would mean that Data Solutions are built in less than optimal technical conditions, thus being less reliable.

While building a Data Solution on Azure, there are a few recommendations that can dramatically reduce the monthly costs of those services. For instance, for Azure Data Factory, it might be the best option to reduce the number of compute hours in case there are so-called tumbling windows, in which the data is loaded from the source system once every few minutes. Another useful tip would be to schedule the Data Factory Pipeline, for instance on a daily or weekly basis, to run only overnight so that the data is not stale, but there are not too many times when the Pipeline is running. For Azure Synapse Analytics, instead of implementing a structure based on dedicated SQL pools, it might be better to go with Serverless SQL Pools, provided that the query execution time is not very big.

12. Case Studies of Azure Data Architecture

Over the last few years, we have helped customers in various sectors and businesses migrate and transform their data in Azure. With the scale and variety of projects we've done, many lessons learned have emerged along the way. In this chapter, we share our collective experience, highlighting our approaches, best practices, and recommendations. By sharing our approach and experiences, we hope to help businesses accelerate their cloud data initiatives.

11.1. Industry Applications

With a heavy focus on real-time data such as market news, press releases, data feeds, bulk ingestion, and more, financial intelligence companies must focus on usability, data quality, and a strong search/search result ranking. Also critical to their business model is having high availability and low latency. Having technology that provides global delivery on demand and minimizes the costs of very large-scale data storage and access environments is key to them.

Governments, NGOs, and research institutions rely heavily on Azure data architecture to combat illegal activities and detect and alert agencies of any possible discrepancies. Partnering with global social media platforms, this research organization collects data that can help them do predictive analysis. Using a database, they performed a country status analysis of the COVID-19 pandemic and presented a solution that continuously collects and retains social media data for a specified period.

In our experience, organizations can make the best out of their, often limited, data engineering resources by leveraging a variety of ready-to-go connector solutions generalizable for their ETL use cases and building data products both fast and

across lines of business. At the same time, organizations should invest in the internalization of data engineering competencies, especially around data quality, taking advantage of platforms.

12.1. Industry Applications

The principles outlined in this book can be applied in any industry. In fact, they have been implemented in dozens of industries. Some of these industries are discussed in the following sections. Data architecture provides solutions that virtually any business can adapt to their needs. The topics discussed cover only a fraction of the industries that have adopted technologies to build their data architecture and frameworks.

Retailers use advanced analytics to look for patterns in customer purchasing data associated with loyalty purchases — customers who use loyalty cards, either physical or digital, when they shop. This process is called customer segmentation, and it groups existing customers based on common purchasing patterns. These customer segments can then be used to predict how likely customers in each segment are to buy certain classes of products. Retailers, before or during each buying cycle, will promote certain products highly targeted at certain purchasing segments to maximize the chances of getting customers to buy those products.

Online retailers again analyze purchasing data, but this time they use transaction impulse purchases at checkout and browsing cart abandonment — leaving extra products in the cart at checkout — much more so than brick and mortar retailers. Other online retailer processes stimulated by advanced analytics include targeted advertising and customer rating of product reviews. All of these processes are designed to reduce cart abandonment and boost confirmation of impulse purchases and purchasing excitement by online customers.

12.2. Success Stories

Industry-leading enterprises in retail, telecommunications, e-commerce, and logistics depend on Microsoft's Azure Data Architecture. These enterprises have achieved incredible success by using the Microsoft data and analytics platform.

Subway, a global quick-service restaurant chain, embraces growth by performing advanced customer and store analysis across 37,000 locations from its headquarters. With Azure, Subway created a central data architecture that delivers store intelligence and supply chain analytics on demand. It uses Azure Data Lake Storage, Azure Synapse Analytics, and Power BI to simplify complex analyses, including product mix, sales productivity, customer loyalty, supply

forecasting, acceptance testing, and market basket analysis. Subway's associates spend less time building reports and validating datasets and more time telling the Subway story and taking timely action on insights.

Ecovadis is an industry leader in supply chain sustainability management solutions. Providing rating services and technology support to a growing customer base, Ecovadis amasses internal and external data related to Corporate Sustainability Assessment scores and datasets from over 100,000 companies. Utilizing an on-premises data warehouse system for its storage needs, Ecovadis struggled to cope with the sheer scale and complexity of its requirements. Reporting, extraction, and load times were high due to immense volume ingested in batch loads. Load and maintenance were sensitive to data availability due to shutdown and production locks, and performance was low on density when scrubbing and sharing the data with customers. Ecovadis turned to Microsoft Azure and used the cloud service to create a reliable, scalable, and performant data platform. The solution is built entirely on Microsoft Azure infrastructure.

13. Future Trends in Data Warehousing

As with every technological field, enterprise data warehouses have a future that is trending towards a more modern approach. Today's enterprise data warehouses have taken shape over decades, with new techniques and concepts continually improving functionality, performance, reliability, and flexibility to meet the constantly changing requirements of organizations storing and analyzing business-critical information. Overall demand from organizations is generating trends that will drive the evolution of data warehousing for years to come. As organizations rethink data in new ways, this creates the demand for their data warehouses to change.

Supporting Observed and Acted Information. To increase the reliability, accessibility, and usability of data, data warehouses are evolving to make it easier to work with previous states of data. These capabilities provide unique views of the information for various constituencies and help meet your organization needs. Data warehouses will be able to create and maintain easy-to-use, subject-oriented business terminologies. This enables business users, architects, and experienced analysts alike to find datasets and join between them for historical query processing. Data warehouses will also help answer questions fast. This requires a two-part approach: simpler, faster descriptive queries and transactions to create

controlled datasets that combine known contents into their latest states for a wide variety of consumers.

AI and Machine Learning Integration. Every type of company is exploring how AI and machine learning can make them more efficient or help them discover new sources of income. Just as current business intelligence tools let users traverse the data warehouse for insights that help answer business needs, in a semi-autonomous way, adding AI and machine learning capabilities to enterprise data warehouses will further streamline and automate tasks. As companies look to explore new possibilities, they can ask their data warehouses to add AI and ML capabilities to their toolboxes. AI and machine learning combined with enterprise data warehouses will allow them to recognize the business and economic associations that drive market choice and translate them into customer preference pools.

13.1. AI and Machine Learning Integration

Organizations are increasingly building their businesses on top of data platforms. Both Microsoft and Azure have focused efforts on integrating tools such as Azure Data Factory, Power BI, and Azure Synapse Studio. With Generative AI capturing the interest of both business and technical audiences, these tools are starting to integrate with LLMs to help dataminers and analysts. Imagine selecting a data source in Azure Data Factory, model-building with Azure Machine Learning through a chat window, or running a complete analysis in Power BI with simple chat interactions.

The integration does not need to stop there. In data science, the notion of code first is slowly being replaced by visual tools. By incorporating LLMs that can work on high-level plans described in natural language, it is possible to reach data sources, apply cleaning transformations, build models in AutoML or design time series forecasts, and produce automated analyses around predictions of interest that should be actively monitored. Automated Machine Learning makes life much easier for domain and industry experts that can extract more unique business insights due to their background and the availability of the key data. Azure Machine Learning provides AutoML and other assistance to help go from ideas to the provision of APIs ready to be consumed within applications in a more productive manner.

Where, from Azure Data Engineering Services throughout the life cycle, do the large models fit? Helping people with coding tasks that require domain knowledge is one obvious direction, enabling easier inclusion of data for processing, gathering insights through analyses, and starting AutoML or prompt

engineering. It will be exciting to see how enterprises integrate various LLM capabilities in business processes that involve data.

13.2. Serverless Architectures

There has been a significant evolution in information technology over the last decade, with the adoption of cloud technology and the switch from capital expenditure to operational expenditure. Infrastructure was virtualized, and data centers became more efficient at using expensive hardware and powering cooling. Built on that foundation, cloud computing providers built API and web-based infrastructure. The idea of server less architectures takes those highly efficient cloud computing service foundations and wraps them up in such a way that the consumers will never see a server and are charged for the actual consumption of whatever service they use.

Companies described in this chapter are providing real-time data ingestion, data movement, transformation, action, and warehousing of streaming data. They all have a differing approach on optimization and time-to-value, and the solutions impact your business in different ways in terms of cost, added sources data, latency, and timeliness of insight. Streaming analytics is not new, but it is served up as a server less architecture, so it is accessible to non-specialized developers. Service providers clearly are responding to business needs: to make quick decisions on data to increase the self-service data model of data insight, so companies can deliver the right data to their customers and customers' customers in near real-time. They also enable companies to respond to customer feedback and solve issues quickly. These encouraging requirements and the accompanying market change is that there are many more companies producing useful software and service, available as software-as-a-service.

14. Conclusion

The cloud is enabling organizations to adopt modern architectures driven by data in a cost-efficient and rapid manner. These organizations are adopting modern data architectures that ingest varying data types via a cloud-scale data lake to store all data at a low price point. They then build cloud data warehouses that integrate data from various source systems into central repositories of enterprise reporting and analytic data. Organizations are increasingly looking to ingest and store all types of enterprise data in its raw format and do analytics on the data as needed. Fully managed services in the cloud abstracts away the operational complexities so organizations can focus on building insights and driving business

decisions from data, rather than deal with the low-level data infrastructure concerns themselves.

Provides a rich set of services and tools that allow organizations to build such modern data lakes and data warehouses. Facilitates storing both structured and unstructured data types in their raw formats and makes such data accessible and available for processing by various compute engines. Organizations can build batch and real-time analytics solutions on top of these services. Provides a rich set of integrations with partner solutions as well in order to facilitate end-to-end data ingestion, preparation, and analytics solutions. This allows organizations to integrate their existing investments and adopt a modern data architecture strategy. Such a service-oriented and abstracted approach allows organizations to focus on delivering intelligence solutions in a scalable and efficient manner.

References

[1] Singu SK. Designing scalable data engineering pipelines using Azure and Databricks. ESP Journal of Engineering & Technology Advancements. 2021;1(2):176-87.

[2] Wu C, Buyya R, Ramamohanarao K. Big data analytics= machine learning+ cloud computing. arXiv preprint arXiv:1601.03115. 2016 Jan 13.

[3] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.

[4] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.

[5] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.

[6] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.

[7] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.

[8] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.

[9] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.

[10] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.

[11] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.

[12] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[13] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63)
[14] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[15] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[16] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[17] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[18] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[19] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[20] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[21] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[22] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[23] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[24] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[25] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[26] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[27] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[28] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.

[29] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[30] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[31] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[32] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[33] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[34] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[35] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[36] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[37] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.
[38] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).
[39] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[40] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[41] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[42] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[43] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.
[44] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[45] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.

[46] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.

[47] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.

[48] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 3: Data Lakehouse vs. Data Warehouse: Microsoft Fabric Approach

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction

As the amount of data that companies collect continues to grow – both in total and in diversity – it is increasingly difficult to move and transform data that is stored in multiple siloed systems for decision-making purposes. Companies aspire to democratize access to data, whether it is structured, semi-structured or unstructured, while keeping it secure and compliant with regulations.

Organizations want to retain all of their data forever for analytics, governance, compliance, and exploratory work. Yet, many organizations treat only a small fraction of their data as a corporate asset, resulting in significant missed opportunities. The corporate data asset is under-utilized because of complexity, redundancy, and inability to drill down with a variety of tools and methods. Formerly, enterprise data warehouses hosted databases specifically configured to support queries from visualization and reporting tools. Batches moved data from transactional systems into the data warehouse, after converting it into a very specific model [1-2].

However, shifting technology and business needs led to a requirement for additional capabilities. New large volume data management systems, called data lakes, were built to mitigate the restrictions of traditional data warehouses. Unlike traditional warehouses, which are viewed mainly as storage repositories for structured data, data lakes store all types of data in their native formats [3-5]. They accept any level of data quality and structuring, from organized and curated to messy and raw. They have the capacity to store consolidated hybrid data from

across multiple repositories, whether on-premises or in the cloud. Since data lakes employ inexpensive mass-storage solutions, they allow for ease of data retention, usually for much longer time periods than traditional data warehouses. There is a clear need for a design that retains the best of both worlds.

2. Overview of Data Warehousing

1. Data Warehousing: A Short History Since the dawn of computing, users have wanted to access more and more relevant data. Every organization has a natural desire to store the information it produces, whether for regulatory reasons or for analyzing how to improve business processes, become more efficient, sell more goods, generate more profit [6-8]. Companies generate data in their daily operations at a staggering rate. Consequently, we have witnessed the growth of enterprise databases over the past fifty years. Enterprise systems, and particularly relational database management systems, were developed in the 1970s. The advantages of enterprise systems have driven the growth of enterprise databases. Nevertheless, transactional systems have center stages in most organizations. They typically provide low cost of transactions, good performance on the commonly executed CRUD operations, and built-in data integrity. Databases for enterprise applications have several disadvantages. First, committed transactions are invisible to other users until it is too late to do any analytics. Because enterprise databases are all about transactions, there usually is little to no data integration. The data are designed to support transactional functions. The stored data structure is oriented around how it will be accessed in transactions. Certainly, there is normally no data warehouse star schema in an enterprise database system. All of this makes it difficult to perform the business analysis that is so vital to successful enterprises.

3. Understanding Data Lakehouses

The data lakehouse, quite simply, is a very big data that contains both live streaming transactional data, lets call it the hot data, and data that is at rest, cold data. The hot data is used for the most real-time, action driven analytics and notifications. The cold data are those data histories that are used for more complex analytical functions and algorithms for predictions [7,9-10]. The data warehouse is strictly hot data; no transactional data no cold data. The data lakehouse allows for the full value spectrum of data driven actions and insights.

Our consideration of such systems forces us to define operational data analysis functions as well as the design for query access methods employed on each data class. Picture then a system that integrates a transactional database with a data warehouse, allowing transactional operations to see near real time changes in a separate analytical database [1,11-14]. Data could be transferred from the analytical database back into shipping late sponsorship items as well as into two compute clusters: one more conventional, which is optimized toward dealing with very large query workloads, and the other having fast micro second access for more real time analytic notifications that trigger interactive actions. From a perspective of systems implementation, the main operational concern is very much the level of batch coalescing of transactional changes into such separate data repositories [13,15-17]. The objective is to have change latency as small as the operational significance of the data flows without suffering too much in the way of analytic latency, since the frequency of queries is related to the transaction intensiveness and/or what percentage of total users are being served. Because it combines both types of data storage and use, the data lakehouse is a very appealing alternative for those organizations focused on maximizing the responsiveness time between data updates and decision action consequences.

4. Key Differences Between Data Lakehouse and Data Warehouse

The distinction between the data warehouse approach and the data lakehouse approach is profound. In the modern world, in which our users demand project data in real time over tools, and in which analysts are utterly siloed from the data producers, being the data engineers, data scientists, or machine learning engineers, a different architectural approach is required. This also extends to our data producers, who wish to generate enriched data to share with BI tools without creating bottlenecks in the production data pipeline. In the traditional data warehouse, transformation primarily happens at batch frequency during an operation, and the processed data is then optimized for storage and query performance for particular dimensions on expensive proprietary storage appliances. These costs are typically projected on an enterprise's revenue, thus leading organizations to restrict the number of queries.

As a result, the transformed data is treated as a frozen view of the organization and not as insight subject to improvement and revision as new algorithms, knowledge, and technologies become available [18-20]. By creating a different

tech stack that provides a clean decoupling between the producers, the pipeline, and the analysis, and which also further utilizes enterprise resources by providing a metadata service for the same long-term storage used for the data lake, the lakehouse enables those tables holding data that change very slowly with data that is constantly in flux to be combined into a single system powering analytics, machine learning, and production data pipelines. A lakehouse optimizes for performance and costs through the “minimizing objectives” of exhausting the ensemble of algorithms that can be generated by commands as long as these utilize more readily available functions, and augmenting those programs via optimization of contents in computation and pre-preparing—building and time-parameterizing sophisticated transforms into base data.

5. Microsoft Fabric: An Introduction

This section aims to give the reader a brief sense of what Microsoft Fabric is doing. It’s not a tutorial; it’s an appetizer, just to whet your appetite for the main course, which is the Content-First Approach of Section 6. Learning about Content-First will give you a very solid understanding of the architecture and workings of Microsoft Fabric. The starting point for creating Microsoft Fabric has been all the needs and pains of business users in enterprises who, at the end of the day, want results. Such users have to face an increasingly overwhelming assortment of data analytics tools and cloud services, each having their own specialized purpose, but where for many use cases, specialized means complex, time-consuming, and inefficient. When the enterprise is so large that there are different teams designing data workloads with heterogeneous technologies and using disparate tools, the chaos and inefficiencies multiply. The business user needs a simple way of getting useful insights into all data in the enterprise, not just subsets of it, spanning all data, no matter where it is ingested or in what repository it is stored, at any point.

Microsoft Fabric implements a streamlined, end-to-end, full lifecycle platform for data analytics servicing the enterprise business user. It is unique and different from other services. The novelty of Microsoft Fabric lies in its Content-First Approach, focusing on the needs and pains of business users rather than those of data engineers and BI developers. Because of this emphasis, Microsoft Fabric minimizes the hidden complexity of the architecture and implementation details hidden behind the content. This allows users to be maximally productive in as little as possible time with the least inefficient iterative trial and error, and

provides fast time-to-value for organizations creating and consuming insights and analytics.

6. Architecture of Microsoft Fabric

Microsoft Fabric offers an integrated architecture that combines various data engineering and lakehouse concepts into a single solution for business intelligence, data engineering, and data science activities. It is a multi-workspace architecture, where each workspace can dedicate resources for specific activities. Data Engineers and Data Scientists can run their workloads using Fabric Spark in the org's Azure subscription. These resources are easy to customize, dynamic, serverless, and cost-effective. The accessible technologies in the Lakehouse and ETL patterns offer an alternative experience for existing SQL data professionals in the same workspace. These technologies help create SQL pipelines that automatically reach the lakehouse or warehouse at optimum times and costs. The built-in Lakehouse capabilities support governance, security, reliability, and multi-tenancy for workspaces with a massive amount of data. Interactive business intelligence dashboards can be used along the way. The Lab and Power BI timers offer to connect external systems and schedule analyzing tasks based on external dependencies [19,21-22]. This architecture is most suitable for integrating reliable, curated, structured datasets into analytics. Microsoft Fabric provides the definition of integrated components for Data Engineering, Real-Time Analytics, Data Warehousing, Business Analytics, and Data Science workloads. Additionally, it integrates with the Azure Data Lake and other storage systems that are usually used together with these components. The Data Factory, Power BI, and Azure Synapse services are also integrated internally. These integrations aim to change the user and administrator experiences. Users can quickly switch between experiences or do collaborative work in special scenarios. Generally, the data professionals describing the integrated components of Microsoft Fabric are lakehouse and ETL lakehouse professionals.

6.1. Components of Microsoft Fabric

The logical structure of Microsoft Fabric contains many components that implement data operations or support and guarantee the execution of these operations. Microsoft Fabric covers all data-related needs in a single product, from data ingestion to data consumption. At the same time, the architecture of its components allows using them separately to cover “more specialized” workloads [11,23-25]. For example, it is perfectly valid to use data warehouses and data lakes from different vendors, and there are connector products provided by third

parties that specialize in using data engineering components for orchestration in other products. Regardless of how you combine Microsoft Fabric components and external products, the Microsoft Fabric security infrastructure covers all data operations based on role definitions and access rights to the workspaces where those operations are carried out.

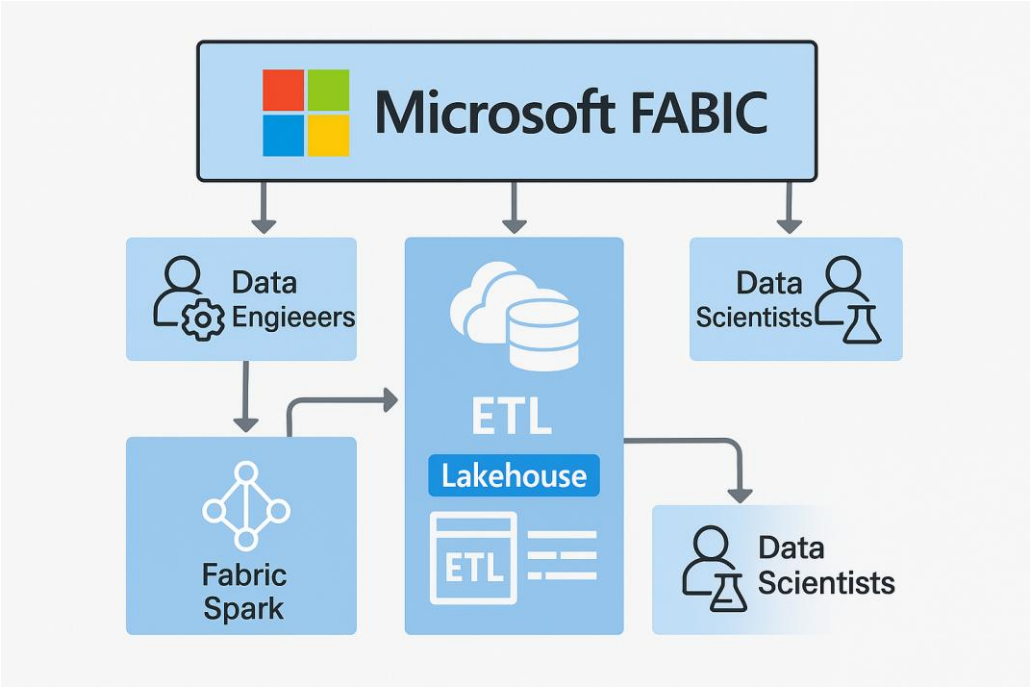


Fig 1. Microsoft Fabric

The components for data operations included in the logical structure of Microsoft Fabric are:

· Fabric Data Engineering orchestrates the execution of complex data manipulation operations across components included in Data Integration and Data Transformation, integrating them with existing external pipeline orchestration tools. It allows users to schedule the execution of pipelines and tasks, implement data quality checks, and set up notifications and alerts according to the objective needs.

· Fabric Data Integration implements the ingestion and delivery of data from and to external sources. The Microsoft Fabric ingestion capabilities are vast and support a multitude of possible external data sources, including external data stored in Software as a Service Cloud Products, as well as on-premises databases

and files. Each of those input sources can relate to the considered pipelines at scheduled time intervals.

6.2. Integration with Existing Systems

Microsoft Fabric provides a foundation for enabling organizations to build their data platform infrastructure consistently in an integrated manner - hopefully, based on the tenets outlined above. To that end, organizations want to avoid the hassle and complexity of having to integrate a hodge-podge of systems cobbled together based on internal departmental needs and budgets [26-28]. This means not reinventing the wheel wherever possible, and providing the needed bridges to allow a new system to work with existing systems until the maturity index justifies a rip-and-replace redesign.

Existing investments in traditional ETL infrastructure, on-premises Microsoft Azure SQL Server DBs, Power BI, Lakehouse and Data Warehouse capabilities from existing Cloud providers, are all examples where organizations want integration protocols to existing workloads. Fabric recognizes the important role of ETL tools in the enterprise and leverages it through the New Fabric Integration Runtime in the Factory tool, providing high-performance access to the Fabric environment. For on-prem SQL DBs, a New Data Gateway provides a bridge for self-service capability with Bi and the Business Applications - with more scenarios with all Fabric components planned for the future [29-32].

Organizations already invested in a particular Cloud vendor for existing BI, Lakehouse or Data Warehouse capabilities may also want to use Microsoft Fabric to lower costs and create a unified data management infrastructure. Support for open standards such as the Apache Iceberg format for the Lakehouse guarantees the ability to read and write from/to Fabric Lakehouses using other vendors, and allows users of those products to collaborate on the same data without data movement delays. Support for the ANSI SQL standard in the SQL Endpoint provides a standard query interface for Data Warehouses created in Microsoft Fabric. ODBC support for the SQL Endpoint makes this integration much simpler for other BI, ETL and Data Science tools.

7. Data Management in Microsoft Fabric

The Microsoft Fabric approach to data management is simple, standardized, and opinionated. Thus, users are provided with simple workflow-oriented ways to ingest, store, process, explore, clean, refine, manage, and secure their data,

opinionated methods to make good choices for their workload based on the type of data, analytics, and user need, at optimal cost, interoperability through Lakehouse storage for code and data, and a comprehensive governance, security, and management plan across all assets. Microsoft Fabric is powered by one of the largest cloud infrastructures in the world and, as such, offers unparalleled availability, elasticity, and security.

Data Ingestion Techniques Microsoft Fabric provides mechanisms for data ingestion pipelines, monitoring data quality as it flows through pipelines, and provides for versioning of the data, checkpoints, alerts, and more [31,33-35]. The primary tool for constructing ingestion pipelines is the newly introduced Data Factory within Fabric. Data Factory is a low-code bridge between your enterprise data sources and Fabric. It allows for creating pipelines that include various activities supported by the Fabric platform, including orchestration between activities, and external services such as job orchestration bindings to reports. Touching upon the data sources reachable via Data Factory, Microsoft Fabric provides connectors to over a hundred services, from Microsoft, and also to third-party and on-premises databases, file stores, analysts, and APIs.

Microsoft Fabric Data Lakehouse is constructed using a set of underlying storage techniques. First, data files can be placed and organized on Blob Storage — which treats your data as raw binary blobs, typically organized by folders. Then, to organize your data files and provide structure about the data, Microsoft Fabric provides Data Lakehouse, which allows users to build an enterprise data lake on Blob Storage using Data Lakes — containers that store your data files, enabling security, sharing, access, versioning using POSIX hierarchical namespace, logs, and browsing via Data Lake APIs and tools.

7.1. Data Ingestion Techniques

All data clearing and preparation activities happen in the Data Factory solution as each file needs to go through a coherent and consistent Data Life Cycle. The input datasets of each activity are datasets stored in the Data Lakehouse or the Data Warehouse, which is governed and with metadata created by the corresponding components.

Efficient data ingestion into its solutions from a set of connectors with numerous configuration capabilities depending on which Pattern you are executing, with the help of pipelines built using orchestration or with Dataflows [36-38]. For batch ingestion, and depending on the Connected System, it can schedule ingestion from minutes to years frequency, while for real time ingestion it ingests messages frequently. Yet, because all pipelines create and store data in Data Lake

Gen2 folders outside of the Lakehouse and Warehouse, the Data Engineering and Management team has the responsibility to manage the Data Life Cycle and create semantically meaningful datasets.

For operational data, the most commonly used Pattern, you can easily connect to your Engine API and ingest data to the Lakehouse to provide a base for all Computed Imports. For Raw Data, the Patterns relied on for the ingestion are: Data Flow Template and Data Flow Activity Pipeline.

You can also ingest data from popular O365 / Azure Data Retail Services with the dedicated and Managed Connectors for SharePoint Lists, Exchange, OneDrive, PowerBI, Azure Monitor, Security, Storage and AD, which copy data to Lakehouse locations.

7.2. Data Storage Solutions

Storage is important to shaping a data management strategy. Data stored in Fabric can be an important consideration for how versatile Fabric is as an all-in-one data platform. Data stored in Fabric can take advantage of the many capabilities Fabric offers. Storage solutions offered by Fabric are one of the main distinctions of Fabric, as you can do everything from run spark notebooks on data stored in an account, to run reports on data housed in a dataset, to allow pipelines to take data in and out of Fabric through the data factories queues [1,39-41]. Below is a preview of major options of data storage offered by Fabric.

Data Warehouse – Synapse Data Warehouse Gen1 or Gen2. A Synapse Data Warehouse's storage is based on the Snowflake Architecture, so that compute and storage are completely separated. This means that query performance isn't impaired when many other queries are being performed.

Business Intelligence – Dataset. A dataset's storage model, which is similar to Data Warehouses, allows advanced calculations, query optimizations with the columnar storage format, and integrated caching for fast report visuals.

Data Lake – Account. Data Lake Storage Gen2.

AI Data Warehouse – OneLake (Best for AI Workloads). OneLake is very similar to a Delta Lakehouse but is a lot cheaper. A OneLake zone can do things that are traditionally done by a Data Lakehouse, but make it easier for people who are developing and utilizing models in Fabric.

8. Analytics Capabilities of Microsoft Fabric

While the traditional role of data warehouses has been one of storages optimized for analytics, with Microsoft Fabric you can run analytics on stored data at low costs and with optimized performance for the coming and going of big workloads. But, additional to that Microsoft Fabric also supports real-time analytics on a separate lake destination for low-latency reading of the most recent events. You can run jobs that with a low frequency ingest data from event queues, enrich that data, and write them into a lake destination. From there, other services process that data at low-latency and your users get the information almost instantly [42-44]. This pattern is known as a Lambda architecture and combines batch processing with interactive analytics.

Real-Time Analytics

Microsoft Fabric also allows you to run streaming ingestion jobs that run continuously at low latency. Once events arrive in the event queue they are immediately picked up, processed, and made available in the Delta table by the lake analytics system. This allows near real-time access of the streaming data. Dataflow takes care of this pipe between source event queue and the lake's Delta table. Transformations can be single 'copy' operations, or enterprise transit jobs where you do lookups from a sat table, or enrich the jobs with AI. And finally, we have what we call a streaming table. You can treat Delta tables as source tables for your streaming, as well as planned batch jobs. This is all Seamless Integration makes all of this architecture easy, high performance, and low-cost.

Batch Processing

On the other hand, traditional analytics have been one of batch processing. Sure you can do analytics on very big datasets – mass volumes of information. And combine them with differing degrees of freshness, special for cold data that doesn't need to be made available in minutes.

8.1. Real-Time Analytics

The excerpt presents two ways in which real-time analytics is possible in the Microsoft Fabric framework. Both on a logical level and a physical level, we can spot some key differences between the architectural approaches of a data warehouse and a data lakehouse. It is interesting to see how both strategies get adopted in Microsoft Fabric and especially the differences from the established approach.

Near Real-Time Data Loading Using Event-based Orchestration: Users or third-party processes produce events on a frequent basis. Each event represents a small change to the underlying data [45-46]. After a while, these events related to a given subject accumulate and have to be processed to update the data at the warehouse or lakehouse. In a classical data warehouse (or lakehouse) system, this execute-and-forget batch process is typically invoked every period of time without any user control. And, since traditional ETL tools originated in data warehouse land, there are various implementations of this strategy.

Real-Time Data Processing Using Streaming State Management: We have seen how batch processing can be used to deal with smaller and larger arrivals of data requiring near-RT or RT updates of the underlying data in a data warehouse. The second fundamental way of enabling RT processing is through streaming. In a nutshell, the information at data repositories gets updated immediately after new events arrive to be stored into event storage.

8.2. Batch Processing

Microsoft Fabric's batch processing capability contains advanced analytics features that were originally introduced as part of Azure Synapse Analytics and later integrated into Azure Data Factory, Power BI Dataflows, and Azure Stream Analytics. These capabilities have proven to be quite popular with customers because they help organizations build modern data pipelines and data warehouses at scale without hassles like dealing with infrastructure and capacity expected from traditional data stores.

Fabric's batch processing features span pipeline orchestration, ETL at scale, code-free data preparation, semantic transformations, data quality and governance, and over a dozen of pre-built connectors to adopt most common data sources for ingestion. These features are built right into the Data Factory and Data Engineering experiences of Fabric, available via a single-click launch from Fabric Home. The Data Factory and Data Engineering experiences support a common development experience from the Azure portal, along with support for additional developer tools.

As a cloud-native service, Fabric's batch processing can handle any scale of data in modern analytics workloads. A few features, like the connected experiences from the Azure portal, use Serverless Data Pipeline under the hood, which only charges customers for the compute used, scaling down to zero when idle. A few other features – like Dataflows in Power BI for data prep, and ETL Pipelines in Data Factory for data ingestion and transformation – support customers' more

complex workloads with generative AI capabilities and enterprise-grade management features for easy adoption throughout the enterprise.

9. Use Cases for Data Lakehouse in Microsoft Fabric

Once data is unified in a data lakehouse, it is easy to access it for a variety of analytical and data science workloads. In this section, we lay out several use cases for interacting with normalized data in a lakehouse. Although the ease of use, commonality of purpose, and availability of reusable objects combines to make these two experiences the most natural way to work with data in a lakehouse.

Business Intelligence The classic use case for data warehouses is, and will continue to be, business intelligence (BI). Various metrics are computed, often using data marts, that enable organizations to understand and make decisions about their business. Organizations create dashboards for various roles, such as product managers, operations analysts, and C-levels, to keep track of those metrics and alert them if something changes or is out of bounds. BI requests generate access patterns that are quite similar: every day, organizations refresh their data, often in off-peak hours, then daily and weekly, and sometimes even hourly, users interact with these dashboards that originate from the data warehouse.

Machine Learning Applications As organizations grow, and by nature of being digital, they collect more and more data, and this data can be used to create predictive models that optimize their workflow, deliver personalized experiences, and ensure compliance. Data that is fresh, complete, and works in pull-based architectures becomes important for the development of machine learning pipelines. Data scientists are building models, often serving them on microservices, which become part of the pulse of their organizations. After these models are deployed in production, organizations are tracking their performance using observability metrics that need to be monitored.

9.1. Business Intelligence

Although numerous avenues exist, the business intelligence space has recently leveraged Data Lakehouse as the foundation for its most significant use case. Data Lakehouse integrates Power BI quite easily, so that teams can automate everything from data ingestion, preparation and modeling to publishing visualizations for easy access to all organization personnel. In particular, expect

to see teams using Data Flow and OneLake to store modularized ELT pipelines as Power BI Data Flow templates and the companion OneLake hub for storing the underlying modularized data in the form of Parquet files, to drive models with 100% automated refresh as Power BI Data Sets. They will then publish the Data Sets to Power BI Workspaces, generating the necessary infrastructure and access policies tied to Data Lake Storage.

Data analysts will then easily connect to the Power BI Data Sets to build scalable dashboard interfaces for Business Users, who will leverage self-service Power BI solutions for these analyses. This model ensures that business users do not need to be heavy users of the Data Lakehouse. Instead, they can leverage simple, intuitive Power BI dashboard interfaces geared for their level of analysis. Prior to the Data Lakehouse, this model forced companies to build data marts in the form of Data Warehouses associated with the Enterprise Semantic Model to avoid blowing up Power BI's in-memory BI engine for excessive amounts of historical Power BI data. The goal is to destaff data mart use cases by building Power BI's lowest level models directly on top of the Data Lakehouse [7,9,10]. However, many enterprises may not fully embrace this model until they feel more confident about the Data Lakehouse's performance and ease of management, especially with regard to multiple BI tool partnerships.

9.2. Machine Learning Applications

Machine learning is one of the most advanced analytical applications that help organizations find and exploit hidden insights from their data. Simplified, machine learning is about building models that help with predictions or classifications. Such a model learns from training data that consists of examples of predictions or classifications that are already known. Predictions can be, for example, the probability that a customer will stop buying a company’s products in the next month or the time it will take for a package shipped to arrive at its destination. Classifications can be, for example, the assessment of a loan application based on a set of financial variables (approved or disapproved) or the image recognition of an animal (cat, dog, or none). The main difference from other types of data analytical applications is that model training requires large volumes of data to reach a sufficient level of prediction or classification accuracy.

The data ecosystem provides many options for machine learning, from data integration and preparation to feature engineering, model training, validation and deployment, and finally to inference and monitoring, all within a single unified platform and environment. Specifically, machine learning algorithms use capabilities from various pillars. Traditional data storage options, including traditional data warehouses and data lakes, are part of the ecosystem available

across these machine learning steps. The difference with the data lakehouse supported by the data warehouse and data lake pillars is the consolidated data ingestion into a unified model catalog, building trust, poverty, and governance of semantic models used across all analytical workloads. These aspects are essential for machine learning, augmenting the productivity of data scientists and analysts.

10. Advantages of Using Microsoft Fabric

In this section, we describe the advantages of using Microsoft Fabric to create a data lakehouse that can easily handle all the described workloads. We will focus on the advantages of ease of use and cost efficiency. We will then describe two additional advantages: those are scalability and the data engineering provided by the Data Factory and Synapse capabilities.

Microsoft Fabric provides Spark Engines with an intuitive experience with automated orchestration that automatically selects the proper number of workers based on smart detection of job resource requirements and work completion. The orchestration feature preserves the cost efficiency of a serverless offering while optimizing job execution time and avoiding resource contention that is present when running several jobs at the same time in the same Spark pool. The data processing and loading costs are also very low compared to other offerings: this is the result of a highly competitive pricing, a seamless integration with the capability of using files as a source and destination for Spark jobs, and the fact that Data Lake Storage acts as a staging area for all files: Ingesting files to a Data Lake from Blob Storage, and accessing them from Spark jobs is very cheap.

Another nice aspect of the Microsoft Fabric architecture is that all the components are heavily tied to one another, and they consume the same Data Lake Storage account. This frees the customer from heavy integration efforts and, more importantly, it greatly reduces the data movement and serialization cost that happens when moving data between a storage account and the other services offered by most Cloud Data Lakehouse offerings. By using a consistent data structure, all Microservices Architecture design concerns like redundancy checks and versioning are made transparent for the customer. All the components assume the same access policies for the instances of the Data Lake Storage used, and a version of the contents (lots of which are cached in memory) is stored as data in the Data Lake.

10.1. Scalability

From a purely technical perspective, "scale" can mean at least two things—hard limits on the amount of data and the number of end users, and a more flexible and economical way of scaling data operations. On the hard limit aspect of scalability, we can compare the lakehouse to a traditional data warehouse, which has more strict limits on what can be accomplished at data scale. Even though or perhaps because a Delta Lake-based lakehouse uses open standards and popular open-source components rather than proprietary closed-source technologies, the practical limits on scale can be very high. You can even grow (technically) "against the wall" using a Delta Lake lakehouse, which is mostly what cloud vendors of data warehouses offer their customers. In other words, instead of actually scaling up a physical data warehouse, you can basically run different physical data warehouses for different workloads but in a way that makes them work together.

Specialized computing engines can handle different kinds of data workloads, and at a lower cost than traditional physical data warehouses, because the cost of cloud computing resources can be varied by taking advantage of the lower cost of data object storage without restrictive data schemas. As far as scaling economically, this can be a very important aspect of running high-volume data processing in more of an operational mode rather than just low-volume batch jobs that traditional ETL and data warehouse stacks in the past have emphasized. Data is constantly collected from data producers, and analysis results pushed to data consumers, who expect high volume but "on-demand" results. Companies are market players interested in specialized cloud-based solutions to operate on that dynamic workloads to deliver better results than a more traditional data stack.

10.2. Cost Efficiency

Microsoft Fabric storage resources are internally shared, so customers can take advantage of economies of scale—reducing costs. By using Fabric warehouse or lakehouse storage, customers can avoid an order of magnitude markup on the prices for using external individual dedicated storage services. In addition, customers do not have to transfer data between different systems and pay those data transfer charges. Fabric users can further reduce cost by employing storage tiers appropriate for the types of workloads they have. With the lakehouse storage, customers can save costs by using storage tiers appropriate to the temperature of the data. For the warehouse storage, when customers pause the warehouse, data is the cool storage tier, and for the data is not used frequently they can set up the automated SQL processing routine to move such data to the archive storage tier, which costs even less. Additionally, optimizing payloads

provides incremental cost savings and performance scaling benefits. In the case of warehouses, optimal sizing of the data units for typical query processing costs and performance is automated, but users have the option to decide on the number of DWUs to use for each warehouse. In the case of lakehouses, formulas have to manage the optimization functions, otherwise, users have to optimize the settings themselves, which is not so uncommon with other lakehouses—this could lead to incorrect costs or inefficient storage performance.

11. Challenges and Limitations

Although the data lakehouse architecture addresses some of the more serious limitations of traditional data processing and performance management systems, there are clear challenges for the lakehouse that must be addressed in future iterations of this technology. Organizations with significant investments in existing data technologies will rightly question the capability, flexibility, and cost issues associated with introducing a data lakehouse platform into their environment. Some of these concerns can be addressed in the short term, while others will likely require additional, near- and mid-term enhancements. Data lakehouse platforms make a lot of assumptions about how multidimensional and other structured data is logically stored and accessed to provide optimal performance, resource management, governance, and security. With unlimited storage and a need for collaborative tandem use of data, it will take time for IT and data governance organizations to define and implement necessary controls and policies to govern and secure what could be a wild frontier of data science and analytics. In many cases, organizations will want to have very different security practices between data used by casual experts and artisans and data used for serious business intelligence and operational management. While there are a lot of storage and management details that need to be worked out more generally across the Hadoop and object storage areas. The performance of a data lakehouse platform is much better than a traditional data lake or a set of disjoint data services, but it does not yet equal the optimized performance of cloud data warehouses or massively parallel scaling of traditional data warehouses. For a variety of reasons, query times, especially for ad hoc queries using co-located or lower volumes of data, can be slower in a lakehouse than a cloud data warehouse, and performance can lack predictability. For example, the performance of a traditional data warehouse during a busy period might be bad but not terrible, but performance issues for even a low-volume data service in a lakehouse

environment, say retrieving data from a shared location, could be so bad that it would be unacceptable for operational use.

11.1. Data Governance Issues

Data governance established in Data Warehousing has to migrate intact to Data Lakehouse Systems that are used in enterprise environments to be compliant with data privacy, data security and data protection regulations; and to respect business policies for data access, as well as satisfy common business requirements for Corporate Business Intelligence. Because for these a controlled environment of data content, structure and coding to avoid inconsistencies in reports is mandatory. To visualize this, consider the example of a shoe store that offers men's shoes and ladies' shoes in the same data model tables. The name of the table is "ShoeStore" but the description of the data in the table says, "Ladies' Shoes Store", when both types of shoes actually are in the table. All shoes are expressed in skillions and the computing commission of the shoe store is e.g. in US dollars; or all sales in Germany are expressed in Euro and otherwise in USD. Such inconsistencies are a nightmare in reporting and these inconsistencies need to be governed. Reports have to be able to rely on well-governed metadata; any changes in the Data Model have to be communicated actively to Discovery Tools and Reporting Tools otherwise the development of wrong reports is the result and such tools either have to be retired, strongly controlled for change or governed intact as part of a more formal Data Warehouse environment.

However, in a Data Lakehouse architecture these Governance Problems become a challenge again as for Data Pipeline Tools or Non-SQL based systems it is usually impossible to build a Business Metadata Layer on top or even beside these systems like Business Intelligence Tools can build on top of Data Warehouse Systems: To enable an easy sharing of tables to Business Users for reporting of self-service tools, Business Information Models need to be developed; Business driven definitions of structured data need to be stored in a Business Metadata Repository and Business User access to any kind of data has to be controlled.

11.2. Performance Limitations

Despite the strong service offering Microsoft Fabric provides for data lakehouses, there are still performance limitations that lakehouses struggle with when compared to their strictly relational counterparts. Data lakehouses build on the concepts of data lakes but often include some additional formatting for optimization purposes. These optimization features can include the compute optimizations of data skipping frame index, caching, distributed and scalable

compute for query processing, partitioning, and sorted files. In simplistic terms, these optimizations add an array of indexing features that allow lakehouses to be competitive with data warehouses while maintaining relative data lake logic and structure. Despite these optimizations, data lakehouses still struggle with the performance management of data warehouses with relational queries. For companies that prioritize performance for interactions with and within their data, relational data warehouses may still meet these company's analytic needs better.

The optimizations many lakehouses provide can help speed up interactions with the data compared to a typical data lake using Parquet files, but still are slower than the interaction options found within dedicated data warehouses. For example, while banks or financial institutions would be very suited for a mixture of both lakehouses and warehouses, retail companies that do mass operations with complex joins that require user interactivity in real time would best be suited for a dedicated data warehouse. When companies and various industry sectors are having to make the decision to build a lakehouse or data warehouse, it is usually because they are leaning on one of these two options primarily for data and query management. Data ingestion, ETL, and low-cost storage are what lakehouses cater best for in comparison with warehouses.

12. Comparative Analysis: Data Lakehouse vs. Data Warehouse

This chapter performs a comparative analysis to help assess the relative merits of each and enable enterprises make the right choice. While comparison between a data warehouse and a data lakehouse is pertinent, it is difficult to make the comparison because either of them does not exist in isolation. Instead, both of them are at some level an extension of an existing data management architecture. Some existing data management platforms have either tried to extend the capabilities of data lakehouses to become data warehouses or created data lakehouses by wrapping their data warehouse services so that they appear or function like a data lakehouse. Nevertheless, it is still useful to compare a conceptual data lakehouse and a data warehouse to understand the relative positioning of either of them. The rest of this chapter describes the representative features in more detail.

This chapter also provides details on relative performance both in terms of cost and technical performance like query performance and ingestion speed respectively. A level playing ground is created using the following process. Both

services are deployed in the same regions (and the closest infrastructure). Both implementations make extensive use of the native capabilities of either service whenever relevant or needed. Tuning was also performed both on data models and implementation to make sure that the differences are not generalized for all workloads, but rather are relative for the specific tested workloads used in this case.

12.1. Performance Metrics

In this section, we discuss different performance metrics of Microsoft Fabric. Typically, the performance of a system is considered superior from a user's perspective when the end results are delivered quickly, and it is considered superior from the provider's perspective when the overall cost of operation is low. Delays in response times or the number of failures can provide motivation for implementing a data lakehouse solution, as they can lead to either lost revenue or increased costs. Although from a user's perspective, only the end-to-end processes matter, from a provider's perspective, different entities such as data ingestion, availability and reliability should be optimized. Since the architecture of a traditional data warehouse is not as flexible and extensible as that of a data lakehouse, it can also lead to several issues pertaining to performance as described below.

Typically, the storage costs for a data warehouse can be extremely high due to the constraints relating to how data is stored. Data is usually denormalized so that various business dimensions are included in a fact table, leading to data redundancy. As compared to a data lakehouse where data can be stored in a flat format without applying ingest-time schema where the user of a dataset needs to specify what attribute types to expect, the cost of operating a warehouse is higher. Additionally, data warehouses may require ETL processing for all data to be available for analysis. This can increase load times and lengthen the time between data creation and availability.

12.2. Cost Analysis

The motivation behind creating a data lakehouse is to reduce the cost of moving data around. With a data lakehouse, data is stored once in a single data platform while still allowing different personas and applications to run different workloads on the data, while managing cost, performance, and security tradeoffs. Moving data and copying data between platforms can be expensive and is a common pain point of customers. The costs and technical implications of having to copy complex operational data models from transactional databases into a data

warehouse, which are normally seen as “system of records”, is a taste for most customers.

Regarding data warehouses – data movement costs. Data warehouses require a significant amount of data moving to do the kinds of things that they do best. For example, OLAP queries on structured data. These are the most advanced analytical workloads. A subset of users needs to build and maintain these advanced analytical models with both strict latency requirements and cost expectations. Once built, those models see high usage so don’t need constant tune-up, but use a significant amount of resources. Data warehouse users are able and willing to pay strictly per session run. Paid session runs can be charged, and data warehouse providers can understand the margins of this workload and price accordingly, including subscription discounting for high-volume customers.

On the data lakehouse, OLAP queries are indeed a significant portion of the portfolio of workloads. However, the access is not restricted from being ask-and-answer on a schedule – which is what the data warehouse design target. Important portions of the OLAP dollar spend do not need strict SLA-type guarantees. While data lakehouse providers also create structures like materialized views to accelerate data loading, they are also being built for the ask-and-answer type workloads. For the highly complex customer workloads that run only sessions with tight SLA-type guarantees, data lakehouse providers also support the execution of those workloads through the batch server resource models within data lakehouses – which is just like data warehouses.

13. Future Trends in Data Management

The technology landscape is in constant evolution, and many predict significant change over the next decade, including Data Management. Changes in Data Management have been happening rapidly in the past five years, with increased adoption of cloud, increased use of Generative AI, and natural language interfaces being added for many existing Data Management tasks. With the strong movement from on-Premises infrastructures to the clouds and the emergence of new types of Databases, Data Lakes, and Data Lakehouses in the cloud, there are many questions about the Future of Data Management.

The majority of the times spent on any Data Management project is around the creation of Data Pipelines, which moves data from multiple disparate sources into a platform where it can be analyzed and queried. Given the Calculate Engine that companies are building, enabling to run similar types of Calculations for a

lot less, it’s likely that the way we build our Data Pipelines will be increasingly unique for any given major corporation at scale. Businesses will need to start specializing in building Data Pipelines for the relevant sectors instead of every corporation working on them singularly, and that could mean a significant increase in Sector-Specific Data Pipeline consulting companies. More specialized Data Platforms designed to work for single verticals.

13.1. Emerging Technologies

The interests that drive new data management technology development include both old and new: the maturation of the non-relational database market, the drive to make relational and non-relational database technology more inter-operable, extreme scale, and the ever-accelerating descent of the cost of storage and compute power. Emerging at or near the confluence of these seemingly conflicting interests is the convergence of relational and non-relational database characteristics [18,47-49]. This convergence will take either one or both of two paths: the hybrid database - a single database system that supports both relational and non-relational data types and access methods, or the database ecosystem - a collection of highly standardized and inter-operable database systems, or services, that use each system's particular strength to support integrated organizational data management solutions. New, disruptive scalability and deployment environments like cloud computing and no-SQL interests are the propellants of this convergence.

At the same time, there are some deeply rooted interests in data management technology that remain constant. Organizations have invested heavily in legacy data systems that are designed to support applications at the volume, variety, and velocity levels that characterized those applications at the time they were implemented. Most of these legacy system designs are at the limit of their capacity to support the needs of current applications. Many of those who own, operate, or rely on these legacy systems are reluctant to invest heavily in new systems that will simply continue to allow the organization to operate at status quo. There are many more people who want to capitalize on new growth opportunities enabled by rapid application development and the use of operational data for business decision-making. These organizations want to migrate to a new architecture that lowers the cost of change, contributes to faster application deployment, and adapts to changing operational needs of the business more easily.

13.2. Market Predictions

Being that the need of enterprises to build a unified and simple architecture leveraging both data warehouses and data lakes has existed for a while, an indication of data management vendors keeping an eye on the market to provide the right Data Lakehouse capability is the increase in relations and partnerships among this kind of solution within Data Lakehouses. Different enterprises have different architecture and data management characteristics, which correspond to different needs to combine these two environments. As such, solutions can be niche, targeting those enterprises that wish to have some integration generated by partners or for whom the integration isn't a concern, and at the other end of the spectrum, we have vendors positioning to provide a unified Data Lakehouse architecture.

A next-generation data warehouse is one that converges the capabilities of data warehouses and data lakes, but it is more than a converged architecture. This next-gen warehouse will be better in each respective area. It will be a single source for secure, governed data, yet it'll still be able to work efficiently with raw, unstructured data. It will have the BI and data science capabilities of the best data warehouses, yet also support modern data engineering needs. It will cover all parts of the analytical lifecycle better than either can do today. The nearest analysts to Data Lakehouses, with influence in the mainstream enterprise analytics arena, are the few vertical data engineering consulting firms specializing in Big Data & Data Engineering technologies, which began weighing on the balance between the data lake and data warehouse on the past few years, looking for trade-offs for respective solutions use cases, for different enterprise profiles.

14. Conclusion

Throughout their decades-long history, organizations have utilized a variety of approaches to gain insights and derive business value from the data they source. As the digital world has grown in size and complexity, tools and systems have evolved, emerged, matured, and optimized that cater to the increasing demands for analytical reasoning. At the core of both historical and emerging solutions is the central concept of a data repository. These repositories, colloquially referred to as data lakes, data warehouses, and suites of both, are the engines behind business insights. Organizations drive everything, from day-to-day operational decisions to long-term corporate strategies, from data available in those repositories. Those repositories and the systems paired to them represent a

significant portion of a company's overall data strategy, financial investment, and influence on the organization's success. Overall, the analytics ecosystem consists of many tools and technologies that serve its purpose well, each with its own strengths and weaknesses.

This paper focused on two key concepts of data repositories—the data lakehouse and the data warehouse—and approached the comparisons through a specific perspective. Through that perspective, we analyzed the history, evolution, purpose, use cases, and customer sentiments of analytical repositories and presented their similarities, differences, and tradeoffs in an impartial manner. The data strategies of organizations across verticals and domains share many similarities, such as an overwhelming volume of diverse data—in increasing amounts of both internal and external sources—in addition to the increasing demand for access and insight across the user base. As such, cloud-based, cost-effective solutions that offer flexibility, scalability, and accessibility across the business fit the need across organizational sub-teams, both business- and IT-facing.

References

[1] Kane M, Emerson JW, Weston S. Scalable strategies for computing with massive data. Journal of Statistical Software. 2013 Nov 20;55:1-9.

[2] Belcastro L, Marozzo F, Talia D, Trunfio P. Big data analysis on clouds. InHandbook of big data technologies 2017 Feb 26 (pp. 101-142). Cham: Springer International Publishing.

[3] Torabzadehkashi M, Rezaei S, Heydarigorji A, Bobarshad H, Alves V, Bagherzadeh N. Catalina: In-storage processing acceleration for scalable big data analytics. In2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP) 2019 Feb 13 (pp. 430-437). IEEE.

[4] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.

[5] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.

[6] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.

[7] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.

[8] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.

[9] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.

[10] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.
[11] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[12] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[13] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[14] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[15] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[16] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[17] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.
[18] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[19] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.
[20] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.
[21] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.
[22] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[23] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[24] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[25] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.

[26] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[27] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[28] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[29] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[30] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[31] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[32] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[33] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[34] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[35] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[36] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[37] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.
[38] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[39] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[40] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[41] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[42] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.

[43] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.

[44] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.

[45] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.

[46] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.

[47] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).

[48] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.

[49] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 4: Managing Data Ingestion and Storage in the Azure Cloud Ecosystem

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction to Data Ingestion

Data ingestion is a crucial step in data management where data is imported, transferred, loaded, or processed into a central database or repository from various sources. These data sources can include operational databases, mobile devices, servers, sensors, web applications, and third-party sources. The data transferred can be structured, semi-structured, or unstructured data [1-3]. The processes that support data ingestion include collecting, scraping, loading, importing, and processing.

Data ingestion can be of different types including offline and online ingestion. Ingestion refers to the transfer of data into a data warehouse. A data warehouse is defined as a central repository of data that can be analyzed to extract information for making business decisions [2,4,5]. Offline data ingestion is executed at intervals or on a schedule to extract large batches of data from multiple sources. Batch data ingestion transfers data at a specified frequency or interval. The typical use case for this type of ingestion is aggregating data from sources where real-time ingestion is not critical. Examples of data sources that may use offline data ingestion include transactional databases or a flat file of historical records.

Online data ingestion generates a stream of continuous data to fully process and transfer data. It is a real-time data extraction method that transfers data continuously and is processed as it ingests the data [6-8]. Any amount of data can be transferred. Online data ingestion is available for scenarios requiring near-

real-time storage and analysis. Data source types where online ingestion may be applicable include web applications, sensors, social media feeds, and news feeds. These types of ingestion transfer data at a very high frequency with low latency for processing.

2. Overview of Azure Data Factory

Data Factory is a cloud-based data integration service that enables the creation of data-driven workflows for orchestrating and automating data movement and data transformation. With Data Factory, users can create and schedule data-driven workflows, called pipelines. Pipelines can ingest data from disparate data stores. Then with the help of external compute services, Data Factory can transform the data by using its data flow feature. Finally, pipelines can publish output data to data stores.

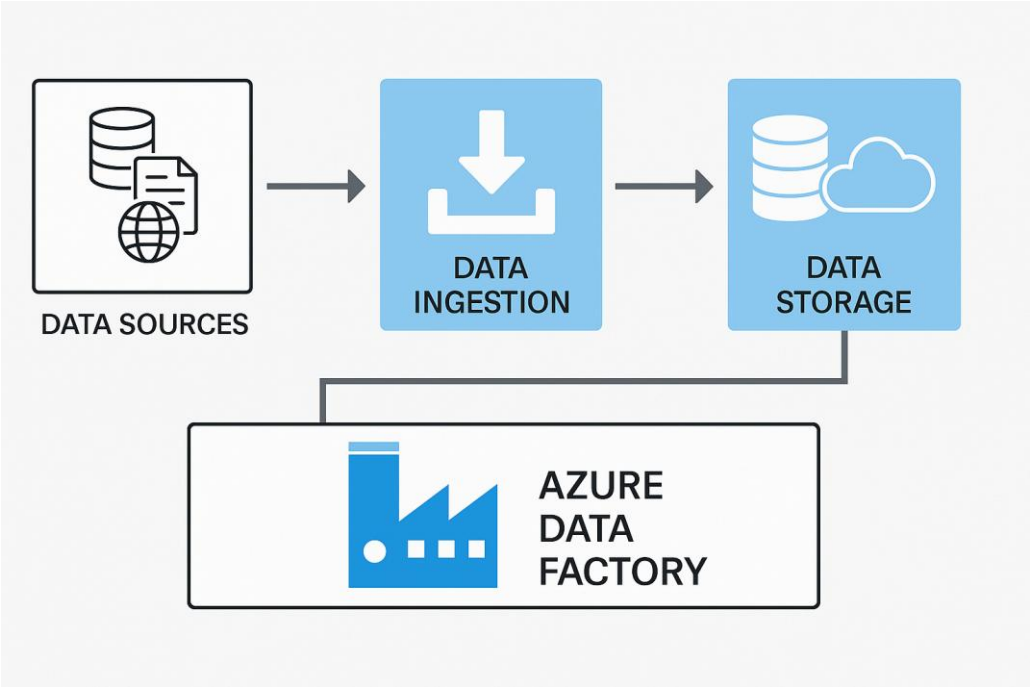
Data Factory's data-driven workflows orchestrate and automate data movement and data transformation. Pipeline activities can ingest data from disparate data stores and then transform the data by using the Data Factory data-flow feature or by using other compute services. After that, pipelines can publish the output data to data stores.

Data Factory allows you to create and schedule data-driven workflows called pipelines. These pipelines can ingest data from disparate data stores, transform the data by using Data Factory's data-flow feature, and publish result data back to data stores. Azure Data Factory can be thought of as a back-end service that lets you create data-driven workflows called pipelines that can ingest and transform data at scale across a variety of data stores. Once you've created your pipelines, you can schedule and manage them through the Data Factory user interface or programmatically.

3. Ingesting Data with Azure Data Factory

Azure Data Factory enables organizations to create data intake pipelines that can run on schedules or be triggered by events. These pipelines can execute several tasks in parallel to load and transform data from multiple sources, including transaction-based systems and web services. Data Factory provides connectors to hundreds of data sources. By employing these built-in connectors, organizations can address a variety of ingestion patterns and scenarios to populate Azure-hosted

data stores. Organizations also use Data Factory to create one-time or repeating data loading tasks.



Azure Data Factory lets organizations build data processing pipelines in a web-based UI with drag-and-drop simplicity [9,10]. The templates provided in Data Factory for ingesting from key services help organizations jumpstart development efforts. Data Factory, however, is much more than a simple point-to-point extractor or tool that uses connectors or custom code to transfer data between Azure-hosted data services and third-party services. Unlike straight connectors, Data Factory instead breaks down data transfer tasks into several steps. Organizations can define which steps to run in parallel, monitor task sequences, and track successful completion for large transfer jobs that run on defined schedules or that are triggered by events [11-13].

After establishing connections to the data source and destination, Data Factory can employ either a Copy Data task or a pipeline consisting of several data processing and movement tasks to perform the task. The Copy task is a specialized data movement agent that can transfer data across various formats and services, while data processing tasks like transforming the data, reshaping it, or moving it across various Azure services are more general-purpose in nature.

3.1. Creating Data Pipelines

Data is everywhere and comes in many different varieties and types. Businesses benefit from the availability of data by performing analysis over that data to derive insights and perform data-driven decision-making [2,14-17]. However, often the data is not available in the right place, at the right time, in the right format, and in the right shape. There is often effort required to find the data in different disparate locations, extract the data, and do data transformation and data cleanup before that data becomes usable. Data Factory lets you create data pipelines to orchestrate these activities for you. Let's explore data ingestion and pipeline creation steps in detail in the section.

In the previous chapter we took a deep dive into the key concepts, benefits, and components, while understanding the purpose within the data ecosystem. In this chapter, we will walk through pipeline creation and triggering steps to help get you started with either a provisioned instance or create one for you. Pipelines help you to build your data ingestion and orchestration within seconds and minutes instead of days or weeks [9,18-21]. In the following example, we perform the following steps leveraging Data Factory to build the ingestion fully encrypted solution and run: Create the resources; Build the pipeline; Trigger the pipeline to run; Monitor data pipeline runs. The data factory internally will manage and monitor the trigger and various integration runtime servers and notify you on completion error and success. Data Factory can be a critical process of your data DevOps. You can build the ADF by writing and deploying it to using DevOps.

3.2. Data Transformation Techniques

Data not only becomes available over time but also becomes available in different schemas, formats, and data types. To make this data more usable for analysis and decision making, data transformation is required. This requires not only conversions but also various join operations, conditional datasets, palette datasets for noise filtering, pivot/unpivot transformation functions, and changing data types of source/target tables.

With Azure Data Factory, data transformation can be done using the following techniques:

Built-in Data Transformation: Perform light transformations from within the Azure Data Factory user interface.

Azure Data Factory Mapping Data Flows: Create visually designed data transformations without writing code [22,23]. Run at scale by executing the flows

as Spark jobs. Data Flows can call external datasets created by using data services, then use these datasets in logic apps and functions.

Matillion ETL: It uses its own proprietary programming language to create ETL jobs and is specialized in cloud data warehousing. Supports integration between numerous cloud services and on-prem sources. Built for transformation using SQL, Python, Bash, and its own JSON-like orchestration script.

Apache Spark: An open-source distributed data processing framework. It allows you to run code based on Java, Python, Scala, and R projects. It is primarily used to read, transform, and write datasets. Needs to be integrated with orchestration tools to be used as a managed ETL/ELT tool.

3.3. Monitoring Data Ingestion

Once pipelines are created and deployed, we can proceed to executing them against provided datasets. ADF offers several ways to execute pipelines: directly triggering the pipeline from ADF UI, the PowerShell command utility, in code, and by event triggers. While the HTTP activity supports only the following verbs: GET, POST, PUT, DELETE, HEAD, OPTIONS, PATCH, and some have limitations, ADF pipelines can execute from simple data copies to long-lasting data movement orchestration.

Additionally, pipelines can be executed also in a 'debug' mode, which is used for testing the pipeline. When validating pipeline activities by executing pipeline activity in debug mode, Pipeline execution debug mode offers the ability to run the activity against a specimen dataset without bothering about the trigger input and trigger output datasets [24-26]. Unlike trigger runs where the trigger input/output datasets are inherited from the data trigger associated with the pipeline, the pipeline execution debug mode allows you to specify and explore arbitrary datasets, while also allowing you to debug a certain portion of a data pipeline which is great for performance. Each activity can be debugged independently, one after the other. The activity will be submitted to your chosen runtime while highlighting any failures and dumping the activity debug output files from the Data Factory on completion.

4. Event Hub for Real-Time Data Ingestion

There are instances when data does not arrive in the same batch but needs to be recorded as a stream of data. Before modern cloud technology, sending a text message or calling someone was not a thing. Today, we have multiple apps

hanging on our smartphones that allow us to send text messages quickly to anyone in the world, not to forget that these messages fly through cell towers and are delivered in nanometers. The apps, through which we send messages, push these messages to servers in real-time, which are then processed, stored, and pulled as needed [27,28]. The migration of physical transactional computers to cloud apps has resulted in a great amount of real-time data being created every second that must be logged and stored discretely to avoid losing even a second's worth of data. Azure Event Hub provides data connection pipelines for real-time data logging in Azure with little to no configuration needed to handle millions of data creation requests and is simple to manage via an Azure portal similar to Azure Storage.

With Azure Event Hub, configuration mainly consists of selecting the number of Partitions and selecting a Retention Time, which can be configured at either Event Hub or Consumer group level. There's no limit on the number of Event Hub instances, but your application requirement determines the number of Event Hub configurations. Azure Event Hub is billed for what is referred to as Throughput Units available to reserve, which can be increased or decreased based on peak loads. Threshold is also defined at Event Hub level, which can be from a minimum of 1 to a maximum of 128 available at a zone. Additionally, partition keys, partition counts, and retention time are considered to ensure logging is optimized, with data ensured to be available until the application baselines [19,29-31].

For a simple test bed to evaluate Message Queue features for data streaming and logging, two Event Hubs were created with distinct partition counts and retention times. One Event Hub would contain the basic properties and be invoked first, followed by a second Event Hub with extended properties.

4.1. Introduction to Azure Event Hub

With the increase in mobile consumers, the demand for speed, scale, and performance in building intelligent and responsive applications is outgrowing traditional data processing patterns. Industry leaders gather data from social media, and through website usage, credit card sales, and machine sensors to evaluate user activity, improve massive data-driven processes, and optimize correctness and performance [32,33]. This dark data, utilized in real-time, is stored via diverse storage components ranging from SQL Server on-premises to cloud data warehouses. To support the external and third-party data traffic, such data processing systems need to be supplemented by hosting services optimized for high-velocity data streaming.

Azure Event Hub is a fully managed timeseries data ingestion service capable of working with any type of data from any source. In Event Hub, data can be streamed reliably at hundreds of thousands (or even millions) of events per second, either in one pass or over long periods. Azure Event Hub has a number of key features. First, it has a dynamic scale of throughput units; there is no performance sharding or session keys required. Any event is accessible from any partition. It creates multiple partition subspaces of the main event queue, and each partition can store events. You only pay for what you need, consuming servers on demand. Traditional data streams are either in JSON or XML formats. Event Hub uses the binary Avro format which is space-efficient for fast browsing and decoding [34-36]. Once data is in Event Hub, it can then be replayed, sent to custom processing applications, or archived directly to Azure Data Lake Storage, Analysis Services, Cloud Batch, HDInsight, Machine Learning, and other services.

4.2. Configuring Event Hub for Data Streaming

To utilize an Azure Event Hub for streaming data collected from various vehicles, certain configurations should be carried out. First, authenticate to Azure using your Azure account. In this tutorial, you will identify an existing Azure resource group, create a new Event Hub namespace, and create a new Event Hub. It is assumed that an Event Hub namespace does not already exist. If it does, please use that namespace while creating the Event Hub, skipping the creation of the namespace. As an alternative, an Event Hub namespace can be created in the Azure portal.

To create an Event Hub namespace and an Event Hub via CLI, follow the procedure below. Note that after each command you will have to wait for a few seconds to allow the progress of the operation. After the Event Hub is created, copy the connection string and store it temporarily in the text file, since you will need it later. The parameters in the command are appropriate for demonstration purposes and can be modified as needed.

You need to set multiple parameters: Event Hub name, Event Hub namespace name, Subscription ID, and Azure region. To access Azure Event Hub or resources, you can create the event_Hub_data_reader role. While creating the role, prefer the Contributor role under Event Hub Data Access category.

Creating an Event Hub for data streaming will allow several vehicles to publish data at the same time. Depending on how many vehicles you want on the road concurrently, you may need to choose an appropriate partition count for the hub cluster during creation. Any vehicle will push telemetry data to the Event Hub

through an HTTP PUT request. In your case, the vehicle should have a SAS key to push the telemetry data, since you configured the Event Hub for SAS authentication.

4.3. Best Practices for Event Hub

Event Hubs has some features and capabilities that you must take into consideration when implementing a messaging system for your platform to avoid future performance-related issues. We typically refer to these recommendations as best practices. Following these principles will help you take full advantage of the Event Hubs services.

Batching messages will allow you to get better performance and lower costs. As a message broker system, Event Hubs is designed to handle thousands of events each second. Storing thousands of events in memory until the I/O channel can accept it is more efficient than opening multiple connections to write messages individually [37-40]. Messages can be bundled in a single batch using the CreateBatch method. For other languages, such as Python or Java, you can create a batch of messages adding them to a special buffer that is sent only when it is full using the SendAsync method.

Use parallel connections to the Event Hub client library. The client library allows you to send messages to an Event Hub with a single connection. However, to mitigate potential latency when sending messages, it is advisable to use multiple connections when sending messages. Using the SendAsync method, messages can be sent in multiple threads. The client library is designed to accept a parallel connection corresponding to the number of logical cores present in the VM or computer that sends messages to Event Hubs, up to 32 threads maximum.

Make sure to enable Message Deduplication if you are worried about duplicate messages. Since the messaging services work over a public network, there is no guarantee that every single event will successfully be received. As a result, both the client and messaging services must use a specific method to recognize when a message is received. This is called Message Deduplication, and you must always consider and implement it on your solution, especially as it's a Distributed Computing Environment.

Make sure you specify a partition key. Partition keys are important for parallelism on the read side of a messaging scenario. Some Event Hub consumer libraries use a round-robin algorithm for distributing partition assignments. However, if you use partition keys to send some events to a specific partition, and consumers assigned to other partitions do not keep pace with that partition, then the result is a backlog of messages in that other partition and a lower throughput.

5. Streaming Analytics using Azure Stream Analytics

Analytic services that are available now are making it useful to not only store data in big volumes but also to process and analyze that data in real-time as it comes in. Azure has an ability to ingest streaming data and process that data as required. With Stream Analytics we can create jobs that are given input streams, and jobs have query statements which have various operations like filters, aggregations, grouping and joins [41-43]. When the statements are executed on the input data, then the output is produced for further processing. Stream Analytics has capability to schedule jobs to run at fixed intervals and produce output of freshly received data when the input data stream is received in batches. Stream Analytics can connect to to receive the data in a streaming fashion. Streaming data is usually generated by devices that are connected to the Internet, and these devices periodically keep sending messages. It is also possible to ingest streaming data from other services directly.

IoT can be considered as a subset of streaming data analytics, which has devices that have trailing edge processing capabilities. Stream analytics allows to process this data in real time. To make streaming analytics effective, we also need to analyze the streaming data in context with operational data that is available in the traditional storage services. So once streaming processed data is ingested, we can run machine learning models to check for any anomalies as a batch job. Once there is an anomaly detected, the data that is available in Stream Analytics can be used to take immediate action, and more decisions can be made based on the aggregated data in traditional storage methods.

5.1. Overview of Stream Analytics

Stream Analytics service is designed to consume data from a data stream, derive insight data using SQL-like queries, and produce the processed insight data to a variety of data sinks. One example scenario is, for a stock market data stream coming from the stock prices data source, putting a SQL-like Stream Analytics job query to read a real-time stream of the input stock prices from events and processes it to find the stocks with prices dropping or rising after a certain percentage within a certain time period, before emitting signals to trigger sell or buy actions. You can choose to store the processed signals data to a database for later review and analysis.

The Stream Analytics service supports various input data sources such as Event Hub, IoT Hub, and Blob Storage. It can also support various output data sinks

such as SQL Database, Cosmos DB, Blob Storage, and Event Hub. More importantly, the Stream Analytics service is fully integrated with other services securely in the Cloud. By using these services, you truly cannot get this flow-analyzing solution built by connectors and a few lines of Stream Analytics SQL jobs faster!

5.2. Setting Up Stream Analytics Jobs

To set up an Azure Stream Analytics job, you log into the Azure portal, create a new Stream Analytics job by providing the job name, the Azure subscription details, the hosting region and timezone, resource group, and the desired pricing tier and plan. You can set the job to run with the default job configuration or customize the configuration, which allows you to specify partitioning options and corresponding details. After that, you select the output type and configure its options, and then you specify the name of the Event Hub from which the job will be reading the input data [28,44-47]. The source should match the schema defined in the input and seamlessly connect the existing Event Hub or create a new Event Hub resource. You are next prompted for configuring the transformation query. You can create the query using the query designer tool. After the transformation configuration, you can then provide the number rows to be loaded, based on which the Stream Analytics engine processes them to optimize the plan. You also have to specify the output in order to know how you would like the results of the SQL transformation to be stored or used. This includes cloud storage or service options such as Azure Blob Storage, Azure SQL Database, Cosmos DB, etc. After providing the desired selected options, Azure Stream Analytics job reads the source data, applies the required processing as specified in the SQL statement, and stores the results into the destination of your choice.

5.3. Querying Streaming Data

Several languages are used to create streaming analytical jobs. For instance, SQL-like languages are employed to develop jobs in Apache Spark Streaming, Apache Flink, and Stream Processing for MPPs. Furthermore, Managed Streaming for Kafka provides a streaming SQL engine that will query JSON messages. All these systems utilize incremental query processing over data streams. This approach is motivated by the following reasons: first, the distant-in-time results are not generally useful; second, ingesting data streams often involves expensive operations, such as decompressing, validating, and sanitizing large messages based on their headers. Succeeding results can supply reduced-computational updates for the earlier associated with the expensive operations, thus efficiently addressing the mentioned reasons. Finally, incremental

processing equips a feedback loop that complements the limitations of transport layer protocols in latency requirements.

It should be pointed out that incremental query processors are further challenged compared to the traditional non-incremental ones. First, they must mobilize multi-core and distributed clusters compared to the usually executed single-core servers of non-incremental ones. They also must scale their clusters to the needs of users, as the incoming data rates are variable over time. In addition, they have to generally deal with multiple query outputs. Finally, they have to enable user-defined functions to directly process data streams, and efficiently process results when the functions conduct blocking operations. These streaming analytics layers complement post-hoc batch jobs that can leverage either pull or push-based post-query processing. The remaining sections of this text address the user and system components of streaming query processing.

5.4. Integrating with Other Azure Services

Stream Analytics works with datasets in a very specific way. It allows querying streaming data and storing the result to several Azure data services, including Azure Blob Storage, Azure Data Lake Store, Azure Table Storage, Azure SQL Database, Azure Cosmos DB, and Azure Event Hubs. The data services mentioned above expose control access and authorization via the Downstream Data Source access control feature.

In addition to storing the results of a computation using a query, you can use the data services as input for your queries. The data services available currently as input are Azure Blob, Azure Data Lake Store, Azure Table Storage, Azure SQL Database, and Azure Cosmos DB. The service is useful as it limits the connectivity infrastructure requirements. For an Azure Service, the service has to be in the same Azure Subscription as Stream Analytics.

To process data from other Azure Services, the integration must be implemented using custom code, but it is relatively straightforward. For Event Hubs, the Fast Event Hubs library is written for the purpose and can provide large throughput. For Notification Hubs and SignalR, an HTTP Listener such as an Azure Function App can listen for incoming data and write it into a storage service accessible to Stream Analytics.

For Power BI, integrating with Stream Analytics is built-in, allowing you to set up a scheduled query in Stream Analytics and have the result sent to Power BI automatically. The service allows you to visualize the data in near real-time.

6. Building a Scalable Data Lake with ADLS Gen2

With the wide usage of Azure, there comes a need to build a storage repository which is scalable and also able to handle volumes, variety, and velocity along with providing secure storage. Azure Data Lake Storage Gen2 is a set of capabilities that are built on top of Blob Storage and which allow it to have the above-stated qualities. It is a hyper-scalable and cost-effective data lake solution for big data analytics, which is highly available and in which a variety of big data analytical frameworks/templates can work on. The unique structure of ADLS Gen2 allows other Azure Services and Partners to build around it, which makes ADLS Gen2 a go-to solution for Data Lake Storage.

ADLS Gen2 introduces a hierarchical namespace, which is a feature that allows organizing files and directories within an Azure Storage account. A hierarchical namespace improves the performance of operations such as move, delete, and listing. These operations have atomic semantics and are done in an efficient manner. ADLS Gen2 also enables the creation of fine-grained access controls on folders and files. Operations such as move or delete, which are particularly important to Data Lake scenarios, have the required security semantics. ADLS Gen2 provides both POSIX-like IAM ACLs and RBAC, therefore allowing organizations to control access at any level within the storage account. Also, ADLS Gen2 can seamlessly integrate with various Azure products, including Azure Synapse Analytics, Azure Databricks for building Data Lake Analytics solutions at scale.

6.1. Introduction to Azure Data Lake Storage Gen2

Azure Data Lake Storage Gen2 (ADLS Gen2) is a hyperscale cloud repository for big data analytic workloads. Azure Data Lake Storage allows you to build a secure, enterprise-grade data lake that can ingest transcendent amounts of structured, semi-structured, unstructured, and event data as well as data in-motion and at rest so that you can easily maintain and unlock unlimited business value and deep insights when and how you need them. This provides tremendous advantages against the other leading cloud providers as there are no resulting data resolution and misalignment issues that are common with third-party solutions.

ADLS Gen2 is also integrated into all the core cloud services that organizations are deploying to tech-enable themselves. Composable data pipelines that extract, transform, and load data into multiple services are no longer required as organizations can now deploy native analytical cloud services that can natively execute analytics jobs directly against the data in the landing zone; from near

real-time analytics against the data being landed in ADLS Gen2 to scheduling periodic data preparation and cleaning analytics jobs. All these can natively work against the data in ADLS Gen2 without data movement using the native connectors that they provide.

6.2. Architecture of ADLS Gen2

The architecture of Azure Data Lake Storage Gen2 (ADLS Gen2) Lower Layer Storage (LLS) is built on Azure LRS as an additive feature to Azure Blob Storage. This architecture has the following features:

- Lightweight. The architecture has a very low footprint as it is built on top of an existing scalable, reliable, and cost-efficient infrastructure. - Cost-optimized. The capacity and network pricing structure could result in lower storage and network costs for Big Data workloads compared to other providers. - Hybrid cloud. The architecture uses SMB as the access protocol which has hybrid advantages. It can provide both POSIX file semantics (with some limitations) to a wide set of applications running on Azure VMs or on-prem filers, teams of data engineers and data scientists working from their on-prem Windows or Mac computers, as well as block storage semantics to heterogeneous workloads running on Azure VMs, Azure Kubernetes Service in Azure, or any other Azure cloud-native compute service. Compared to NFS, SMB is natively supported on Windows clients, and it has larger market share than NFS on Linux clients. Compared to other proprietary file storage access protocols, SMB has higher usability and is much lighter weight, and is the de-facto standard of the file access protocol. Compared to the proprietary, cluster-wide communication protocols, SMB is natively supported on every client and has lower installation and maintenance overhead. It is also more secure as it provides a full authentication and authorization infrastructure. - Security and compliance. ADLS Gen2 also has deep integration with Azure's security and access control capabilities. It supports role-based access control at the directory and file level, and ACL type permissions for fine-grained security.

6.3. Data Organization and Management

Data stored in ADLS Gen2 can be organized in directories and subdirectories as is common with file systems. These directories can also contain containers, whereas containers in Azure Storage are often referred to as a flat structure. Data in ADLS Gen2 can also be managed using lifecycle management rules. Using these lifecycle management features, data can be conditionally deleted or moved to a cooler storage tier, such as Hot, Cool, or Archive, depending on how frequently the data is accessed. It is worth pointing out that in Azure Data Lake,

regardless of the lifetime of files, they remain available as long as the storage account is not deleted.

ADLS Gen2 offers a new feature called Blob Indexer, which allows us to apply metadata for the data stored in Gen2. Using a key-value pair, it allows us to set up to 256 key-value pairs for a single file, whether it is a text file, JSON file, Office file, PDF, image, or many others. Once these key-value pairs are set, it allows us to use Cmdlets or the API to easily extract and perform queries on the metadata as well as the actual data.

Moreover, ADLS Gen2 also offers event triggers, so that events can be monitored directly from the storage account. These events can be set to run with Functions or send a message to a Topic, such as Service Bus, Functions, or Logic Apps. By doing this, operations can be set up on the data once it is ingested and organized in ADLS Gen2 as well as monitor these operations. For instance, if an image is uploaded in ADLS Gen2, an event trigger can invoke a Function to run Optical Character Recognition (OCR) on the image. The text extracted from the image can then be stored in a database or even sent to a Cognitive Search where the images can be searched based upon their contents.

6.4. Security and Access Control

Much of the security and access features that Azure Data Lake Storage has integrated come from Azure Storage. In addition to the standard Azure Security features for every Azure service, like 1st party Azure Security Monitoring, Security Health Monitoring and Reporting, and native DDoS Protection, ADLS Gen2 provides security at multiple levels, from the storage account itself down to the level of the directory and file in the data lake. Azure Data Lake Storage Gen2 accounts can only be created from the Azure Portal and use AXDS itself as the storage account type [2,4,5].

At the storage account level, you can use 2 different types of accounts: local and geo-redundant. Local accounts replicate data across the Azure datacenter region where the Azure account has been created. Geo-redundant accounts additionally replicate data to a second Azure datacenter region. Replication across data center regions provides a backup of the data for security and availability reasons.

The compute infrastructure that handles Azure Data Lake Storage Gen2 data requests is isolated within its own hybrid model, only aggregating with other customers at special times and points. The Azure Storage service itself uses a Security Token Service method backed by certificates. This means that in addition to key-based authentication, your apps can be enabled to authenticate automatically with the STS Service to access ADLS Gen2 without requiring

storage keys for authentication every time. This is the recommended way to authenticate the app as it is more secure, providing support for corporate employee login information where the STS will pass back temporary tokens that are used in-memory.

7. Data Governance in Azure

Data governance constitutes the framework for ensuring the availability, usability, integrity, security, and compliance of the data in your organization, allowing your organization to leverage its data as a business asset. What typically does not work is to assign data governance to a few members or consultants and ask them to deliver data governance. The team needs to work with business stakeholders to create use-case-based data and analytics projects as proof of concept for what data governance will look like. Use trials to define templates to help departments enable data governance. Use the experiences from the trials to build a community of interest around pragmatic use of data and analytics. Following the projects, departments become ready to take ownership and will have kicked off their own projects and reworked templates.

The first step to data governance consists of enabling the right people in the organization to be responsible for their own data. Assign ownership of the data for each project. These data owners are the link between the technical definition of data and how the use of data is governed in activities, roles, and tasks in the department. Information technology needs to provide cultural and domain leadership, tools, and support during various phases of the data governance process. Providing infrastructure and tools to allow teams in a department to enable business data governance can sustainably do this. Gamified data management platforms allow departments to be more accountable for their own data while collaboration tools facilitate user literacy and knowledge flow. By creating a dedicated space to store information on their data and relevant business context, business users can create a first layer of data and knowledge governance before technology experts enrich and socialize the data itself.

7.1. Importance of Data Governance

Data governance is often an afterthought when organizations pursue a modern data strategy. But weeks or months into a data project, the lack of governance policies becomes obvious. It starts to impact organizational work processes by introducing chaos to the workflows. Individuals become reluctant to leverage shared assets when contributing data. Duplicated and redundant data proliferate.

Improperly curated content is cited, resulting in inaccuracies in analysis and reporting.

The bigger the dataset and the more people that touch it, the greater the chaos and the more imperative the need for established governance processes. A company with a few thousand customers needs to manage privacy well, but may not be forced by practicalities to have explicit data governance processes around their pleasing their customer relationships. But a healthcare analytics company that combines data from several states in processing their claims, or a social media company that chronicles each frenetic second of a million users' online lives, will be compelled to have elaborate data governance processes to comply with regulations, keep customers happy, and stay in business.

Government regulation mandates various standards for governance. Protects privacy and security for citizens' data that is stored and processed by any company globally. Will honor that trust for consumers. Sets down requirements for electronic medical records. Protects credit card transactions. Payment platforms must adhere to strict security policies to avoid liability for data theft and fraud. Tax preparation firms must invest considerable resources in cybersecurity to maintain the trust and goodwill of their customers.

Nor is regulation the only motivation for data governance. There is also a strong market force driving its enforcement. A common way to unlock the value hidden in a company's data is through cross-department collaboration. Once disparate business units create a data lake populated with their organizational data, business units previously unaware of each others' data may find competitive advantage in analyzing that merged data.

7.2. Implementing Governance Policies

In the previous section, we identified role assignments, data protection, and resource management as the key governance components. But how do you implement those components? With Azure, you define and implement a governance policy for your organization. A policy encapsulates all business rules, and compliance ensures resources used in your organization are built following the established rules. We can think of the policy as a factory automation rule defining how products (here, Azure resources) in an organization should be built.

Implementing a policy can be done with Azure Resources Policies, by defining Suggested and Required Policies in different Assignments, which assign locations for each Azure Subscription, or by using Azure Management Groups. A Management Group can contain multiple subscriptions; those can be used to assign a higher-level policy for a set of subscriptions. The advantage of this

management group hierarchy is that it enables you to assign policies at a higher level so that they apply to all lower levels. For a subscription, you can define Required Policies or Default Changes; it is also possible to define lower-level policies for the same resources or resource groups. The subscriptions are also part of the Azure Organization, which means that services tags or category tags must be present in each Subscription to prevent the traversal to the Management Group. In a Resource Policy definition, you can define the logical processing with the Not, AllOf, AnyOf, and True functions, which allow or prevent changes or even queries on following or other Azure attributes. Then, a policy is assigned to one or multiple resources to query or manage.

The Azure Policy Resource and Role for Azure Data Governance can give us the contact and resources needed to handle Data Governance in Azure. With Azure, you can be sure that management is automatic and that your infrastructure is predictable.

8. Performance Optimization Techniques

It is very common to improve data ingestion and storage performance with some optimizations in the ingestion and compute stages. In this section, we summarize relevant techniques to achieve quicker data movement from on-premise to Azure and optimized storage performance in Azure.

8.1. Optimizing Data Ingestion Processes

Most optimizations recommended here refer to Azure Data Factory. For ingestion performance, you have different options that may affect the time for data transfer. The first one is the regional settings of the integration runtime. A self-hosted integration runtime enables data transfer at cloud data planted region nodes so make sure that the integration runtime is placed as close as possible to the region where the Azure storage account is allocated. Consider, when designing a data factory to use Azure Portal to create the repositories on Azure and the resources needed, and only use templates when you have finished the process in Azure. Also, consider using PowerShell, versus the portal, or templates via Git to minimize access time in the data factory.

8.2. Scaling Azure Resources

Since performance is a topology and usage pattern and service-specific notion, there’s no one-size-fits-all answer to design a performant Azure service. Nonetheless, a majority of Azure resources expose configuration options like

scale, size, degree of parallelism, etc. which allow you to back your Azure resources with the appropriate performance. This can be achieved through native support in the Azure SDKs, PowerShell commands, Azure CLI command, Azure Portal, or templates. Choose accordingly to the need. When using serverless options, performance for services like Azure Functions or Logic Apps is managed by Azure. You need only to design and code your business logic, taking care of some aspects like implementation patterns, setting up connectors to be reused, and others. When considering performance optimizations in long-running operations like big data ingestion pipelines, multiple factors are to be considered.

8.1. Optimizing Data Ingestion Processes

The performance of any application running on Azure depends on almost all the factors that influence its performance while running on-premises infrastructure. Among these factors, the performance of data ingestion processes may heavily impact the overall performance, especially if such processes are responsible for making the data available for analysis in near real time. In this section, some of the techniques that can be used to optimize the performance of data ingestion processes are addressed.

Data ingestion processes can be optimized at different levels, for example, optimizing the design/architecture of the application responsible for the data ingestion, optimizing the Data Flow activities inside the data ingestion pipelines, and optimizing the parameters of the services/resources involved in the data ingestion process. Speeding up the processes that are responsible for the data production and storage might not be possible because, in most cases, they are external to the Azure environment. The next sections explore the common techniques that can be used to optimize data ingestion.

The first thing to consider is some patterns that can be followed to design data ingestion solutions. Such patterns follow the traditional experience built over the years when building ETL processes in any tool available to perform such task. In the end, the design of data ingestion processes usually consists of the same basic ingredients: data should be processed in batches, when possible, data should be processed in parallel, and the processes must be idempotent [28,44-47]. These three principles can be applied at different levels throughout the ingestion process, as well as the data processing and staging services and tools involved in the data ingestion.

8.2. Scaling Azure Resources

Data Integration Services allow you to create, edit and schedule data integration workflows. These services enable you to create and schedule pipelines that can

ingest, process and orchestrate data. The pipelines can perform discovery, ingestion, transformation, and movement or orchestration of data flows. They can schedule and control data integration and data movement between different stages of your process [48,49]. These services help you build, orchestrate and auto-trigger workflows built using several components or tasks and can ingest and flow data from heterogeneous data stores.

If a Data Factory Pipeline does not yield the desired performance or throughput, or takes unusually longer to execute due to bottlenecked or expired delete activities, performance tuning is usually the first step in diagnosing the problem and potentially resolving the underlying issue. However, if it takes too long even after optimization, the pipeline may be implemented with gradual and controlled scaling or parallelism of pipeline properties. It’s usually possible to finish a data pipeline quickly, but that would involve additional API usage which could cost additional money, not to mention hitting throttling limits for dynamic workloads.

Azure provides a convenient way to increase some resources available to your services to allow for increased workloads. This method is called scaling. Scaling is actually pretty much touching on only two components in Azure – vertically or horizontally scaling bundles of infrastructure called instances and VMs on a Size by Location setting. Both vertical and horizontal scaling involve modifying the quantity or scale of Azure resources and can work either together or separately to achieve the desired level of service. Work for these services is partitioned between services using queues and each service uses a separate processing tier. At a high level, horizontal or vertical scaling are distinct ways to increase the power of your Azure Data Integration active pipeline.

9. Case Studies and Real-World Applications

Data is an important part of every organization's day-to-day operations. These organizations use data to measure their work and take meaningful actions. Using analytics, companies can make better decisions; improve current processes, products, and services; find competitive advantages; and better understand their customers or target markets. However, the amount of data involved in day-to-day decisions increases every day. Organizations must decide which data to use, how to structure that data, and how to make that data available to teams through processes, products, services, or other tools. To solve these complex issues, organizations consider utilizing cloud services that expose advanced tools and services that require minimal configuration. This chapter presents ethnographic

case studies on the implementation of services to assist with these tasks and how one cloud provider system can assist in common data ingesting and storing tasks.

These case studies draw from a collection of successful and unsuccessful data storing and ingesting implementations of services. The goal of this chapter is to present and facilitate conversations around these implementations and help organizations speed their learning and implementation processes. These case studies cover a range of industries and sizes, including pre-revenue startups, national healthcare systems, regional healthcare organizations, national telecommunication providers, nonprofit shared infrastructure organizations, large banking institutions, and academic research. The findings of this chapter emphasize the need for best practice literature while also developing a resource to facilitate learning and future design conversations.

9.1. Industry Use Cases

The world is moving towards accepting data as the new oil. Industries that leverage data by storing and manipulating it are reaping the benefits. A reliable platform can achieve this data ingestion and storage pipeline. However, it is essential to understand the different types of ingestion methods and storage solutions used before approaching any business problem within any domain. In this chapter, we study popular business domains such as retail, ad-tech, and media, and how various companies from these domains are leveraging data for better decision making. We investigate the various data ingestion methods and storage options used. We draw various conclusions and assumptions based on their implementations.

In the retail domain, various companies have implemented different data ingestion methods. For example, one company makes purchases from various different retailers and stores them in Blob storage as a raw file. Their storage is highly available, durable, and vastly scalable for large files. On the other hand, another company integrates and analyzes behavioral data so customers can receive coupons for specific products at the times that they are most likely to buy. Also, customers need to load coupons on their members' cards before they are eligible for a discount. This company efficiently designs this pipeline using various components. Another example is a retailer, which uses Event Hubs and Stream Analytics to detect sales outliers in near real-time, enabling the company to understand and adjust supply chain management as necessary. This retailer implemented a pipeline that successfully identified outlier changes in sales and conveyed possible motives. By computing alerts in near real-time, the retailer was able to react to potential issues and discover new demand-driving events much faster.

9.2. Lessons Learned from Implementations

In Chapter 7, we reviewed a wide range of scenarios along 12 different application domains, from industrial automation to financial services, including travel, advertising, logistics, utilities, climate change, cybersecurity, social media, retail, and smart cities. These scenarios encompassed multiple types of obligations, from real-time predictive monitoring and decision support to historic forensic analysis and insight generation. In Chapter 8, we further illustrated the implementation of these kinds of systems through 14 different projects for companies. Based on experiences from the aforementioned industrial projects, here, we summarize a few important lessons learned about the development of data ingestion and storage systems.

TL; DR: Deploying a successful data ingestion system is not necessarily trivial. We found that one needs a combination of suitable tooling, adapters, and eventually – if not commonly – custom development in conjunction with a solid, modular design. While the tooling landscape is relatively mature, we oftentimes found that our ingestion systems were close to unique. Just using a singular existing solution lacked adaptivity and scalability. Moreover, we predominantly used traditional data storage and data warehouse solutions for storing raw data, often augmented by search solutions. For storage architectures dedicated to data analytics for opportunistic workloads, we predominantly employed single storage systems with raw data retention. We used relational database technology or other industry-grade storage systems for structured storage with high query throughput demands. Yet, for both data storage types, we solely implemented systems with small data retention times by importing only recent-period data and aggregated historical analytics results.

10. Future Trends in Data Ingestion and Storage

As data technologies continue to advance at a rapid pace, organizations require the latest technologies and tools to remain competitive and embrace a data-enabled culture. While some companies continue to ingest and analyze data in batches, eventually shifting to streaming architectures, a growing number of organizations are choosing to ingest data as events, in real time. Similarly, many organizations continue to store their data on premises and in data warehouses and data lakes, while many others are choosing cloud solutions because of their lower costs and better performance.

Edge computing and 5G communications are enabling organizations to store and process their data closer to where it is generated. Organizations are choosing to use edge computing devices to filter voracious volumes of data, leaving only the relevant data to be sent back to centralized storage clouds. Other organizations are adopting centralized storage solutions for the large data volumes they generate and store. In practice, many companies do both, using hard disk drives and cloud solutions for large volumes of data while relying on low-latency flash and all-flash storage solutions for real-time data transmission chains and workloads.

As companies reconsider their ingestion and storage architectures to reduce costs and improve processing times, the cloud is attracting a growing share of on-premises storage and processing requirements. Organizations are also considering their platform requirements for moving to the data cloud. Open computing architectures are quickly becoming a trend for both storage and computing because they can support a wide range of data processing workloads from HPC and AI to the more traditional data warehouse and massively parallel processing workloads. It is likely that some future deployments will rely on heterogeneous computing solutions, where some workloads are running on traditional processors while others are offloaded to accelerators.

11. Conclusion

In this book, we discussed various aspects of data ingestion and storage in Azure. We recognized this need in the age of Big Data, which comes in many forms and from various sources. Once data is ingested from data sources, the next step is to store the data. We highlighted that Ingestion and Storage are closely related operations. The stored data serves as a source for various Azure services, such as Analytics, Machine Learning, Data Science, and Visualization. Thus, careful planning and execution of ingestion and storage operations would lay a good foundation for successful Big Data projects, which would provide a good Return on Investment. We also mentioned that Data Lakehouse concept extends Data Lake and Data Warehouse concepts, but unifies the benefits of both these systems. Thus, in Azure, it is natural to use Azure Data Lake Storage service for storage and Azure Data Factory as the main tool for ingestion of data from sources to target Data Lakehouse systems.

References

[1] Zhelev S, Rozeva A. Big data processing in the cloud-Challenges and platforms. InAIP Conference Proceedings 2017 Dec 7 (Vol. 1910, No. 1, p. 060013). AIP Publishing LLC.

[2] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.

[3] Simmhan Y, Aman S, Kumbhare A, Liu R, Stevens S, Zhou Q, Prasanna V. Cloud-based software platform for big data analytics in smart grids. Computing in Science & Engineering. 2013 Mar 7;15(4):38-47.

[4] Dzulhikam D, Rana ME. A critical review of cloud computing environment for big data analytics. In2022 International Conference on Decision Aid Sciences and Applications (DASA) 2022 Mar 23 (pp. 76-81). IEEE.

[5] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).

[6] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.

[7] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.

[8] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.

[9] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.

[10] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.

[11] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.

[12] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.

[13] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.

[14] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.

[15] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.

[16] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.

[17] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[18] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[19] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[20] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.
[21] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[22] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[23] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[24] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[25] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[26] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[27] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[28] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).
[29] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[30] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[31] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[32] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[33] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.

[34] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[35] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.
[36] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[37] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.
[38] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.
[39] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[40] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[41] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[42] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[43] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[44] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[45] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.
[46] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[47] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.
[48] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.
[49] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 5: Data Processing and Modelling in Azure Ecosystem

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

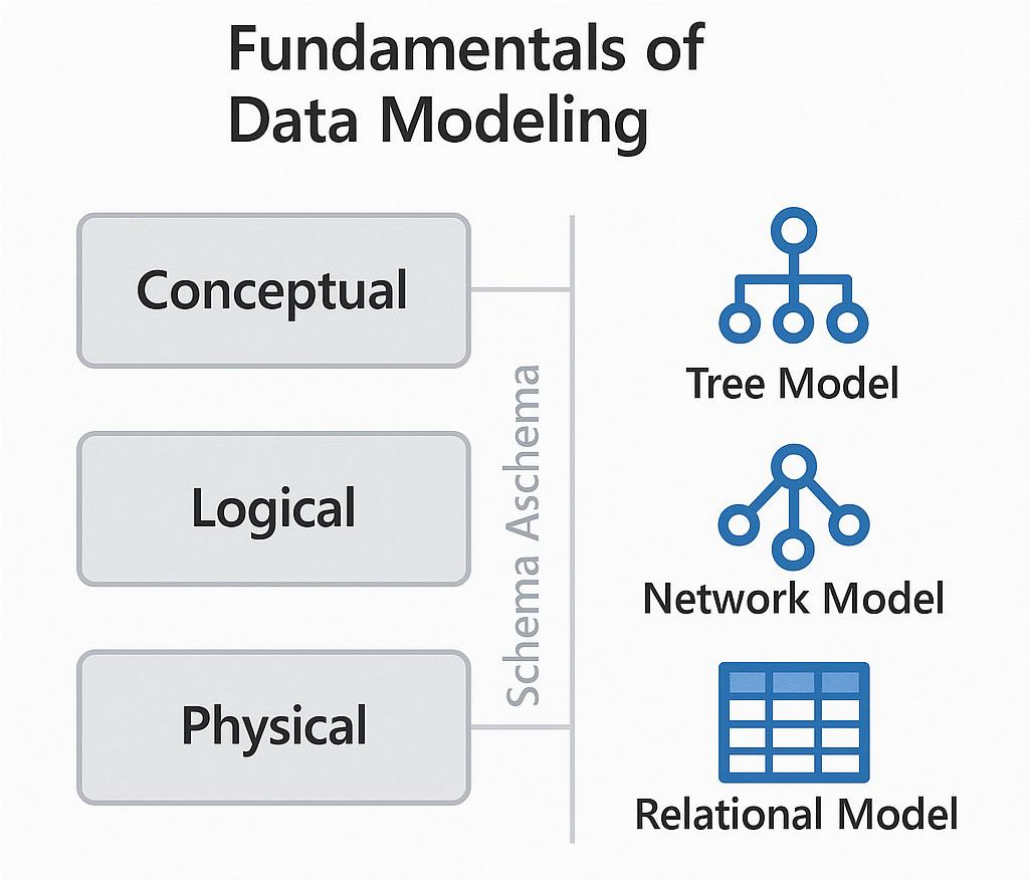
1 Introduction to Data Processing

The data processing workflow typically consists of potentially multiple sources of raw data stored in Database Management Systems, Data Lakes, Data Warehouses, and various types of files in object storage and file systems. Although there is no unique or universally agreed-upon definition for each of these types of information storage systems, they essentially differ in various aspects including scalability, types of user demand, preferred physical design, format, and organization of stored data, data access protocol, etc [1,2].

Collectively, all of these distinctions define the various performance and functional characteristics reflected in their Individual Areas of Applicability. Each of the data processing applications individually accesses and utilizes relevant subsets of the raw data either to curate it for long-term archival or to create analytic data products specifically tuned for their intended use cases. The curated data products are typically more refined, more compact, and optimized for their intended use cases than the raw data containing higher information entropy levels. Various physical, budgetary, and time constraints imposed on the overall workflow define the performance and functional tradeoff metrics such as data latency, economic efficiency, ease of operation, etc., specifically relevant to the organizations, departments, or groups of teams required to maintain it [3-5]. The resultant choices define the logical architecture of the global data ecosystem.

The curated data products may in turn be used across the larger organization or business to address a wider variety of cross-cutting and independent commercial

services. Collectively, all of the different data processing applications required to successfully build and maintain the various analytic data products form the Data Processing System. The primary motivation to create the Data Processing System is to mine for precious business knowledge hidden in the data to create long-term business value for the hosting organization.



2. Batch Processing with Synapse Pipelines

Synapse Pipelines is a serverless option for orchestrating extract, transform, load (ETL) processes for batch processing in batch mode. Pipelines orchestrate the movement and transformation of data using scheduled activities and control flow structures that can be created through an intuitive drag-and-drop experience, or built in code using templates or CLI. While Synapse Pipelines has its own web-based graphical authoring experience for pipeline creation and management, it can also be run and monitored from within the Synapse Studio. Pipelines may

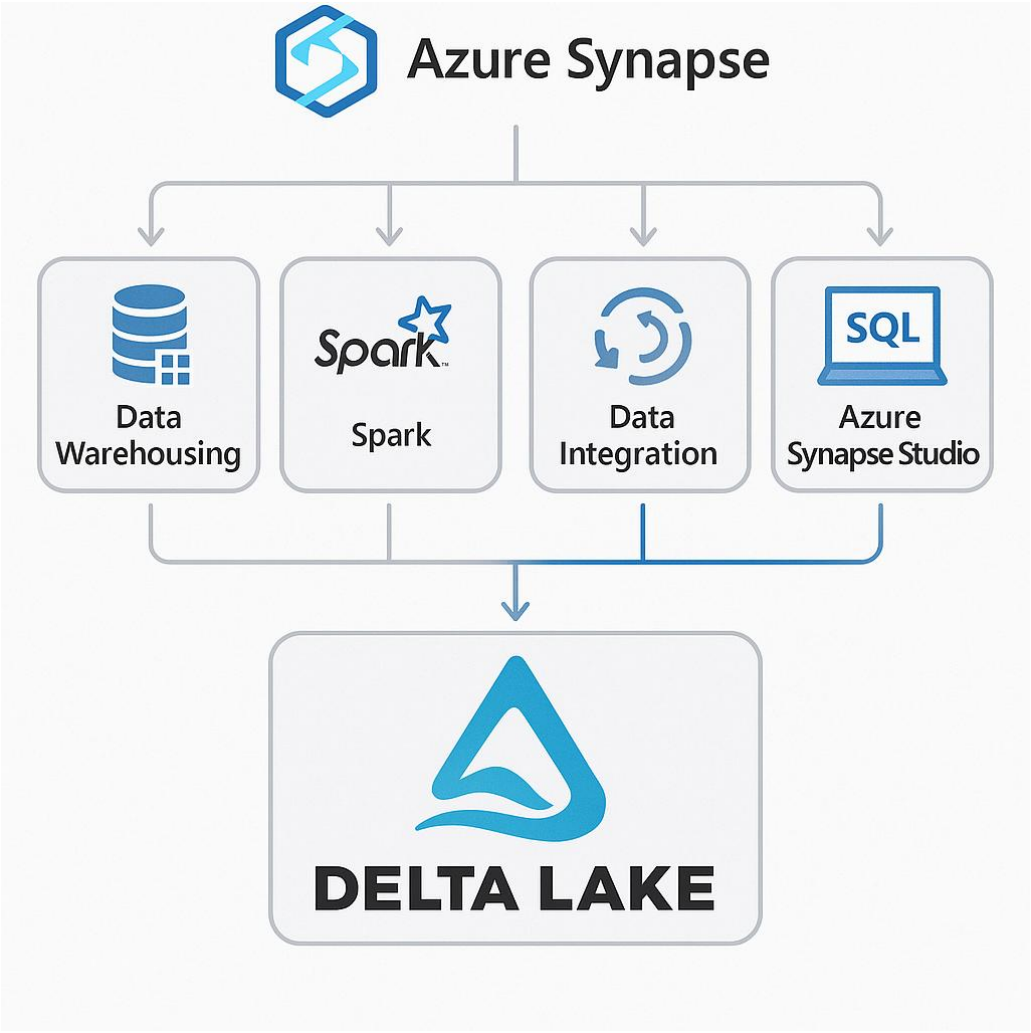
connect to and manage data within a variety of differing data stores, including Data Lake Storage, Blob Storage, Cosmos DB, Oracle DB, SQL Server, MySQL, and many others [6,7].

Pipelines operate on the concept of activities. Such activities can take the form of data movement, data transformation, or general control flow execution. Data movement can be executed via internal Copy Activities or using a set of Data Flow activities that manage data transformation during the movement. Data transformation can be executed using native data transformation capabilities in either Synapse Data Flows or data transformation capabilities available in Data Factory [2,8-10]. And general control flow can be executed via webhooks that invoke functions or services hosted outside of Synapse or Data Factory Pipelines. Pipelines act as a workflow orchestrator that connects, monitors, and manages activities occurring both inside and outside of the Synapse and Data Factory.

3. Delta Lake in Azure Synapse

SYNAPSE & POLARIS: You can now build a lake house architecture using Azure Synapse and Delta Lake. Azure Synapse Data Warehousing allows you to build your enterprise data warehouse or on-demand SQL serverless that accesses the data in Delta Lake tables. Azure Synapse Spark uses Apache Spark and Delta Lake to provide a big data processing experience. Azure Synapse Data Integration has a pipeline activity that executes in Spark and provides a visual authoring environment for Delta Lake pipelines. Azure Synapse Studio, the top-level IDE experience, also provides a unique integrated experience for analytics developers working on both lake and warehouse. It uses a unified metastore to give a single view of warehouse and lake data and a query editor that runs any T-SQL query on either data stored in the warehouse or lake. Finally, Synapse Studio is integrated with additional Azure services that help build analytics solutions.

Azure Data Lake Storage Gen2 allows you to build your data lake, a secure and cost-effective cloud storage solution designed for big data and you can access it directly or via Azure Synapse Data Integration. Delta Lake allows you to add data reliability and data integrity, including ACID transactions, to your lake while running your big data jobs using Apache Spark. Delta Lake implements these core features using a delta log stored in the data lake, which allows concurrent read and write accesses, is scalable without additional overhead, and can be used as a unified metastore [1,11-12]. It is natively accessed by Apache Spark using its Delta format.



These are the unique capabilities of Delta Lake within Azure that help you solve today's analytics challenges. Easily building a lake house architecture using Azure Synapse and Delta Lake with a multi-language development experience using SQL and Spark. Enabling both analytics and data engineers in the same IDE with a single integrated experience giving a unified view of lake and warehouse contents with fine-grained security and cost-effective storage for both lake and warehouse data. Running your analytics workloads for both batch and near real time data and protecting them from failures in concurrent updates.

3.1. Overview of Delta Lake

Delta Lake is an open-source storage layer, used to build data lakes on large-scale files stored on cloud storage and enhance their reliability and performance for data processing. To achieve that, Delta Lake provides transactional guarantees

for schema management, concurrency control, and atomic writes. Delta Lake is supported in a number of data processing services and tools, including Databricks, Apache Spark, Apache Hive, Apache Flink, and Presto. Delta Lake is also an integral part of Lakehouse architecture, a highly efficient hybrid of data warehouse and data lake concepts.

Delta Lake extends the capabilities of cloud-native object storage to support data processing workloads with large scale and complexity, using the "lakehouse" data management paradigm. In a lakehouse, data is kept in cloud-native object storage as files organized in a hierarchical structure, similar to data lakes. Like a data warehouse, the lakehouse provides transactional semantics on writes to the landing layer, audit/replay features on data changes, data structure governance, performance optimization utilities, and a fully managed query service on the data structure [13-15]. To make that policy-driven management possible, the lakehouse stores metadata in a unified metadata store, which accommodates data structure and asset management capabilities for both raw source and the processed data in the landing layer. Delta Lake is the only open-source tool available today that provides transactional features for large-scale file uploads to cloud storage. Delta Lake is maintained by the Linux Foundation.

3.2. Key Features and Benefits

Delta Lake overcomes limitations of traditional data lakes enabled by cloud storage. Data lakes on blob store allow for scalable, low-cost storage for massive amounts of data across multiple formats. However, the lack of ACID transaction support makes it difficult to use these solutions for operations requiring concurrency. For example, high concurrency data ingestion processes cannot publish updates into tables that are being read into notebooks, reports, dashboards, or other environments. As a result, big businesses need to choose more expensive database systems.

Delta was originally developed to overcome the limitations of distributed data lakes. Over time, it has been enhanced into an open source project supported by a large ecosystem of tools and has been renamed to Delta Lake. It is now a project with active companies participating in its development efforts to add features in a sustainable way. Delta Lake uses cloud object storage to store data in Parquet format and adds a transaction log to store metadata as well as update history for the Delta Tables. Built-in Delta Tables are stored as Delta Lake format and can be queried through SQL or tagged as dedicated SQL objects for new insights. Data engineering teams can leverage Delta Lake to build Delta Tables with ETL pipelines or Notebooks.

While Delta Lake does not require a Data Engineering team with years of expertise to build and maintain data pipelines, it has added features that make it easy to use and recover from errors for such teams. For example, duplicate data in Delta Tables can be removed with a command while partitions can be easily added or dropped. Delta Lake adds transaction support, schema enforcement, schema evolution, time travel, support for concurrent read and write, upsert and other features that are traditionally available in databases and data warehouses, on top of data lakes.

3.3. Integration with Azure Synapse

Delta Lake is integrated with Azure Synapse Analytics and is built into both the Azure Synapse workspace and Azure Data Factory. When you build your data pipelines in Azure Data Factory, you can create mapping data flows that leverage Delta Lake's ACID transaction support to run data transformations. Azure Synapse offers an enterprise-ready platform built for data integration, analytics, and consumption, and provides a collaborative user experience that spans Architecting, Visual Development, Automation, and Monitoring [16,17].

Delta Lake provides support for your Azure Synapse based lake house architecture. In addition to the Azure Synapse features that are built around Delta Lake, some of the key features of Delta Lake that offer value in an enterprise-ready architecture include the following: Time travel (data versioning): Delta Lake makes it easy to roll back against earlier versions of Delta Lake tables. ACID transactions: Delta Lake supports atomic write operations. Schema enforcement: Delta Lake supports automatic schema enforcement through Merge and Merge Into operations. Schema evolution: Delta Lake supports automatic schema evolution with Merge operations. Data validation: Delta Lake supports Type checking during Writes. Data Quality: Delta Lake supports CHECK constraints. Data Curation: Delta Lake supports Data Quality enforcement with DQ reports.

Analytics workloads including batch processing, hybrid transactional and analytical processing, and BI with Microsoft Power BI, can all work directly with the data in the Delta Lake as an enterprise source of truth for analytics and reporting that ensure the best possible quality of reported data. Data scientists can empower machine learning operations to do the same with Spark workloads in Azure Synapse and Azure Machine Learning Service [12,18-20]. Query performance can scale up or down to meet your workload requirements by adding or reducing compute nodes for Spark, or using the pause and resume features for Synapse Serverless. Built-in caching further boosts performance for repeated workloads.

4. Spark in Azure Databricks

The Azure Data Science environment we've just explored is very suited to Data Analysis Tasks. But most of the Data Engineering and Processing Tasks must be done in The Azure Databricks Ecosystem. Built on Apache Spark, it allows for distributed processing of Data Tasks across Multiple Machines in a Cluster, allowing Tasks to finish faster. Below we summarize the contents of this section briefly. First, we'll introduce Spark, what is it, its integration with Databricks, its main components, and why we have chosen it for Data Processing Tasks in Our Research Work. Next, we describe how to set up a new Databricks Workspace, create a new Notebook, and run some Spark commands. Finally, we finish by describing how to use Spark for Data Processing Tasks, including the DataFrames API, Data Queries with SQL, Data Exploratory Tasks, and finally some Performance Considerations [21-23]. Data Processing and Modeling Tasks typically require Training Runs regularly due to New Incoming Data or Changes in Old Data. These Tasks need to be executed on all available Training Data, to ensure the Best Performance of the resulting Models. Furthermore, the Amount of Available Data might be Large to the Point that the Ordinary Machine used for Development cannot Handle it. Given these Considerations, we feel it is imperative that any and all Data Processing Tasks and Steps are done in an Infrastructure with the Ability to Scale Out on Demand without Breaking the Process of Development. The Azure Databricks Ecosystem incorporating Apache Spark was built for exactly that.

4.1. Introduction to Apache Spark

Apache Spark is an open-source framework that provides a fast and easy-to-use engine for large-scale data processing. It was created for speeding up workflows and to provide a more general data processing framework. Spark builds on concepts of the model, using the idea of processing a sequence of transformations over a data set, using distributed lazy evaluation, where data is not processed until an action is called. Data is cached in memory by default, so iterative processing, which is common in machine learning and other workloads, runs much faster on Spark. However, it also provides methods to save data to disk in case of errors or if the data set becomes too large to fit in memory.

Spark accepts its own application programming interface and provides languages bindings for Scala, Java, Python, and R, as well as a SQL language for creating SQL-like queries. It can read data from various sources, including local file systems, distributed clusters, and others [24,25]. Data is stored in Resilient Distributed Datasets in a lazy fashion, only being read or written when a

corresponding action is called. RDDs are fault-tolerant collections that can be rebuilt based on lineage information in the case of a failure. Spark can be executed on a local machine or distributed across a cluster, where it supports multiple data processing engines for various workloads, including batch processing, real-time stream processing, machine learning, and graph processing. Moreover, it is integrated with files and objects on distributed cloud storage systems.

4.2. Setting Up Spark in Databricks

In this section, we will discuss how to set up Apache Spark in Azure Databricks. Before doing that, let us get an overall idea of Databricks to know what to expect and what our requirements are. Azure Databricks is an Apache Spark-based analytics service optimized for Azure. In addition to the out-of-the-box features of Apache Spark, Azure Databricks has additional capabilities, including automated cluster configuration and management, a highly optimized runtime engine for practicing Spark SQL, and out-of-the-box integration with Azure services, including Azure Storage, Azure Event Hubs, and Azure Machine Learning, among others. Azure Databricks accelerates Azure data services for analytics, data science, and machine learning scenarios.

Azure Databricks is a managed service that simplifies setting up and deploying Apache Spark when compared with a self-managed environment. Because Azure Databricks manages the infrastructure, users can quickly start coding and derive results. This section provides a step-by-step guide to creating a Databricks account, loading and preparing the data, and configuring Apache Spark in Databricks. Users can sign up using various subscription offers [26-28]. After you create your Databricks workspace, you can start with creating a cluster and your first notebook. A cluster is a set of computation resources that run your notebook code and is required for executing your notebooks. The resources include virtual machines hosted in Azure. Notebooks are web-based interface that enable you to create documents for executing code in Databricks.

4.3. Data Processing with Spark

4.3.1. Spark Internals

Spark in Databricks

Apache Spark is an open-source, distributed computing system that enables fast computing by using memory. It was developed in 2009 and open-sourced in 2010. In 2013, it became a top-level Apache project. Spark provides an interface for programming entire clusters with implicit data parallelism and fault tolerance.

Spark also supports a set of higher-level tools, including Spark SQL for SQL and structured data processing, MLlib for machine learning, GraphX for graph processing, and Spark Streaming for streaming. In the Databricks platform, the Data Processing APIs provided by the Spark kernel are made available as native Python, Scala, SQL, R, and Java APIs that run in different execution engines and runtimes integrated with the Databricks notebook.

The Spark kernel, together with the File/DB Kernel add-on, delivers support for native kernel-based IPython execution. The support and popularity of the Spark platform are due to its early start, low latency, multiple programming languages, arbitrary data processing, setup, simplicity, ease of deployment, faster performance due to in-memory data caching and optimized query execution, scalability, support for a rich set of higher level libraries for common tasks, its easy-to-use API, and its support for both batch and streaming data modes [29-31].

With Spark's rich set of APIs, data can be processed using Spark in the same way data is processed in other programming environments. To further clarify, we illustrate in the following sections, how data processing can be done in Spark, using the popular word count example. It is of course not the only word frequency count example done in Python, but it is a standard go-to example for explaining data processing in MapReduce-based, distributed data analysis environments.

5. SQL and Spark Notebooks in Microsoft Fabric

Microsoft Fabric is a cloud-based unified analytics platform, which can be optimally employed by companies to combine analytics use cases that require data engineering, data warehousing, and machine learning operations. Fabric combines the power of Microsoft Azure and SaaS Business Intelligence tools with software development tooling and open source analytics Stack. The main goal of Microsoft Fabric is to enable companies to analyze their data in a more streamlined, efficient, and cost-effective manner [3,32,33].

Microsoft Fabric is powered by the SQL engine, which abstracts much of the complexity of these custom infrastructure components while leveraging the power of the open-source foundation to scale in a cost-effective and high-performance manner. Fabric integrates and enhances the SQL engine with an easy-to-use UI, rich notebooks experience for SQL and Python, integrated dataflow orchestration for Spark and Dask, without-code and low-code experiences, multi-language notebooks that allow using SQL, Python, R, and

Spark languages together in a single notebook, low-code SaaS and developer experiences, and source control built into the platform. These advancements allow Fabric to unlock scenarios for architects, builders, and business users. From the Spark notebooks, you can leverage Spark and Dask support for your Python/R needs, and those notebooks are integrated with our data pipeline service behind the scenes to allow for easy data preparation workflows.

In this chapter, we will demonstrate how to create notebooks in Microsoft Fabric, both SQL and Spark notebooks, and what development features you might find useful. Let's start with SQL Notebooks, which leverage the SQL engines in Fabric.

5.1. Overview of Microsoft Fabric

Microsoft Fabric introduces a Unified Analytics experience in the context of a Lakehouse. No more pipelining between different tools with dedicated experiences for data prep, data engineering, machine learning, and business analytics. With Microsoft Fabric, all of these experiences are accessible from a single pane of glass, and all directly targeting a single data source: the OneLake, the OneDrive for Data which is a managed and governed layer on top of the Microsoft Azure Data Lake Storage service [4,34-36]. The OneLake is the common data layer for all Microsoft Fabric workloads: dataflows, pipelines, catalogs, SQL, Spark, data science, report, dashboard, etc. And they allow both self-service Analytics and Enterprise Data Analytics Governance experiences!

Microsoft Fabric consists of several components. The Data Integration component allows to orchestrate events between data sources, moving and transforming data into the OneLake, and connecting on-prem and cloud data sources to other Fabric workloads. Power Query, the very popular Microsoft Tool for low-code, no-code Data Preparation is the tool in charge of Development, using pipelines for execution. The Data Engineering component enables data engineers of all skills levels to discover, transform, and analyze data stored in data lakes. Fabric's Data Science experiences empower data scientists with PySpark and R Data Science capabilities on Parquet data. The Data Warehousing component brings industry leading speed for analytics workload, nurturing a SQL experience driven on top of Fabric's Lakehouse [37-40]. The Real Time Analytics component ingests streaming data into Fabric's Lakehouse using the familiar Spark Structured Streaming API. The Business Intelligence component combines the buzz of Power BI.

5.2. Creating SQL Notebooks

SQL Notebooks are a high-level container to work with SQL language-based Fabric Data Warehouse Data. Within Notebooks, you can organize your work – by splitting various tasks within the same Notebook on separate cells – and easily visualize output examples (still in the same Notebook). The summary output of every executed Notebook cell is either a Markdown text (that you can create on every cell) or a SQL Language query output.

A functional SQL Notebook can be created in two ways: - You can explicitly create a Notebook and add SQL cells to your Notebook. - You can use the "run" option of your SQL Script to run your script. If you use this option, the outputs of all commands in your SQL Script are automatically saved into temporary markdown cells, creating a new Notebook for you. You can, then, save this newly created Notebook.

Creating an SQL Notebook by creating the Notebook means 1. Select the "Notebooks" page from the left-side menu of the Microsoft Fabric Studio. 2. Click the "New" button from the upper left side of the Notebooks page. 3. Select the "Notebook" item to create a new generic Notebook. 4. From the newly created empty generic Notebook opened on the screen, click to the "+" button on the top center of the Notebook. 5. Select the "SQL Cell" item to add a SQL Cell Section to your Notebook.

Creating a Notebook based on an existing Script allows you to quickly create a new or to review an existing SQL Notebook. You can run any existing SQL Script to build a Notebook [4,41,42]. Outputs of SQL commands of the SQL Script are automatically saved into temporary markdown cells. A few considerations: - If any SQL command has no output, a blank markdown cell will be instantiated on the output location.

5.3. Using Spark Notebooks

I stated earlier in our discussion that while Data Factory predominantly concentrates on pipeline computation, if you need distributed computer resources for more complicated tasks yet nonetheless want a managed service that handles services like Cloud Storage, Azure Synapse Analytics, or Power BI connection for you, the Fabric ecosystem provides the option of Notebooks, where as a Data Scientist you want to administrate clusters as infrastructure resources, and then access all those sources for your analysis independently of the security context of a single user or a service principal that pipelines usually implement. Notebooks, provided by the Lakehouse family of Fabric, will give you cores from the Spark ecosystem, and then bundles of Spark libraries for your project

according to the Python or .NET language you'll decide to program the Notebook. From a few samples over there, it looks like Spark Notebooks implements Python environments in Java – totally understandable as the idea of backend interpreter modeling execution environments in Java with the .pyc package, which internally uses the JVM built-in functionality for executing bytecode [43-45]. This means that for the Notebook domain that accompanies the Lakehouse, probably for performance, you will only have access to the Python environment available in the backend interpreter, and not the .venv feature that was natively implemented, later copied, and now available in the other interpreters. Besides this clarity for your project, and some libraries you may have to install again in clusters with Configuration settings using the Notebook kernel, once your Notebook is working, you may call loads of libraries, modules, functions, and classes like any module using Python import keyword, reducing dependencies –especially if your classes help any other Notebook that you or your organization have done in Azure– to the few ones in the Configuration settings.

6. Dimensional Modeling

A well-designed dimensional model is crucial to delivering effective business intelligence. Whether in a small environment, using the information in a spreadsheet or a large enterprise where most critical company decisions are based on the analysis of data in the data warehouse, if the data model is not designed in a user-centered way, it may cause frustration to the business community. They may not trust the data and lose the will to analyze it [9,46-48].

We want the users to easily find the information they are looking for and we want to understand their business processes in order to give them the proper guidance. Dimensional modeling is based on business processes. It is more than just a structure for the data warehouse; it is the actual representation of facts, metrics, KPIs, data, and information that the business uses to make decisions. Understanding the core processes of a company, what happens and how it is related to analysis is the first step to effectively model the data warehouse.

Unlike transactional systems that typically contain a large number of tables to track all attributes and relationships within the business, dimensional modeling relies on business metrics that can be described by a relatively small number of dimensions. Dimensional models are also directed to easing queries and enabling analysis. Reporting and analysis focus on aggregated data and dimensional

models typically rely on the de-normalization of the data to avoid joins. Data from each dimension is related to each measure (fact) and the dimensional model is built on how aggregation occurs.

6.1. Principles of Dimensional Modeling

Dimensional data models help the human mind quickly sort and classify event data for the purposes of query answering and drawing inferences. A dimensional data model consists of business questions, keys to locating data points that answer questions, and data attributes that provide the specific content that answers questions. The models are commonly visually represented with a table layout. Quick event lookup and content qualification are performed by enabling hierarchical database indexing and table pre-aggregation. This enhances query performance but sacrifices some data storage efficiency. Dimensional models are best suited for fairly static events since they require significant redesign and data reorganization to accommodate changing event schemas.

Fact tables are the main data components of dimensional models. Generally, fact tables contain the data qualities that describe the event measured by the table. Fact tables augment the derived statistics of stored fact observables. Fact tables are augmented with a key to each of the dimension tables associated with the fact, plus an occurrence counter. This structure "flattens" the dimension record linkages in a way that produces a denormalized representation of those schema aspects of the model. Dimension tables do not represent facts but provide qualification attributes for those facts stored in the fact tables. Dimension tables contain one record for each unique attribute value combination. Because dimension tables can grow large, a data storage design consideration for them is whether they should be partitioned into multiple physical tables.

6.2. Designing Star Schemas

Fact tables contain numerical data about the business such as sales, sales quantity, profit, etc. Then surrounding dimension tables contain qualitative characteristics of the fact table metrics such as product, customer, time, location, etc. Fact dimensions are usually low-cardinality and numeric such as currency, quantity units, etc. The dimensions related to the metric data in the fact table are usually the dimension tables that have basic attributes that we have seen so far. Such as product, sales location, customer, and time pill. Therefore, we must contemplate which dimension tables can contain the static attributes of the metric data, to create the dimensional schema.

In the star schema, a hub-conformed dimension, such as a product dimension table that contains its static concatenated characteristics, exists in separate

dimensions from the fact table. A hub-conformed dimension can give us any categorical information about any of the numeric metrics in the fact table. These can be the basic categories for products, customers, and sales locations which country, province, city, and postal code dimensions represent. These are time and date, hour, and year dimensions which are represented by the date dimension. Finally, the pill metric data which companies and organizations use as output. Their mini metrics or sub metrics, which are the basis for decisions because they can easily become measurable reports.

7. Star Schemas

A star schema is the simplest type of data warehouse schema. A summary data table is linked to one or more detailed data tables via foreign key-primary key relationships. The summary data table is usually around 100 times smaller than the detail data table but may be up to 10 times larger. Unlike some other types of schemas, star schemas are not recursive; the summary data table contains summary data, not summary pointers. After database creation, star schemas are relatively easy to add to because no complex parent-child branching or linking structure needs to be modified. Star schemas support simple SQL queries because the SQL queries usually only use a table join between the summary and detail tables. Contrary to some other schemas, star schemas are also efficient for data retrieval and are fast for data processing.

The point of a star schema is to make the retrieval of interesting subsets of multidimensional data a simple and fast operation via the SQL select statement. An interesting subset is a subset that is carefully designed to be accessed frequently. To understand the structure of a star schema, it helps to know how the SQL select statement retrieves data. The public interface for the data in a star schema consists of two very distinct parts. The first part is the detail data; the second part is the summary data. The detail data contains the information that describes the specific events that occurred, the ones we want to get answers from. For the stock data, the detail data describes what happened on specific days. The summary data contains counts of the events and other aggregates. Each summary table contains an entry for each value of the summary attributes on that summary table. For the stock data and the summary by day and ticker, the summary data contains an entry for each ticker. The detail data has a more complex structure than the summary data. Some of the detail attributes contain pointer values, which point to an entry in one of the summary tables. In the case of the stock data, the date and ticker form a composite pointer.

7.1. Structure of Star Schemas

A star schema consists of one or more large fact tables that are surrounded by several smaller dimension tables. Star schemas use different data types for dimension and fact tables. Fact tables typically draw on numerical data types such as integer, float, or decimal for their non-key columns, which track things such as quantity and sales amount, and though fact table keys can be defined using a variety of types, they are normally defined using a numeric data type. Dimension tables use character or date data types for key columns and use character or integer data types for non-key attributes. Dimension tables often use compact data types to limit size since dimension tables, while much smaller than the fact table, are not normally fractured.

One important constraint imposed by the star schema design is that minimum cardinality of join links between the fact and dimension tables is one; every value in the foreign key attributes of the fact table must correspond to a value in the primary key of its associated dimension. This means that every fact must be assigned to at least one of the categories or classifications represented by the dimension. Although this may seem a trivial constraint, for validation purposes it is actually quite significant; in a normalized data warehouse, it is entirely possible for a fact to have “no” category assignment. Some parts of dimension tables, such as satellite dimension tables in a type 2 slowly changing dimension, may have no corresponding facts. From a design perspective, the category “common for all” enforced by a star schema enables designers to better understand the data and how they are presented.

7.2. Benefits of Star Schemas

Star schemas provide a simplified view of the business data structure, making it more intuitive to business users. With interesting data content in fact tables supported by referenced dimensions, the user has a selection of refined dimensions to narrow down fact results of specific interest. The presentation of data to users in a simple format can help them to decide what data they would like to analyze. Once the data selection has been done by the user, there are many advanced analytic options that a company can offer back to them. Some of these options are multidimensional cubes, dashboards, and reports using advanced analytic techniques. Business needs to understand that a simple well-organized data mart presents a lot of possibilities for the user. Star schemas also speed up query performance. With the business community users and the advance applications standardized around important fact tables and key dimensional structures, there is an ability to tune performance for these common intensive queries. With a good indexing strategy plus other database tuning techniques

such as denormalizing fact tables, creating materialized views, partitioning, etc., these fact tables can be designed for maximum performance for the desired type of query. Some databases have boosted the query speed by using the star-join feature. This feature helps the database territory join the fact table with multiple dimension tables when needed in an ad hoc query. After tuning, the common star join must be the fastest database join second only to the special case of a database merge join.

7.3. Best Practices for Implementation

Start by analyzing the business user requirements, data source volumes, and data sizes that will be imported into the star and data marts. Understand the type and quantity of new data that will be imported and how often the data will be updated before designing the logical and physical aspects of the star schema. Assess the resources to minimize the impact of refreshing the data with your organization's business operations. You usually run data import and processing jobs outside of normal business hours. Typically, dimensional models with frequent changes, such as daily sales fact tables indexed by location, are usually contained in a data mart rather than the enterprise data warehouse.

If the volume of updates and inserts does not negatively impact performance, star structures can be incrementally populated throughout the day. First, create the dimensional and measure tables. Create policies to determine how much activity can occur on these tables without creating contention with the source. Use the policy to schedule updates and inserts for each table. In some cases, you can populate the tables with a complete refresh and create them as empty tables. Old data can be created whenever you merge or stage data. However, to merge, create a journal table for appending records. Creating journals speeds up the process because they can be inserted after the period is completed. Each execution of the process after the execution of the previous can be factored into incremental increases in the fact table.

8. Data Integration Techniques

In its simplest form, data integration is the process of combining data from different sources. Data integration techniques are used in the preparation stage of a data integration workflow, often called a data pipeline. The preparation stage is defined as the process of ingesting, cleaning, transforming, and loading data into a data warehouse or other analytics-friendly destination. The technology used to support this integration process can vary widely. A modern data architecture

typically uses batch and streaming ingestion services, both of which can trigger event-based processing via orchestration tools. Some pipeline features, architectures, and design patterns are general, while others may be more specific, especially in a cloud native context.

Extract, Transform, Load and Extract, Load, Transform are two popular techniques for data pipeline implementations. Both ETL and ELT copy data from data sources to a staging area, then clean and transform the data as needed, applying business rules and correcting for errors. Previously, these techniques were typically called warehouse loading techniques, but as data warehouse microservices have been released that offered serverless compute and storage architectures, these two upload strategies have started to be incorporated into more general service ecosystems, at larger scales and at finer frequencies than traditionally conceived of enterprise-scale data warehouses.

8.1. ETL vs ELT

Processing and modeling data is the final step of getting the data ready for insightful reporting and advanced analytics in the Azure Ecosystem. Typically, there is a lot of focus on loading the data from various data sources to the data warehouse or the data lake. However, the choice of data processing and modeling techniques can significantly impact the performance and cost of query workloads that analysts and data scientists run for business insights. This section will explore the different techniques available in Azure along with the pros and cons of various techniques to help you choose the right one for your needs.

ETL and ELT represent the ways data can be processed before loaded into a modeling system. ETL was the preferred architecture for many years. A traditional ETL architecture uses a staging area to transfer data from different data sources and an ETL tool to extract, transform, and load the data into a target system. These ETL tools include various options. As the enterprise data landscape expanded, companies began to require wide data availability and integration for constant analytics building. ETL had two major drawbacks: Data Availability and Latency. Data Staging cuts down the speed and frequency from which data can be loaded into the analytics platform. The data also cannot be monetized until the ETL load cycle is complete. ELT addresses the above two drawbacks. Once the raw data copy from various disparate data sources is available in a data lakehouse, Data Engineers can open access to the data for various stakeholders like Data Scientists, Data Analysts, and Business Users.

The availability of cloud data warehouses or data lakehouse has changed how data should be transformed. With ELT, data is transformed at the end or as

required by end users rather than pushing all data into an analytics platform. This means that the analytics platform is never behind on the latest data available. Because cloud-based data platforms can easily support many concurrent users. Customers don't want every different kind of processed data copy. They would prefer to store a single copy of the raw data available rather than at least multiple copies of processed data in the ETL architecture.

8.2. Data Pipeline Design

The emergence of cloud engines that natively support the processing of big data and the introduction of Consortium-based approach in data integration turned the stage of building Data Consolidation and Repository platforms upside down. The "model to deliver" has become a "deliver then model" approach: build the pipelines based on business consumption and deliver the data as raw and fast as possible and postpone data quality metadata modeling and business-level dashboards and reports implementation for a later date in the BI ecosystem lifecycle. In "classic" Data Warehousing, the ETL processes had the BI Application layer as the sense of the universe from a Data Consolidation perspective. The data extracted from upstream entities was rarely recycled without the involvement of IT. Today in the Big Data Era, the Data Lake is becoming a central hub of corporate information. For most enterprises, the Data Lake is a space to land Data from "Production" Systems or Third Parties in a non-structured way. The Data Lake consolidation comes as a centralized reservoir to harbor corporate information in its most heterogeneous forms and usages. The Data Lake often leverages on the availability of big data engines. The Product IT Infra structures for Big Data have all the characteristics for becoming the corporate Data Lake: pricing per volume, ability to store data in multiple or no structures, consumption at the second. Organizations have the opportunity to build a Data Lake with the characteristics of the data warehouse of the 1990s and this can be done for a fraction of the cost: store the corporate data in its pure state in the corporate Data Warehouse and leverage on horizontal price pressure and commodity infra structure-based on non-structured data for economic exploration such as volume analysis and data mashups.

9. Performance Optimization Strategies

Besides ensuring correct execution of data processing and modeling tasks, an important aspect of working with data in the Azure ecosystem is performance. Processing and querying large petabytes of data and optimizing the orchestration of different tasks for a data pipeline is crucial for reducing costs in terms of time

and cloud service usage. In this chapter, we highlight different strategies for optimizing power of Spark jobs in Databricks – an important and popular tool for developing data pipelines in the Azure ecosystem. We also describe how to improve the performance of querying and accessing data stored in Data Lakes.

9.1. Optimizing Spark Jobs

There are various design-phase considerations for maximizing performance of a Spark job. The most common include avoiding shuffling of data across partitions since this leads to heavy disk and network I/O. This involves minimizing the number of joins that require data to be shuffled and stragglers being created by uneven partition sizes. Considerations that can be performed in the implementation include ensuring that input data for workers is appropriately sized in terms of partition sizes, correctly using persistence and caching functionality of Spark, ensuring that Spark transformations avoid creating large data intermediary files, and configuring parameters such as memory and CPU cores that balance cost of configuration against execution speed. In addition to these considerations, performance bottlenecks can also be diagnosed from measures reported in the Spark UI. There can be tasks that take orders of magnitude longer than other tasks, which lead to extended execution time of the job, as well as common stages across multiple jobs, indicating that those stages can be cached for specific jobs.

9.1. Optimizing Spark Jobs

Data processing requires a great deal of optimization efforts to enhance performance. Developers need to apply strategies at every stage of their development lifecycle. One way to introduce performance variations is to deploy workloads in different velocity thresholds. Some services expose a threshold parameter to influence the acceleration in query performance and the added resources. One is the parameter that developers can set to optimize job performance. Some additional specific settings to consider for performance include other parameters. Job and query performance can vary a lot depending on the velocity at which the user wants the results. These velocity considerations naturally need to be discussed in conjunction with the cost associated for improving the performance. However, there might be certain critical paths, such as a recommendation model that needs a low latency, which would need to negotiate costs at such times.

The parallelism should be tuned based on the size of the task and the degree of resources capable of loading that task. However, there are certain optimizations that require no additional deployments or high-level programming to achieve

considerable performance boosts. The key optimizations that are generic and language agnostic should be performed. Most of these optimizations use basic concepts. DataSkew is a commonly seen issue that could cause multiple driver and executor failures. It causes some partitions to become significantly larger than the others, creating partition imbalance. DataSkew mostly originates from the join operation and could affect the join performance. Other operations such as group by, reduceBy, and aggregateBy are also occasionally susceptible to DataSkew. Another big consideration for addressing DataSkew is to review the Execution Plan. A visual dataflow representation of the computation behind the execution of a job is provided.

9.2. Improving Query Performance

In modern data analytical workloads, applications keep changing based on how enterprise and business decisions keep evolving. From changing business decisions to seasonal changes, every application is bound to achieve good performance and what dictates the important performance is the SQL analytical queries. The top challenges faced by any analyst who is developing SQL-based dashboards or applications is to liberate the power of SQL statements to achieve faster response time. Such responses are not only needed for SQL analytical dashboards which require data to be near Real Time but also help prepare reports on medium-term basis as well as on Long-term basis and are usually read-based data models. The performance of analytic queries is critical because it controls the speed with which data is ingested, processed, and reported on. The main driver for optimizing analytical query performance is time. Long latency is a consequence of complex performance optimizations. A range of philosophers and business leaders have commented on the nature of time and its value in human affairs. In our case, the underlying theme is that time governed by the statement pushes cloud analytics toward a high level of hardware and system efficiency and a corresponding demand for response time determinism in query processing.

In this section, we explain a collection of query optimization techniques that may be done with the data in motion or at rest. The optimizations of data at rest are performed at the level of data warehouse products. The optimizations for data in motion are done at the level of streaming analytics engines. The analytical query performance optimizations are divided into two types. 1. Techniques that are done to analyze SQL query running plans to improve performance on intermediate changes. 2. Techniques done to logical or physical optimization of models or schemas of the datasets modeled to achieve analytics optimization. We detail these techniques preferably in tabular form in the following sections.

10. Data Governance and Security

10.1. Data Governance Framework Data democratization can reduce workforce silos, bringing together operations teams, data engineers, and data scientists around the goal of obtaining faster insights from the data. At the same time, the number of queries run in the data platform increases, and the need for a solid data governance framework to manage the organization's data increases as well. Good governance policies make all data accessible to trusted users while managing privacy and compliance requirements. These services help customers create a scalable, comprehensive governance framework across the entire analytics lifecycle. You receive classification of sensitive data stored in data storage to simplify compliance and privacy programs. Built-in labels help protect sensitive data wherever it resides, and you can also classify and label such data across on-premises databases. With the help of a centralized access control model, you can create a fine-grained access control model across all data assets.

Allows you to discover, catalog, and classify the different entities that are within the data platform and that are produced and consumed during the data pipeline run over the data flow, including external data sources or services. It also allows you to implement lineage tracking of the data, helping you to better understand how your data is structured, the different transformations applied, and the associated quality. Resource monitoring and alerts are used according to the different services offered status.

10.2. Security Best Practices Security is among the principal topics to take into account when designing and implementing a data platform. Data breaches are the most exposed topic in industry threats in the world today. Security is usually addressed at different layers, meaning starting from the network level, down to the application and data levels. In the case of a data platform, security services offered by cloud providers are usually implemented at the network level. Customers are responsible for the security of the application logic and data levels.

10.1. Data Governance Framework

Over the past decade, data management has become one of the key areas of success for both technology companies and traditional industries. The evolution of so-called "big data" has increased the amount of data generated or stored by organizations, which has become one of the assets with highest impact for companies. These organizations need to be guided by regulations, laws, and risk management to make the most of their data. Given the speed at which technology evolves, how it affects the way organizations work and generate business

intelligence, many companies are left without the necessary structure to ensure that the value generated from data is higher than the risks associated with it. In order to mitigate some of these risks, organizations have been implementing data governance.

More specifically, all employees need to understand what data is available, who is responsible for its quality, what the data is used for, and what metadata properly describes it. It is also necessary to assign data stewardship tasks, governance committees, data strategy leaders, and data stewards in each organizational pole. Then, data governance needs to be prioritized and be part of the organization's culture. Technology will be a great ally in the implementation of data governance, as it provides the necessary tools to facilitate data quality assurance, data management, usage monitoring, security, and training campaigns. Many times, vulnerabilities discovered in production environments are a result of misconfiguration or lack of a proper data governance policy. In recent years, data breaches have cost companies billions of dollars, and many data leaks or damaged reputations could have been avoided if data governance had been implemented properly from the beginning.

10.2. Security Best Practices

Data is the new oil. As such, it must be safeguarded, and best practices for security should be implemented in protecting data during its journey through the data lifecycle. While tenants inside a cloud must have adequate separation in the physical layer, it is the responsibility of users to ensure that the layers higher up in the stack are also secure. Mother Nature's inability to produce blank checks for everyone requires that the cloud service be cost-effective while still remaining secure. In this age of social media, where access to your data can make the difference between success and bankruptcy, it is imperative that proper data security measures are in place. The building decision is often driven according to ease of service rather than data safeguards. So, the best practice is to think of these services as a data offering and a data consumption offering. Only by cloistering the data and only speaking through these services can the organization hope to remain secure. Some services can inherit the storage Account Keys or generated identities. Other services will require the organization to be vigilant about ensuring that services are not able to read/write to the blob container directly, and that the only channels/routes are through the services.

While the guards are mostly put in place at ingestion or consumption points, it is important to also monitor for unauthorized access on the detection side. Monitoring lets you set an alert when personnel get too or too little involved in activity that is otherwise anomalous; i.e., every day a particular user logs in at 9

am and logs out at 10 am, and suddenly on a particular day, the user logs in at midnight and does not log out — set an alert for that. A service built upon the platform can be easily configured to detect such anomalous activity, and it is important for organizations to make use of the available monitoring solutions.

11. Case Studies and Real-World Applications

The concept of the Azure-based data processing is implemented through the Azure Data Lake, the application of which is illustrated in two case studies from different domains: retail and finance. Both enable you to ingest data from many disparate sources, including on-premises sources, load business-critical data into the Azure Data Lake, and get timely insights for better business outcomes. This chapter focuses on providing a rich experience of data processing in the Microsoft Azure Ecosystem—the Azure Data Lake and its services. The usage of the Azure Data Lake Store—one of the core components of the Azure Data Lake—is illustrated through two case studies from two domains: retail and finance. Both enable you to ingest data from many disparate sources, including on-premises sources, load business-critical data into the Azure Data Lake, and get timely insights for better business outcomes. The implemented data processing steps include data ingestion, data generation, data transformation, and data discovery, among others, using complementary Azure services, such as Azure Synapse Studio, Azure Blob Storage, Azure Data Factory, Azure SQL DB, and Azure ML Services. The completed data processing project can then be published to Synapse Studio for future use or to support business reports for enterprise decision-making. These two examples of rich experiences prototype Azure Data Lake Store interactions, from creating a source data pipeline to preparing the data and loading it to ADLS for future use, including those needed for enterprise-level business intelligence, as well as for machine learning-related activities.

11.1. Case Study 1: Retail Data Processing

Building an effective data processing and modeling pipeline is essential for deriving value from the myriad of data available in today's ecosystem. In this chapter, we will explore specific examples of pipelines built using cloud technologies for complex analytics, predictive modeling, and recommendation problems. We will focus on online click prediction and online item recommendation for a large retail/ecommerce company and prevention of large transactional losses for a large banking and credit card services organization.

Everyday, consumer retail stores collect vast amounts of shopper behavior data from transactions, loyalty programs, and more. This data is then stored in questionable unstructured form in public clouds. Such clouds can be accessed directly from business intelligence and visualization tools. However, given the inhomogeneities in the data distributions across these retail stores for different items, these first-pass visualizations may not yield any noticeable market structure. This was the original motivation behind the recommendation system design and implementation at a major North American retail chain. Despite the wide variance over geographic regions, shopper cohorts, loyalty program rules, promotional events, retailer value propositions and item categories, how do you match the right electronic and email marketing communication for a specific shopper? Simply put, how do you combine the right shopper, product communication, and timing to re-engage the shopper at the time of intent? By increasing the focused accuracy of at the right moment marketing, retailers can not only decrease their marketing spend but also increase their revenues resulting in greater EBITDA margins.

We have multiple databases that warehouse store transaction and item level data to analyze customer shopping patterns and predict their responses to marketing at various times corresponding to their assumed shopping lifecycle states. Additionally, the reasons for product stockouts and the exact timing for restocking based on predicted demand patterns can be constructed. The online product recommendation system is one of the key systems used for accelerating the in-store restocking process using principles of supply chain optimization.

11.2. Case Study 2: Financial Data Analysis

Today, financial data plays a more important role in developing new solutions. Different financial data analysis helps in understanding various ways of forecasting, trading and investing. Financial data also helps in analysing historical incidents or patterns and lays down certain strategies and policies which if followed, can be beneficial. Thus, various services can be used to analyse and forecast the financial data and prepare a report. The dataset has thousands of records and has been modified. The data used is CSV for as well as. Storage is used for storage and input for and.

is being used for documentation, for data preparation, cleansing, model training and scoring, for Hyper Parameter Selection, for final Report Generation, for Data loading, Preparation and Moving Stage and finally, for model training and scoring. The architecture helps in understanding the flow of data with the respective services and gives a brief overview of how each service works and is

interconnected with each other, thus preparing a beautiful and light-weight report with more accurate output.

The solution is highly scalable and can also be made more automated and cost-effective using different services such as and. The required output depending upon what is to be forecasted. This particular solution would help practitioners in the domain of financial data analysis better understand the model used and help in further improvements as well as practitioners in the field of data science understand the different services and their uses when working with large volume as well as variety of data.

12. Future Trends in Data Processing

Data processing is not a static area of research and development. It continuously evolves to provide better performance, lower costs, and solve challenges from increasingly complex business use cases. Artificial Intelligence and supercomputing with fast memory and interconnects are emerging trends in data processing and they apply to both big data and classical data processing problems. With respect to big data, the most relevant emerging technology focuses on improving cost and performance efficiency with special purpose hardware and architecture or specialized protocols to speed up communication between distributed nodes. A second line of innovation focuses on improving the user programming experience for the explosion of new big data frameworks. Examples are new APIs that simplify the orchestration and usage of heterogeneous big data frameworks that ease user ceremony or frameworks and interfaces that combine model training and inference under the same programming interface. A third line of innovation simplifies provisioning and orchestration of performance clusters in the cloud or hybrid environments. Examples are new serverless abstractions, clusters, related services, etc.

Besides big data, classical data problems, such as tabular data, are also attracted for innovation, with focus on the design and programming experience of ML and its optimized infrastructure implementations. Algorithmic innovation is the deepest driver of what works in ML, paving the way for new algorithms that greatly improve prediction performance or for de facto industry standard algorithms that improve prediction accuracy and simplify programming, like low-code solutions. However, algorithm improvement is not the only kind of innovation, improving user programming experience and generalizing predictions APIs is also important. Such innovation has to do with designing

elegant APIs shared by libraries that provide general-purpose solutions. But library APIs are not the only ones supporting user experience with ML code. There are properties that improve prediction user experience across different models, like specifying a schema for your input data or having the framework automatically create train and validation splits and remember them across execution runs.

12.1. Emerging Technologies

The amount and diversity of data produced by humans and machines is at an all-time high. In order to extract value from this data, its owners have to employ advanced and intelligent systems. The advancements in Artificial Intelligence and Machine Learning, and the emergence of interesting technologies to process, analyze, and model data in various formats are enabling this. In particular, we highlight three technologies that have impacted Data Processing and Modeling workflows: Edge Computing; Large Language Models; and Responsible Data Initiatives.

Emerging Technologies

The application of Artificial Intelligence and Machine Learning has predominantly been to manage the more traditional type of data that is produced mainly from business processes, i.e., structured data. However, the amazing progress in Computer Vision, Natural Language Processing for text, and for Speech and Audio Recognition now allows organizations to unveil value from all kinds of data. Organizations are beginning to realize that they can analyze and model more of the data they already own or have access to, leading to new insights, predictions, and recommendations.

In addition to society, what is driving this expansion of data are the increasing number of devices producing data and capturing knowledge. These devices include smartphones, drones, satellites, cameras, sensors, and industrial machinery. Even more important is that these devices are constantly producing data. The simple act of clicking on a link to read the news, or producing save an email, an image, or a purchase on an e-commerce web site leads to data being generated in different formats and types. Devices are at the edge of the data management and processing workflows, and therefore the execution of some of the data processing operations, such as validation, cleaning, and preparation, need to take place closer to the point where the data is produced.

12.2. The Role of AI in Data Processing

The amount of digital data we generate when we do something in the universe is large and is growing rapidly. Generally speaking, we can scale it as the volume of data is proportionate to a factor size that tells how complex our reality is. This is the price to pay to build digital twins of the physical, biological, and human worlds. The amount of data generated definitely looks larger than the mass of dark matter in the universe. Yet, it is full of gaps and imbalances. This ever-increasing volume of data requires new methods to process data and reveal models that allow us to control our digital, and eventually physical, action. So, the trends we have seen so far in segmenting the data processing disciplines along functional lines will gradually evolve into a trend towards consolidation, along increasingly intelligent lines, of data processing activities, with data as core.

We will be able to unleash the true potential of data-driven decisions and action because of the emergent capabilities of AI as a mix of algorithms, techniques, and systems that are able to extract knowledge from data, both structured and unstructured, at a speed and efficiency higher than that of single humans and of groups of humans working together. AI will enable the automation of knowledge extraction from data across the board. This trend will require, on one hand, the standardization of data collection and preparation, model training and prediction, and result validation and integration steps, and, on the other hand, supervision and governance methods to integrate human knowledge in the activities that require it.

13. Conclusion

Over the years, we have dealt with different types of data processing and modeling systems and tools in the industry applied to the different types of analytics areas such as Big Data, Data Warehousing, or Data Science. Each analytics category has specific infrastructure, services, and tools to support the different patterns, features, goals, and objectives. Azure provides a comprehensive set of ecosystem services, capabilities, and support to cover all types of analytics areas, and is still broadening its capabilities through continuous integration for new tools and technologies. The goal was to demonstrate the breadth of Azure services and tools for building data processing and modeling pipelines.

There are different ways for architecting the data processing and modeling pipelines. In the analytics journey, it is important to define the main pillars that

will influence and shorten the time to production such as experimentation, scalability, reusability, automation, governance, and cost. In practice, there is not a specific answer to how to balance each pillar in a data project. Each of those pillars needs to be analyzed according to the specificities of the business problem being solved. The decision will define the strategy for how to build the data pipeline, selecting the right technology components with the specialists available in the team that should match the complexity of the problem being solved. The main Azure ecosystem services, tools, and technologies for different areas of the data pipeline journey are presented: ingestion, data preparation, data processing, data modeling, and deployment. The objective is to provide a single point of reference for the main services and capabilities exposed by the Azure ecosystem.

References

[1] Dzulhikam D, Rana ME. A critical review of cloud computing environment for big data analytics. In2022 International Conference on Decision Aid Sciences and Applications (DASA) 2022 Mar 23 (pp. 76-81). IEEE.

[2] Emami Khoonsari P, Moreno P, Bergmann S, Burman J, Capuccini M, Carone M, Cascante M, de Atauri P, Foguet C, Gonzalez-Beltran AN, Hankemeier T. Interoperable and scalable data analysis with microservices: applications in metabolomics. Bioinformatics. 2019 Oct 1;35(19):3752-60.

[3] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.

[4] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.

[5] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.

[6] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.

[7] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.

[8] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.

[9] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.

[10] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.

[11] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation

in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[12] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[13] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[14] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[15] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[16] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.
[17] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[18] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.
[19] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.
[20] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.
[21] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[22] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[23] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[24] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[25] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[26] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[27] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[28] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.

[29] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[30] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.
[31] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[32] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[33] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[34] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[35] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[36] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[37] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[38] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[39] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.
[40] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).
[41] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[42] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[43] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[44] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[45] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in

Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.

[46] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.

[47] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.

[48] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 6: Performance Optimization and Cost Management

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction to Performance Optimization

Successful businesses rely not only on investments and predictable cash flows, but also on their IT systems. Almost every business's services, clients, and partners are intertwined with its IT system operations. Thus, the performance of IT systems is directly related to business productivity [1,2]. Over the past years, significant operational time and cost savings have been accomplished through IT performance optimizations. IT performance optimizations can lead to operational time savings of 80%. This drastic reduction in wasted time can convert directly into dollars as labor can then be allocated to "dollar productive" processes.

The process of optimization refers to making the best use of resources to achieve goals. Businesses reach varied performance and financial targets, and resources are allocated to use in achieving the desired targets [3-5]. Some companies want to answer calls in 5 minutes; others are satisfied with answering in 30 minutes. One company may want to generate an outsourced account statement in 3 days, but another in 30 days. Each activity has an associated allocated time and cost. The concept of optimization means either providing great performance at a low cost or great performance through high spending. In general, large organizations with high processing loads demand great performance achieved by high spending. Others prefer low spending for scaled performance. Thus, performance optimization is the way to turn the knobs to achieve the desired goals.

1.1. Definition and Importance

1. Introduction to Performance Optimization

Performance Optimization is a discipline striving to drive the key performance indicators (KPIs) of given processes, buildings, or even production plants to their optimal values with the help of design changes, modifications, or refinements. In other words, Performance Optimization is a formal approach used to improve solutions given by production simulations or similar applications, mostly undertaken without prior knowledge of where such enhancements can be found. Performing such optimizations is of high importance whenever business systems, for example, are designed and developed [2,6]. Poorly optimized systems can create excessive and unwanted costs and poor customer satisfaction through delays and negative impacts on the product quality. Luckily, there are several typical industries where performance optimization has been extensively used and developed over quite some time, and it has proven to be of high benefit to those industries having high safety and quality demands. However, there are challenges associated with performance optimization [7-9]. While general mathematical optimization techniques lead to stable and practical solutions for many applications addressing a wide area of research, business logic makes it difficult to find and implement optimal solutions for company-specific production tasks such as planning, scheduling, and controlling.

In this chapter, important issues of Performance Optimization for making it a standard equipment for the design and development of production planning, scheduling, and control systems are outlined along with solution strategies developed in the project. With thousands of users of commercial simulation software in the world but only a few users of process simulation based Performance Optimization packages, and the few Advanced Planning and Scheduling (APS) systems currently on the market not making use of optimization techniques, the development and implementation of formal optimization techniques for discrete event dynamic systems and concurrent process control has not yet reached a level that seems to be commonly accepted by the simulation software companies and the user market.

1.2. Key Concepts

This section presents key concepts about optimization, availability, and cost management, and discusses how the basic measures of speed, efficiency, cost, and effectiveness are related. Collectively, these concepts provide a foundation to guide the design of a cost-efficient and high-performance computer system, and present system cost management and performance optimization issues and solutions within the appropriate context.

1.2.1. Speed and Efficiency: What Are We Trying to Optimize? Optimization is an often-misused term. In engineering and economics, the term usually refers to

maximizing or minimizing a specific metric of interest, which is termed the objective function or optimization criterion. System builders use this criterion to determine an optimum or near-optimum allocation of limited resources.

1.2.2. Cost Management: What Costs Should We Minimize? In general, "cost" encompasses all the measures of resource use mentioned above: monetary, energy, and time. In the case of performance optimization, we may choose to reduce any of these measures alone, or combinations of them, based on the situation. In certain cases, it is appropriate to minimize only one part of the total cost function. However, in many other cases no such simple division is possible, because of the interrelations between the various costs.

1.2.3. Effectiveness: Are We Solving the Right Problems? "Effectiveness" refers to the design of a computer system that meets the needs of its users. In many cases, maximizing the system effectiveness may involve utilizing resources inefficiently. However, there are also times when there is a strong connection between efficiency as well as overall system effectiveness. Performance optimization at the "lower" level should not negate effectiveness at the "higher" level.

2. Cost Management Fundamentals

Cost management is an ongoing strategic process concerned with developing and monitoring cost standards ensuring that all costs are reasonable, allowable, and allocable and that the avoidance of unanticipated costs is built into the project from the beginning and throughout its life cycle [10,11]. Unlike cost estimation, which typically is limited to scheduled, planned, or expected milestone dates, cost management is a proactive process of real-time analysis and review for every milestone. It examines readily available project information throughout the project life cycle to identify development, implementation, and cost performance issues where actions, or corrections, can be taken to avoid and control excessive or unacceptable costs. It employs project reporting to facilitate communications concerning cost/schedule variances and related issues among project stakeholders. Understanding and managing costs are among the major issues besetting project management. The who, what, where, when, why, and how of project work requires spending money. Sooner or later, and most logically sooner, an account must be made of the resulting costs against the original forecast [12-14]. Cost management policies may vary among types of enterprises and levels of management, as may the emphasis on various decision factors, such

as cost versus utility, service, and appearance, risk and timing of occurrence, and estimated versus actual costs. But at the element level, important day-to-day operational decisions must be based on the timing of estimated costs as compared with the current and projected cash flow.

2.1. Understanding Costs

An important aspect of performance management is understanding the costs associated with delivering an output. In the common example of manufacturing, the costs are generally obvious — there is a straightforward breakdown of the cost of parts, labor time associated with assembly and distribution, and any other ancillary costs. In executive payments, the principal cost generally is uncomfortable but not difficult to calculate. Unfortunately, while computations of costs associated with the delivery of outputs may be simple, they can also be far from trivial, particularly when applied in a large corporation that applies a fully distributed cost model. In such a system, an appropriate division, or assignment, of costs to output of corporate departments and groups creates arcsin curves for many elements of group performance [3,15-17]. These curves relate outputs and performance. Inputs, and their associated costs, determine the overall corporate performance profile.

In this section, we will discuss just two types of cost calculations that are frequently applied in corporate environments to various forms of capital allocation. These two calculations, average cost and marginal cost, can appear to be similar but, in fact, are used for subtly different purposes. While our focus is cost management, we will discuss how resource consumption and production techniques interact with corporate performance and output requirements. Indeed, a complete understanding of corporate performance management requires a detailed analysis of the interaction from various perspectives between performance and the costs of inputs applied to resource consumption, goods provision, and output generation.

2.2. Cost Analysis Techniques

Hardware performance and cost are commonly investigated with cost–performance analysis. When communication and memory subsystems are included, cost–performance analysis becomes complex. For example, using a parallel processor a structure may be stored with one access, but with a serial processor its structure may require stack management and lead to 1 or 2 access cycles per program unit function. Cost analysis has substantially ignored the issue of advantages obtained through more complex systems [18-20].

In simple systems like DRAMs, performance can be fairly accurately predicted from a small set of parameters such as number of bits, access time, and power consumption. With complex systems like microprocessors, relatively good performance estimates can be obtained from sets of empirical constants obtained from measurements and statistical analyses. In both classes of examples, cost may be predicted from mounting area cost and packing technology, gate or memory cell area requirements per function, and the relative area and cost use of nonfunctional aspects like packaging and dissipated power. This is simple, not naive, because cost and performance generally vary independently in strongly asymmetrical fashion with changes in number of pins or modules.

Cost estimation models provide cost estimates using various level input descriptions. Different cost estimation models are available based on various techniques. Because of cost vs performance trade offs, to be most useful cost estimation models should also provide estimates of performance and other characteristics such as power. Many estimates use summary parameters defined at component levels and should be accurate for most type of components. These types of estimates have been developed and incorporated into many typical component cost and performance programs. Whenever possible data from specific vendors should be used to obtain initial estimates. Component level models are most accurate if used for the technology they were developed for, although they can usually be adapted for other component technologies. Also, they should be applied to complete subsystems rather than single elements of the subsystem.

3. Strategies for Performance Improvement

Performance is a primary concern for virtually any enterprise. The drive towards performance improvement is undertaken for various reasons. First the enterprise may want to pursue greater revenues, profits and market share as an end in itself [21-23]. This is typical in the competitive environment that most enterprises are located: each enterprise attempts to become “the best in its field”. A second reason for pursuing better performance is the need to survive. Historically, many enterprises have been forced to change the way they operate to reverse a downward trend. Usually, this involves slashing costs and laying-off workers. The recent trend towards out-sourcing activities indicates that optimizing enterprise processes, in order to enhance overall performance, is becoming vital even for high-performing firms. A third enterprise-wide reason for greater

performance is the ever-increasing burden of regulations and statutory requirements imposed by governments, at all levels.

This drive towards greater performance, increased revenues and reduced costs is manifested in several changes currently taking place in the way enterprises compete: deregulation, privatization, globalization, empowerment, innovation, and other changes. There are three primary strategies available to improve performance of an enterprise. One approach, using methods derived originally from manufacturing and operations management, is to attempt to optimize all of the business processes that support internal and external customer needs. A second tactic is to improve the systems, the tools and technology, that enable the enterprise to perform its business processes. The rationale for improving these pieces of technology is that there are industry benchmarks, metrics, and best practices that can be adopted. The third way to improve performance is to try to achieve greater effectiveness of the output of the business enterprise center on the customer. This process is different from optimization in that process areas are the focus, not just individual functional areas.

3.1. Benchmarking and Metrics

Before pursuing any improvement efforts for a process, it is critical to take an inventory of the current expected and actual performance. There are two primary activities that help us gather the necessary data: creating metrics and benchmarks. Metrics and benchmarks are a critical part of any performance optimization strategy. The information they provide on a process allows the task manager to make informed decisions about which process features determine the expected completion time for the task, as well as which optimization strategy could best reduce that time. Metadata collected during task execution enables performance optimization approaches to dynamically avoid predicted bad performing scenarios, without having to resort to conservative solutions that prolong task execution most of the time. Benchmarks are a set of widely characterized representative inputs and their task performance profiles [9,24,25]. They allow task managers to know a priori how a task is expected to behave with the current platform having a particular configuration. On the other hand, metrics are measures that task managers can collect while their tasks are running and that characterize task behavior. These metrics can potentially allow task managers to accurately predict how any task is going to perform any action, no matter how different that action is from those included in the benchmarks. Metrics are useful when there is no performable benchmark for a given task; or worse yet, when there is a performable benchmark available that is worse than something

predicted by the metrics for that task on that specific platform with that specific configuration.

3.2. Process Optimization Techniques

The majority of high-traffic web applications are architected as a stack of various components. When investigating performance issues, the immediate question to ask is: which component is the culprit? Simple back-and-forth request tracing would instantly reveal the component responsible for the delay, if we consider it independently from other components. Unfortunately, many components are fairly asynchronous in nature and the coordination overheads incurred in distributed communication may obscure the bottleneck.

Increased hardware speed and easily available scaling resources both provide organizations with tools to eliminate performance issues. These are costly options, both in terms of time and financial costs. Therefore, it is natural that human trends in many disciplines have tried to start with changing how effectively a system uses the resources currently available to it, and to choose to only add additional resources once there are no longer significant gains available from trying to change the process [26-28]. Creating initial baselines to apply a process approach is itself enabled by technology - existing monitoring tools can be supplemented by easy data acquisition tools, or logging of control parameters.

Inside many organizations, performance metrics that can provide insight into the appropriateness of the balance between process optimization and resource optimization. Assuring that focus is on completing as many user interactions as quickly as possible will tend to build an organizational inertia to further effects towards full utilization of the resources currently available. Ideally, a user interaction should be completed whenever it is dependent on no resource and no piece of specialized code is tied up with a different user interaction.

3.3. Technology Upgrades

Performance optimization is continually being redefined by the principle that the number of transistors that can be manufactured on a chip is doubling every other year. A consequence of this is that system bottlenecks typically shift over time from CPU computation and memory latency to disk latency and I/O contention. Another consequence is that the cost of a specific performance upgrade is reduced over time, leading to the possibility of refreshing servers and storage with new hardware technology. Vendors of database and application server products, as well as vendors of enterprise software, are continually improving their offerings. This leads to the conclusion that carefully selected and timed technology

upgrades to better hardware and better software can lead to major improvements in a database application's performance [6,29-31].

Technological obsolescence occurs at an accelerating rate. Fortunately, continued scale reductions in microelectronics technology lead to serious performance improvements every few years. Upgraded hardware technology often allows database users to affordably accomplish what they were doing before, more cheaply or more quickly, with lower quantities of hardware. The high cost of database applications, in terms of the amount of hardware required, the resources they consume, or their response time or turnaround time per job, has led most users to a policy of "refreshing" their systems with new hardware technology every few years. Upgraded software can also lead to performance improvements, due to reduced access times from the underlying hardware or algorithmic improvements in the software implementation [32,33]. What follows is some advice on how to successfully pinpoint the expected contributions of technology upgrades, refresh your system, and determine the timing of the change.

4. Cost Reduction Strategies

To compete in an increasingly competitive market, organizations are seeking performance improvements through a variety of means, including reducing product or service costs in order to enhance profitability. Reducing costs is the primary focus of every organization, irrespective of the nature of business and the sector to which it belongs. Cost control and reduction is a vital function because it ensures a reasonable rate of return on investment to contribute towards the development of society. Furthermore, during recessionary periods, it is vital to eliminate waste, inefficiency and lower excessive production costs for products or services. Such measures increase the chances of survival. Costs are defined as an outlay of money that is a result of a business activity [34-36]. Cost control is the practice of identifying and reducing expenses to increase profits. Cost reduction, on the other hand, is the process of decreasing the expenses for a company without affecting the product quality. Cost reduction is a specialized approach that looks to reduce the cost base of the organization.

The purpose of this study is to assess the practical application of various cost reduction techniques for their effectiveness in helping organizations reduce costs in a way that supports their long term strategies. The process is not only concerned with cost control but seeks only to enhance profit margins by

maximizing the difference between costs and revenues. The concept is too frequently misunderstood and seen as a variant of cost control or a further application of activity based management [16,37-40]. The two concepts are closely related; a successful cost reduction program depends on a clear understanding of what drives costs. It usually refers to functions other than primary activities. The aim is to reduce the time and resources expended on support functions, which could lead to better returns, improved cash shortages and market credibility. A growing source of organization ineffectiveness is for business support functions to overwhelm the primary activities of the organization. Superfluous resources are channeled into unit level administration and compliance monitoring, staff education on company procedures and controls, quality assurance and internal policy auditing, asset management, information processing, and information dissemination.

4.1. Identifying Cost Drivers

When trying to optimize your budget, the first things that you should identify are your Cost Drivers. Cost drivers may include anything from the complexity of your product offering to the volume or frequency of customer service calls. To help you discover the drivers of your product costs, use the following steps:

Take an inventory of your costs. Gather three to five years of comprehensive financial statements. Get detailed reports from your accounting system about cost items and also subaccounts for depreciation, materials, supplies, payroll and employee benefits, account collections, third-party service contracts, maintenance and repairs, facilities, and outside processing and transportation. Consider other indirect costs as well, such as employee recruitment and training, quality assurance, information systems, and customer service [41-43].

Group the costs by use. List the costs as Production-Related, Selling-Related, or General and Administrative. Next, within each group, list the costs in descending dollar amounts. For example, in the Production-Related category, sort costs such as direct materials, direct labor, and manufacturing overhead by use. This is a crucial point: the cost variances must consist of high-cost line items. Also, be sure to include not only direct costs but also a large proportion of indirect costs as well.

Identify product-level costs. Examine how costs vary as volume changes. Ask yourself: What must I spend in any event? What are my fixed costs per unit? As your cost list dwindles, probe further. What are my variable costs? For what costs do I see a true relationship with a single product? Although these are normally

just direct materials and direct labor, sometimes indirect costs are also directly attributed to products using cost accounting techniques [44,45].

4.2. Lean Management Principles

Lean management is a successful and recognized strategy for reducing costs and improving business performance, that implies eliminating what is not adding value for the customer. Lean means to take only what is required to produce value for the customers and to take at the right time, focusing on eliminating non-value adding activities, whilst optimizing the associated costs [22,30,46-47]. Lean is considered a philosophy because it defines a guideline for decision making; it goes beyond the tools available and the detailed enablers necessary to implement everyday continuous improvement. In addition to the use of tools and techniques, successful implementation relies heavily on management commitment, cross-functional participation, a culture of teamwork and continuous learning. Recent studies observed that lean management did it for their companies, considering that only 10% of the implementations failed, and in the first years of the implementation, results were typically not perceived.

Originally derived from the Toyota Production System, Lean promotes a reduction in cost through the construction of a company-wide cost management system, covering suppliers, plants, and distributors. Routinely used tools in self-assessment and subsequent improvements cover:

Simultaneously optimized production planning and control: Capacity constraints in production planning must consider capacity utilization metrics also service levels in inventory control; Pushing “what” to manufacture and “how much” is needed from upstream to downstream based on demand and within capacity restrictions leads to higher levels of responsiveness, reduction of overall costs with inventory, higher levels of service to the end-customer, and increased utilization of capital employed. Although such an approach is accepted by the theory of constraints, it is seldom used. In the past, enterprise resources planning modules focused on what pending orders were delayed; now, improved modules are predicting shortages with appropriate warnings. Some organizations are hiring consultants specialized in supply chain management to define better their ways to design plans.

4.3. Outsourcing and Automation

Introduction

Every organization relies on skills and research for their original concept, product or service — what is called their original competence or expertise. Although it

may be small in scale, it is key to the future because it is fundamental for the success of the entire company. In addition, it is the area which the organization needs to protect against competition, otherwise it will not be able to resist the attack of competitors. Product offering is basically the place where a company earns its keep and also the essence of its business, which it naturally has to foster. Other "supporting" functions and activities, for example marketing and selling, a company normally opts to keep going in-house. However, it may be possible to save resources for the entire company, devoting the functions, activities and services of fewer and fewer people, through outsourcing or subcontracting for responsibility cover, specialized suppliers, and automation.

Outsourcing

Outsourcing answers the targeted organization's requirements with speed and efficiency. For example, it costs less and less to use video conference facilities, instead of maintaining offices. On the other hand, considering that there are many types of activity capable of supporting and ensuring the success of organizations, many functions and activities are becoming key factors for success with intense pressure to outsource. These can be identified on the basis of systems: it is necessary, therefore, within the framework of primary and secondary processes, to identify the areas where operations can create a differentiating factor for the organization. For the secondary processes, outsourcing is a suitable strategy.

5. Balancing Performance and Cost

The world of business is composed of many competing objectives and associated choices. Optimizing performance on a single dimension can be tempting. Indeed, many organizations do forget the fact that resources are limited and advance a wide variety of different demands on these limited resources. Such organizations tend to become disappointed with the systemic effect of the choices made by decision areas such as marketing, operations, and finance.

It is, however, possible to cut across business functions and simultaneously optimize two or more objectives. When several conflicting objectives are advanced at the same time, the organization employs what is known as a multidimensional decision model. Cost and performance are important examples of dimensional objectives. The optimization of one or the other dimension may lead to adverse effects. A situation where total cost is minimized will often lead to unfortunate product performance. Similarly, extreme performance demands

may lead to excessive total costs. Performance and cost are different requirements of the same physical product.

The constraint of being resource limited means that the decision-maker in an organization is concerned with resource burning. A certain quantity of certain resources is consumed in the development and the production of a physical product. If these resources are wasted or not used with care, the organization will incur more expenses than necessary. Hence, the manager in an organization cannot allow the company to be a poor performer from a cost perspective. But the company must also avoid becoming a poor performer when judged from a performance perspective. An organization with resources in the shape of capital, skill, employees, or other important ingredients express the resource burning priority when management states the desire of balancing performance and cost.

5.1. Performance vs. Cost Trade-offs

Generally, companies use their most costly and most capable systems to process their most important workloads. With an ever-increasing focus on the bottom line, companies are increasingly focusing on optimizing cost and power consumption without sacrificing performance. In this way, companies are reducing costs and increasing margins. In the past, companies could afford inefficiencies in their technology environment, but today it is critical that systems and applications are architected for the best business outcome. The business outcome may not only be the least cost for the most performance, but may also focus on the most business value created for the least cost. At a higher conceptual level, the business performance outcome may be driven by customer satisfaction or market share growth. But at a deeper technology level, such an outcome might be defined as the lowest cost for a given quality of service, or the maximum performance for a given cost.

The interaction and balance between IT performance and cost is what this paper terms performance management, and is a large area of ongoing research. The focus of performance management is essentially a trade-off between service quality and service cost. In the past, more than 30 years ago, there was discussion about extreme performance in the business environment with respect to scheduling theory. This extreme performance is quality (or performance) at any cost. The most natural examples are emergency and security systems. Business expectations continue to grow, and systems are expected to be available 24 hours a day, 365 days a year, and to have zero downtime. More recently, others have also stressed the importance of infrastructure redundancy so that some systems run without issues. However, costs and allocation of resources must be judiciously done. For less than extreme applications, these considerations of

doing things at extremes become extreme without compromise. A careful balance of cost and performance is a more reasonable business goal.

5.2. Value-Based Management

Value-based management is a powerful approach to measuring, managing, and maximizing shareholder value. A company creates value for its shareholders by implementing a growth strategy that strengthens its ability to earn an economic profit. Value-based planning is the vehicle for ensuring that a company's plans are clearly defined and communicated, and that they are consonant with the goal of maximizing shareholder value. Value-based performance contracts managers to reward or punish them based on the parameters that determine value creation. Value-based investment decisions require capital allocation procedures that prioritize projects according to their impact on economic profit.

The central precept of value-based management is this: Shareholder value is not merely a bottom line item that gets looked at once a year, as with net income or profits. These standard accounting figures, along with economic profit, are not valid indicators of value creation, because they do not assess size relative to tonnage nor risk relative to return. Recognizing that both tonnage and risk are absolute measures of cash flow, and knowing that the cost of capital is a multiplier of cash flow size, value-based management goes beyond simple financial metrics. Value-based management evaluates performance and makes investment decisions according to absolute price tag size, risks associated with price and return relative to cost of capital, because these parameters determine value creation. Shareholder value is a company’s worth, not its stock price. It’s the net present value of its cash flows plus the risk related to achieving those cash flows. The cost of equity capital is a multiplier of risk, not merely a skim from cash flows.

6. Tools and Technologies for Optimization

Keeping the cost down on important system resources is critical to preventing losses and improving profits. It is critical to monitor performance and costs for the significant resources of the systems so prompt corrective action can be taken on outliers and unusual spikes in consumption. Today, there are many tools that can help in these efforts. Some execute scripts that automate and monitor your systems. Others provide analytics capabilities, allowing you the ability to keep track of costs and usage in a self-service manner. These tools also keep a history of changes that can be useful in charge-backs and resource planning.

Some factors to consider when deciding what tools are needed to help monitor performance and cost is the structure and size of your environment. If you have a small scale system with very few resources, it might be fairly easy to hand-collect and monitor performance metrics. However, as soon as you start looking into larger enterprise scale systems with various different resource versions and even cloud-based systems, it makes much more sense to automate the process to eliminate human error and save time. For smaller enterprises it may be sufficient to use only the basic capability included within the native services. Each cloud service provider includes some sort of cost management tools that allow tracking of normal dashboards without purchasing third party services. So smaller companies can start off just using the cloud-native solutions.

6.1. Software Solutions

In the current computing landscape, different types of optimization products are available. Some of these products are based on hardware while others are essentially a software solution on top of hardware. Cloud computing enabled cost-effective tools that are marketed for ease of use and simplicity. Many times, optimization problems are built in close relation to the architecture of the machine being optimized. In this case, the compiler designer or runtime systems are required to develop an optimization solution that works well for the system as a whole.

Compilers are the first software system that tries to optimize the already defined application semantics. Many times, the application is completely clear from the compile-time perspective, and there is a plethora of optimization techniques to be used. Other times, compiler optimization requires the aid of the programmer, who provides hints or already developed, platform-specific libraries. Compiler optimization improves the normal run-time of applications and implements ad-hoc transformations for the parallel execution of the code on different hardware for a specific architecture. Code generation for state-of-the-art hardware architectures might be considered a final-form optimization technique. However, the majority of available compilers are unable to provide good optimization solutions for the variety of parallel architectures.

Modern task-based runtime system technologies implement compiler-like optimization functions at run-time, but work with less input information and focus on selecting the best adaptation policy for the future course of the application. These runtime systems are providing adaptive parallelism mechanisms without requiring any modifications to the initial coded application. An interface between the application task scheduler and the run-time system is able to manage the task performance in such a way it keeps the system busy while

the tasks are still performing considering external influences like cooling down and ac-stop events.

6.2. Data Analytics

The core idea of data analytics is to explore the history of the business and identify problems and opportunities that the business did not take advantage of. Historical data may indicate that some specific sector of customers increase their purchase rate after receiving an email, which would indicate that they are susceptible to promotions. Using that data, the business could create an automatic promotional offer for their next purchase in order to increase sales. For example, a hotel chain realized that guests who booked a room for three days gave a bad review regarding the lack of flexibility. Based on that insight, they created an offer in which guests who were dissatisfied with the process could change their booking up until four hours before check-in, without losing the money they had paid. These companies increased their sales during off peak days, solved a problem that had been created, and increased customer satisfaction with the service.

Data analytics can also be used to improve business operation efficiency, thus reducing costs. Data analytics may indicate that overtime accounts for a large portion of the budget, which leads the company to be more strategic about holidays and important dates in order to avoid high overtime costs. In the health sector, oncology and cardiology procedures take longer than is required, which increases costs, depriving other patients of getting treatment, and increasing waiting lines. In the financial services industry, a bank identified that around 30% of customers had been left with an outstanding debt after the loan expiration date. The bank sent those customers a friendly reminder and requested that they return to the agency to eliminate the outstanding debt and avoid penalties.

6.3. Cloud Computing

Cloud computing is one of the most omnipresent digital technologies. Its relevance is due to the utilization of services, of resources, and of applications delivered via Internet. Companies, allowing users to rent from them, rather than buy, online storage, computing power, software applications, or hardware resources such as networks or servers, are building their business on the provision of cloud services. These technologies, already addressing traditional IT needs, are starting to face more advanced needs, for example, enterprise resource planning and data warehouse support.

Cloud computing services can bring several advantages with respect to those called conventional dimensions for performance. First, for computing power

workloads, cloud resources can be easily, rapidly, and linearly augmented, within some limits. Second, for data storage workloads, cloud storage seems to be cheaper and can be augmented, also linearly, at anytime, within limits. However, additional costs in some data access patterns should be taken into account. For other workloads, such as for transactional processing, some workflows combining in-house resources and external cloud resources, or of hybrid clouds, can also be set up, with the aim of improving the responsiveness of a cloud database. However, such strategies need to be properly studied in advance since cloud services are always at risk of bandwidth overload, while capacity planning can also be more difficult than for in-house solutions. Third, for applications, both cloud services and utility services seem to be reliable. In fact, cloud service providers offer guaranteed availability, while utility computing providers sign service level agreements.

7. Case Studies

Performance optimization and cost management within the context of wireless networks, especially cellular, has recently gained increased attention due to the additional capacity and better performance offered by heterogeneous networks or HetNets. Several works have targeted different aspects, like performance or cost individually. Others have tried to optimize certain aspects but focusing only on a few parts, or very specific keywords, thus not delivering a clear picture of the problem. However, it is clear that every part of HetNets should be taken into account in order to accurately manage performance and cost. Embedded in the general area of Self-Organizing Networks, several works have tried to take SON a step further boosting the use of data mining leveraging machine learning methods. In these works, the decision-making process is offloaded to training and predictive, multi-criteria optimization and decision-making systems.

To validate the approach and understand the limitations and obstacles for a real-world implementation, four real-life pilot test cases were conducted. In these pilot test cases, SON applications for automation and intelligence were developed, deployed, and validated with real customers' traffic from commercial systems. The pilot test cases were conducted in various countries. These pilot test cases have a strong focus on Self-Organizing Network optimization, aiming for better managing performance and cost for the mobile network operators in both urban and rural environments. There are clear and additional enhancements for urban environments where mobile network operators have localized congestion hotspots while building a HetNet from scratch in rural environments. In urban

environments, small cells are provisioned normally in a traditional way for localizing footprints without any demand consideration. On the other hand, these small cells have their operational costs when deployed, preferably, noise, making providing and managing additional capacity at lower cost less than optimal.

7.1. Successful Implementations

Exponential increases in the complexity of compute workloads is driving both industry and academia to investigate performance modeling, optimization, and cost management. Over the past several years, we have worked on several projects with both local and distributed production resource providers. While this section speaks to some lessons learned, it heavily draws from the specific invalidations and pain points that we foresee based on our deep engagement with the field and its stakeholders, and not an extensive survey of prior work.

We motivate our conclusions with results from a variety of tools, applications, and use cases, including dynamic heterogeneity-aware batch scheduling for production heterogeneous computing clusters; workload management on energy-latency tradeoff using time-varying energy savings models; managing and building APIs for Heterogeneous Nonequilibrium Manifolds; online machine scheduling algorithms for heterogeneous single machines; work independence and transportation planning of job-shops with a focus on services with traversing multitasking jobs; scalability modeling for parallel phylogenetic likelihood inference; and energy-latency-product based scheduling for cloud infrastructure. We believe that the modes of optimization employed for these use cases, as well as the key considerations, priorities, and design choices, are common and extendable to several other domains. The remainder of this section briefly discusses some of these lessons, largely ignoring a precise technical description of the relevant pieces of infrastructure, model, and capability unless absolutely necessary. The goal is to help readers unearth practical modeling ideas and considerations behind some of our case studies, and motivate them to go experiment and build their own tools and frameworks!

7.2. Lessons Learned

The four case studies shown above provide a handful of lessons learned during our experimental implementations of performance optimization and cost management. Although the functionalities built into PEM have enabled the deployment of these optimizations on the two deployed Grids and have provided tantalizing results, there are a number of areas we feel PEM could be enhanced. In addition, many of the same areas could use additional research as it becomes increasingly important to further automatically manage Grid system resources.

The most important improvement involves case tracking. Adding case tracking to PEM would improve its overall functionality and enable it to handle higher overhead applications using policies that are not perfectly specified and whose selection also changes over time. Hopefully, such capabilities would enable PEM to track the case of such inputs through the selection of known new policies, or more dangerously enable it to discover policies that select other known policies.

The second needed improvement involves policy specification. While specifying simple "if-then" style heuristics is helpful, more general policies would severely increase the number of possible funded optimizations and enable the optimization of a wider variety of longer duration, higher overhead workloads. Thus, research is needed to enable PEM to support the automated discovery of such policies. This research would both improve performance during multi-case optimization and "training times" used in the case tracking of a higher overhead application.

The next area is planning. As many computer users have a largely static policy throughout their usage, it would be helpful to develop or trial a centralized policy repository where users would upload the optimizations they used and the selection periods to assist in the automatic detection of optimization policy changes. Finally, while PEM attempts to be smart, it would be helpful to explore the trade-offs of having different passive approaches that users could specify.

8. Future Trends in Performance and Cost Management

Modern organizations are increasingly relying on technology to make bold decisions quickly and accurately. The pace of change and the accompanying pressure on public and private organizations create unprecedented opportunities, as well as challenges for leaders to do it right. These trends are pushing organizations to look for new ways of measuring performance and of applying associated incentives in order to deliver desired programs. Recent trends in the performance management landscape are a measurably broadened perspective on what constitutes successful organizational performance and a movement toward a more holistic view of measurement-based performance management.

A growing number of groups are interested in holding organizations accountable for more than just economic surplus or return on investment and likewise, more companies are being judged on the business's ability to positively affect other stakeholders. Additionally, recent corporate scandals and changes in legislations

relating to economic and environmental sustainability have focused attention on how organizations should be measuring and managing not only their economic performance, but also the impact of their operations on the environment, upon the community, and upon society at large. These changes have put demand on organizations to create comprehensive performance management and control systems to minimize the possibility of unethical behavior. An expanding perspective on performance creates more demand on companies to be transparent about their performance. The use of balanced scorecard to create a comprehensive picture of organizational performance is increasingly common and additional technology efforts are being developed to enable organizations to provide memorial, audit-proof statements of their financial, operational, and compliance performances.

8.1. Emerging Technologies

Performance and cost considerations will always be a part of enterprise networking; however, emerging technologies are bringing forth paradigms and metrics never before seen that will fundamentally change the way performance and cost are measured and monitored throughout the enterprise impact loop. Two examples that hit at the heart of performance and cost materialization are network and application virtualization. Virtualization started as a way of hoarding computing resources. Server virtualization technology enables multiple independent OS instances run on the same physical machine, thereby drastically improving utilization on the large servers that have been foisted on IT departments. Network virtualization came of age as a way to economically scale the ever-expanding enterprise WAN without an ever-growing budget. Emerging virtualization technologies for enterprise IT will stretch the capabilities of both server hardware and WAN links to the breaking point. However, ultimately performance and cost are only the intermediaries of the service quality that enterprises need to make money. Waiting for file downloads, needless application troubleshooting, and unplanned office visits by technicians who are either waiting for computer installations or have been summoned to fix problems are a few of the service costs that give IT operations headaches. New technologies will be developed that will give IT departments better visibility into service quality. These service management tools will provide useful business context to the analysis of performance and costs by correlating specific incidences with service impacts.

8.2. Sustainability Considerations

Over the last decades, the importance of sustainability, and how businesses grapple with it, has been a hot topic for every business leader. Management

decisions have had repercussions at the environmental level, often with devastating consequences, and nowadays companies must respond to this reality, integrating policies in line with the long-term sustainability of the planet. The future, with zero pollution targets, cannot be achieved without a great collaborative effort. Enterprises at all stages of the chain need to make decisions that do not lead to exploitation and destruction of the environment. Accountability and transparency will be the key words in this process.

Probably the most important decision a company needs to make is how many resources and in what time frame it wants to produce its goods or services, i.e. it wants to decide which level of expected pollution could generate for the community. Managing and optimizing this emission is not easy because companies act during a lengthy period, generally several years. However, setting an emissions budget is something that is already being implemented by different actors at different levels and that is slowly becoming part of the strategic planning.

Recent years have also seen the rise of other decision variables driven by regulatory authorities or, more generally, external pressure groups. For instance, some fans want to encourage companies to invest in policies to reduce and optimize pollutant emissions, such as allocating funds to develop solutions to develop carbon-capturing technologies or technologies that generate clean energy at a competitive price. In these cases, stakeholders at different levels, people, local communities, states, etc. will put pressure to urge the company to allocate some funds to this purpose if you want to receive their support.

9. Conclusion

This section highlights some of the concepts and approaches described in this chapter, emphasizing how to use the orchestration and provision services to define enterprise workflow and on-demand enterprise model. Faced with the constant demand to reduce time to market while increasing flexibility and customization of their service offerings, organizations need to embrace an increasingly wider and more complex range of enterprise models, from hierarchical to on-demand. The provisioning and orchestration tools can help to get there, but it is not enough to just implement and adopt these tools. Organizations will need to drive a new approach to managing the enterprise from a business and a technology process perspective. Business process tools focus mainly on defining how the workflow is conducted, with some only including

elements to define timing and sequencing. These tools tend to map the enterprise to a model that is rigid, and any modifications required – for instance, to accommodate differences between customers, product specialties, or seasonal variations – require a full change cycle and a new deployment. The actual management of the business activity is done outside the business process tools; these tools are mainly an interactive viewer for workflow and associated activity. Integrating customization, special occasions, and exceptions to the workflow into these tools would help to overcome the deficiencies in the current use of business process definition, implementation, and management systems. In developing an on-demand enterprise model, there is an additional dimension to consider in the outsourcing decision. On-demand enterprises will make heavy use of outsourcing not just to free capacity and expertise to focus on core activities but also to deliver a wider range of options.

References

[1] Ji C, Li Y, Qiu W, Awada U, Li K. Big data processing in cloud computing environments. In2012 12th international symposium on pervasive systems, algorithms and networks 2012 Dec 13 (pp. 17-23). IEEE.

[2] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.

[3] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.

[4] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.

[5] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.

[6] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.

[7] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.

[8] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.

[9] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.

[10] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.

[11] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[12] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[13] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[14] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[15] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[16] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[17] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[18] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[19] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[20] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[21] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[22] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[23] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[24] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[25] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[26] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[27] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.

[28] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[29] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[30] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[31] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[32] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[33] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[34] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[35] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[36] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.
[37] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).
[38] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[39] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[40] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[41] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[42] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.
[43] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[44] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.

[45] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.

[46] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.

[47] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 7: Dimensional Design and Star Schema Structures in Data Processing and Modeling

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction to Dimensional Design

The rapid growth in the amount of data generated by modern organizations and their dependence on a varied array of information systems creates a need for faster turnaround times for decisions based on the analysis of such data. Successful organizations require a change from stand-alone, departmental decision systems to an enterprise-wide knowledge base that can link disparate functions, processes, and systems and be used to support cross-organizational decision-making activities [1-2]. Historically, organizations have relied on both operational systems and decision support systems to meet their information requirements. The operational systems are not sufficient because the aggregated data may not be available fast enough to support key management functions.

The amount of data being generated has continued to expand exponentially. Further, the data may be physically distributed within the organization as well as being functionally distributed across various departments [2-4]. Finally, organizations are increasingly dependent on knowledge workers who seek immediate access to all the data needed to create a solution or strategy. Operational systems, as well as existing DSSs, are not sufficient to meet these demands. One of the goals is to provide a comprehensive discussion of those key functions of DSS and data warehousing that are particularly important to meet the needs of modern organizations in addressing competitive advantages globally. Central to this discussion is an examination of the business needs and technology developments that necessitate the evolution of DSS and the data

warehouse and the implications of this evolution on design and implementation issues.

2. Fundamentals of Data Modeling

Data modeling is concerned with the collection of facts represented in a structure that can be translated into a physical database with implemented access rules. Data models have natural generalization by taxonomy through levels of schema abstraction. At the highest conceptual, or abstract model, the representation is indicated, showing the kinds of data in the model and the connections between them. Each higher level branch of the taxonomy shows increasing levels of restriction imposed upon the lower members of the lower level branches, which in effect is a hierarchy of structural specialization [5-6]. The higher level structure shows the essential content, while the lower level branches must comply with the definition of the lower level structures. Special-purpose models have increasing detail. They are designed to represent a specific subset of the structure of particular applications. Lower levels present tree, network, and relational models, depending on the purpose of the special representation. Ultimately, the physical implementation of a particular application in a specific platform must comply with the design parameters imposed by the lowest level model [7,8].

In a nutshell, data modeling identifies a consistent and predictable means by which to represent reality, with "data" being an abstract concept representing observable items and relationships that possess common attributes and functional content common to an application. The "model" is a conceptualization that defines specifications and constraints that allow realizations of data representation. A semantic model represents the maximum freedom of abstract representation. The level of abstraction in a semantic model corresponds to its relational indifference or completeness. A full semantic model specifies and relates attributes for all kinds, but contains no further structure specification. Thus, many different actual structures may arise, all yielding the same interpretations of the data. The constraints imposed by a fully constrained model structure may be classified.

3. Overview of Star Schema

The goal of design is to capture that which is relevant and important to the business, and to do it as simply as possible, to simplify implementation, to

increase understanding, to provide the needed flexibility, and allow for long range planning and the ability to grow and change. In data warehousing processes, the focus is supported by business goals by capturing business processes in business terms, orienting the data to include primary business activities and events as focal points, applying appropriate measures to provide a clear indication of the business status and dynamics, and providing an orderly means of traversing both the dimensions and measures [9-12]. Star schemas are characterized by the fact that the central part of the schema is a single fact table. This is a mechanism that stores business events in a highly normalized form. All measures and keys into the auxiliary dimension tables can be retrieved from the single table. Auxiliary tables provide the means for tagging events with descriptive values. In proper designs with moderate model sizes, users get the benefit of easier understanding, increased performance, increased flexibility, and simplicity of design. The main concept of the star schema is to have the fact table at the center and all the normalized dimension tables surrounding the star. Each fact event is represented as one record in the fact table, which only contains key values to the dimensional tables and measures. The actual reporting for the query is done by joining the required normalized dimension tables to the fact table.

4. Key Components of Star Schema

Four objects are key components of the star schema. Three are physical tables that contain the data; they are the tables of fact data; and the character definition value tables throughout the factor accommodate alternative value definitions. The fourth is a relationship, or link, that describes data relationships, previously is to be used to refer, to indicate to what (if any) other part of the schema a specified value refers [7,13-15]. The word value like a string of text, from a text document of some size, but it also has a definite information context (connotation) of interest to the schema user and is necessary to notationally return the factors.

Fact Tables

The tables of the factor contain the data of interest to the schema user. The tables are self-describing entities that allow the user to significantly reduce the overall volume of information being enabled by the schema. General facts usually refer to a specific entity type of sufficient value dimension that tells about the "size" of each individual entity instance.

Dimension Tables

The tables provide another level of characterization of the data. These tables provide additional information that allow the schema user to more specifically locate the entity states of interest. They enable the user to sub-specialize the general fact information to, and even further characterize, small value dimensions of the general fact to keep a limit dimension count.

Relationships Between Tables

The database contains special relationships to define what the various data items contained in the various tables of the stars are and how they relate to other parts of the star. A major attribute of the schema is the use of required schema links. The links identify to something else in the schema that is of known content and what constitutes the overall content of the star schema as a whole.

4.1. Fact Tables

Most commonly, fact tables express a quantitative relationship between two or more dimensions; in these cases, facts found in a fact table are said to be "measured" or "analyzed." Examples of common facts that readers may recognize from business analysis include sales in dollars, the number of units of product sold, the score made in a game, and so on. The practicality of analyzing a specific measurable entity is what drives its expression in a fact table. This emphasis on a single topic is what distinguishes a fact table from a data dump or data warehouse fact cluster; in a fact table, redundancy and normalization cannot be justified and must be prohibited, making facts unique unless expressed on the micro level.

In a strict and lowest common denominator sense, all fact tables contain foreign keys representing the primary keys of dimension tables. Some facts are expressed as a weighted sum [9,16-18]. When an analyst refers to any summation of some measure or one of its parts, this is the sum associated with a foreign key value that identifies the transaction. The exception to foreign keys being the only columns in a fact table is when foreign keys are used as an additional dimension. Such applies as a generalization or a multidimensional cube, whether in paper or electronic form. Readers may more commonly think of OLAP as representing such cubes. The exceptions to a fact table are solved in session sophisticated backroom environments, where many views of the actual tables are made available for the analyst and others in an organization.

The rows in a fact table represent transaction data, and the columns represent specific details about those transactions. In a practical data design for OLAP, there will usually be more fact tables than dimension tables. Each fact table will contain at least two foreign keys pointing to the primary keys in two dimension

tables. Transactional fact tables will usually also contain numeric measures associated directly with the foreign keys. Such measures do not have to be expressed on the micro-level, or for each row in the fact table. Sales at retail locations, for example, may be expressed as the accumulation of sales for an entire day.

4.2. Dimension Tables

While a fact table is a big, sparse, low-density structure dense with numeric data, dimension tables are small, often denser structures that contain descriptions of the facts. There is usually several times more rows in the dimension tables than in the fact table. For dimensional data warehouses, however, the number of records in the dimension tables can be very large, existing in many millions or even billions.

The dimension tables contain descriptive attributes, which are typically textual fields for which each possible combination of attribute values becomes a large concatenated string. These descriptive fields characterizing the data in the fact tables are often the ones that the end-user queries. Some examples are CUSTOMER NAME, PRODUCT NAME, SUPPLIER NAME, and STORE NAME. These attributes can also be used to query on particular values, e.g., "Total sales for Store 25," "Total sales for state of California," or "Latest shipments from Japan." For all such queries, the dimension tables are used to join in the textual, descriptive information.

Some dimension attributes may not have values for every row in the fact table. For example, some sales order rows in the fact table may not yet have a ship date because the corresponding order has not been shipped to the customer. In this case, the fact table has values for the corresponding ship date fields.

All dimension tables have columns of values that are used directly in the queries. Since each value can also represent an index, these are also the primary key columns. Usually, these key columns consist of one main key since generally only one set of values is being used. Other column (nonkey) data fields can have small but lengthy combinations of text linking them together.

4.3. Relationships Between Tables

At the star schema's core are relationships between dimension and fact tables. All fact table rows are connected to dimension tables through foreign keys [2,19-20]. Foreign keys are identifiers that are defined inside a fact table to reference primary keys defined within the associated dimension tables. As an example, a transaction fact table is organized by transaction date. This fact table also

specifies the employee processing the transaction, the customer doing the buying, and the product involved in the transaction. Each of these keys (date, employee, customer, product) points to an entry in a dimension table where more detailed information is stored, including name, address, category, and descriptor fields. Foreign keys from the fact table lead to the primary key entries in the dimension tables, where supporting information is held.

The relationships between tables in a star schema are almost trivial but also deeply and intricately complex. That is, the relationships in a star schema are almost trivially simple. In the simplest implementation, the relationships in a star schema are a series of foreign key to primary key relationships whose primitive form models the relationship type of "is a part of", where "is a part of" refers to the lower granularity column of the dimension table and the higher granularity column of the fact table. For the reason that, inside the fact table is stored a set of rows that identifies a unique event, the user feels overwhelmed when analyzing the data in the star schema, and a confused relationship appears between fact table and dimension tables.

5. Benefits of Star Schema

The star schema structure is very natural. The dimensions are distinct and, for the most part, physically separate from the facts stored in the database. The physical layout is easy to implement in a relational database, and the queries are usually simpler than those for a normalized schema. Each dimension can be fully used in each fact query and need not be joined with other dimensions to select the appropriate values. Also, since the dimensions are separate, they can be joined with the facts without additional joins to other dimensions when selecting a value in a dimension. If dimensions are related (as will often be the case), then the relationships can easily be represented by defining them together with a join predicate. Database implementations supportive of this style of querying can implement substantial optimizations on star queries, which is of great help for response time and query throughput. In addition, various star joins and star caches, as well as other implementations, can also take advantage of this structure by inferring the optimal physical organization.

The star query structure naturally organizes facts to maximize the efficiency of a range of important queries useful for decision support [9,21-23]. Typically, the response requirements for such queries are far stricter than for usual OLTP; they stress the volume, not the response time of transactions. Such requests are usually

ad hoc but tend to originate from a common set of users with similar decision support requirements. The principle ingredient is usually expressed in terms of a narrow band of operations through a hierarchy that aggregate and correlate. These requests tend to query small segments of the multidimensional space and return small answer tuples, which form a serious bottleneck for throughput if the response time for these is not very small.

6. Challenges in Star Schema Implementation

The important aspect of practical implementation of star schema objects is their proper development at the level of dimensions and fact tables [24-26]. In general, a star schema refers to a data structure consisting of a large central fact table surrounded by a set of smaller tables that connect to the fact table via foreign key links. In this way, the dimensional tables can be viewed as the star's points and the fact table can be considered as the star’s core. Any dimensional table can have one or more attributes. Attributes create hierarchies that assist in presenting data at different levels of detail. The attributes are often characterized by a very low selectivity with respect to the amount of data reflected in the fact table. Therefore, the dimensional tables tend to have a small size compared to the size of a fact table. The relationship between the fact and dimension tables usually accepts a one-to-many correspondence.

Typically, the time, organization, and geolocation tables comprise the main dimensional components of any data warehouse schema, having consequence during the time and effect of the event described by a fact in the central table. Then, in commercial actions, the geolocation table is usually addressed in two parts: the organization that sells and the country that buys [8,27-30]. That said, the fact tables are intended to keep the history of the commercial activity, while the dimensional tables are responsible for their description. This description will necessarily change over time (i.e., the organization can change its name, the country can change its currency or the way it is called, etc.). The temporal characterization becomes mandatory in these cases. Thus, changes must also be reflected in the star schema objects of the computer.

7. Comparative Analysis: Star Schema vs. Snowflake Schema

The construction of multidimensional data models can be accomplished using different techniques. The most well-known of these are the star and snowflake schema. In the star schema, we assume a central fact table that is connected directly to dimension tables. In the snowflake schema, similar to the star schema, we have a central fact table. However, this fact table is connected to additional tables that represent some of the dimensions, and these tables are connected to more specialized tables that represent lower-level dimensions [9,31-33]. Hence, in a snowflake schema, we assume dimension hierarchies. Fact tables in both schemas are usually large, dimension tables are usually smaller. When multidimensional data models are populated with data, users typically want to query data from specific dimension tables to analyze the data [34-36].

It is important to note how dimensional data models are created, stored, and used in a given DBMS. Once we create the dimensional data model, we need to populate it with data. For the population step, there is a difference between using star and snowflake schemas. If we assume a star schema, all measures concerning a specific event are stored in a single record in the fact table. This assumption may lead to a restriction in the design of the operational data model. If we assume a snowflake schema, population of the dimensional data model can be easier, faster, and simpler compared to when using a star schema. This is possible if we assume that the fact tables in the data warehouse contain large cover data. Such an approach is not new; however, it is less well-known within the community of data warehouse researchers and practitioners.

8. Best Practices in Dimensional Design

In this chapter, we consider some of the dimensional design details. Working with dimensional design involves understanding that there are many ways to create and store facts, which can later be brought together for consumption. Questions like whether dimensional keys should be surrogate or natural keys, whether dimensions should store pre-joined reference data, and how descriptive attributes should be formed are design considerations that do impact performance of the model. However, just because the design can impact performance, it does not mean that all dimensional designs need to be parallel-optimized. It is just like creating an application. If it is expected to have 10 transactions an hour, any

coding style will probably do. If it is a stock exchange application allowing thousands of trades a second, you will consider using the smallest data types and pre-processing everything in the batch mode.

8.1. Naming Conventions

What is in a name? A lot, really! The difference between a confused analysis and a happy one is understanding the meaning of the tables and columns he or she is working with. And that meaning is often hidden in the identifier of the object. And object names can get very long. So, we suggest the following language elements be observed in naming data warehouse objects.

8.2. Data Types and Formats

Analysis using a data warehouse is mainly at an aggregation level. What this means is that, although the current situation for the business at the aggregate level is that those months and years have been chosen to look at, the details at the lowest level create the details of the report.

8.3. Handling Slowly Changing Dimensions

There are a few different types of slowly changing dimensions. Each has disadvantages in modeling, space, and use. Some have disadvantages in keeping historical data. The dimensions can be classified into three types. Type 0 is where historical data longer than the time series is never required.

8.1. Naming Conventions

Most data modelers have their own “style” for writing data models, and each organization may impose its own rules on the design team as to the exact details to be used when creating a model. There are several decisions that have to be made before any model is started. These rules are used consistently in a model to provide a clear, unambiguous representation of a data model.

1. Naming Conventions for the Database

1.1. Table Names

1.1.1. Sensible Plural Names. Star schemas consist of a central fact table, related to several dimension tables. Dimension tables usually correspond to nouns, while fact tables hold quantities associated with the nouns in the dimension tables. Table names should be plural nouns, except when a single dimension is being filtered by each record in the fact table, or the table contains pre-calculated data that have a single attribute value for products, customers, salesmen, etc. in daily

sales fact files. In these two exceptional cases, fact table names should be singular.

1.1.2. Standardized Short Names. In environments with many one-second refresh daily files and an enormous number of tables and enormous query complexity, it is important to convert long dimension or event descriptions into standardized short names. For example, all probability-related tables might start with “prob”, all coupon tables might use “coup”, and variations for interactively-driven event tables might be “man_int” or “sys_int”.

1.1.3. Standardized Detail Qualifiers for Repeated Dimensions. In the above example of the fundraising event, the actual cost of the event might be stored in several fact tables, such as contributions made via the event, expenses associated with the event, and the income generated through the event. In this situation, it is essential to clarify the repeated dimension qualifiers by naming the appropriate relationships. These may consist of qualified hierarchies in one or more dimensions, or fact table suffixes like “_contrib”, “_expense”, and “_income”.

8.2. Data Types and Formats

Creating a structure with keys is the first step in data structure development, and there are many rules that need to be addressed when building data structures. This begins with creating the appropriate data types, making sure lengths and hierarchical sequences are respected, ensuring that dimensional attributes are built, and logical formats are agreed to [3,37-39]. Likewise, assigning standardized formats, always exclusive to that dimension key, facilitates coding and helps prevent errors. Determining the structure will ensure that codes are meaningful. General standardization, as much as is practical, will help eliminate problems in coding. But the reality is also that the more you standardize coding, the more you will limit the ability to reflect “real world” differences; the standardization needs to be used sensibly but secondly there is often a need to reflect real world differences and those codes should be variable. These types of codes are referred to as data types, describing the allowable “size” of the data type [36,40-42].

Every item must have allowable values, and invalid data should be rejected. There must be standard rules for the allowable values for every dimension and each different pseudo fact. Dates and time slices must be consistently represented and banded. The rules that reject invalid data must be consistent across all sinks for the same data, even when the data flows through different sources before being delivered to the sinks [40,43-44]. For values that are not dates – such as numerical increments, decimal fractions or text strings – we also need to agree

on formats for intervals, increment ranges and the decimal point or decimal comma to be employed throughout the scheme. The more standard rules for values, formats, ranges, sources and sinks can be applied across a project the better the data quality will be.

8.3. Handling Slowly Changing Dimensions

The three strategies are the SCD types 1, 2, and 3. Each of these methods maintains a different level of dimensional history in a dimensional model. SCD type 1 overwrites the existing value of a certain attribute but does not keep a history of the previous value. SCD type 2 creates a new record for the dimension member each time the attribute value changes. The record must contain a system date-start and date-end column or a current row column, such that valid time for that record can be determined. The idea is to utilize a surrogate key in the dimension table to signify an active member and retain historical versions of that member in the same table. Implementation of SCD type 2 typically requires specific data files per new period, allowing to reload dimension data with inserts into the dimension. Existing dimension tables and associated fact tables should be kept intact without alteration. A mapping-relational table linking the active surrogate keys for the specific attributes involved in fact records is created to link transactions. The task of updating the dimension table with new surrogate keys can be done periodically or as a one-time data processing operation followed by moves to archive data storage for the older fact records that refer to inactive keys. If the SCD type 3 method refers to a changing attribute of a member, there will be an attribute for the active value and another for the previous value of the member. With the fact record indicating which is the current member, history will exist for only one event and not the dimensional member.

9. ETL Processes in Star Schema Design

ETL processes are an important element of the design methods, for example, of data warehousing and OLAP. For Business Intelligence implementation, DMs can visit appropriate ETL sections in this work, but those sections describe techniques used in real implementations that differ from the theoretical approaches deeply considered in the ETL literature. Here we consider ETL processes in star schema design on a prescribed design path and strategy: the star schema design is done before any physical implementation. The goal is business domain exploration and preparation of customized views beneficial for business needs, which are created by collections of facts and business related dimensions, with appropriate dimensional design principles applied.

The extraction step consists of identifying sources and retrieving data according to the dimensional model principles, semantic and business characteristic of dimensions and facts, and delivering selected data to temporary tables used for filtering and some preliminary cleaning [3,45-48]. This is a business processing step, semantically preserving and with no unique operational business expertise. The extracted data collection is subject to further processing – cleansing, preparation, augmentation, and translation to the semantic and business spaces of the target dimensional model. Cleansing, preparation, and transformation of operational and external data are an important element of dimensional modeling, as they produce dimensions that fulfill the dimensional model characteristics for business activities or facts. Once all dimension and facts tables are processed, meaningful, and in the target dimensional space, the final step is to copy them to the data warehouse repository. Dimensional model design is a prerequisite step.

9.1. Extraction Techniques

Data extraction is an important process in the data warehouse development. The interests in data warehouse inhibit the issues in extraction, transform, and load (ETL) process, optimization and automation are a major issue in the development and maintenance of data warehouses [5,19,49]. Data warehouse installations have increased dramatically, primarily driven by advances in technology and the amount of data being produced and required for decision making.

The data extracted using different techniques; some are complex and some are straightforward. Data extraction technique is conceptually simple; its simplicity hides a vast amount of complexity which is at the heart of any data warehousing system. Data extraction is the process of getting data from various data sources, in data preparation this is the first step, and simply extract the data from operational sources or from external sources for data warehouse population. To these operational sources we can add web sources, sensors, and also sophisticated computational systems. Data from these various sources can be extracted through custom-written extract routines, or through sophisticated ETL tools provided by vendors that combine various industry standards.

The extraction process is distinct from the transformation, loading, and further paths of the whole ETL process, because extraction can more heavily universe-dependent. For instance, data extraction from flat files or object relational mapping is simple and straightforward, while other data sources like relational databases it is complex and costly. The ETL process can be implemented in a linear, batch-mode fashion, with data extract routines located in the target data warehouse servers. For example, data residing in data messes or distributed across several databases have to be consolidated before the extraction stage, and

there are no rationale behind scheduling processes in a compressed, inactive data warehouse period.

9.2. Transformation Strategies

The Data Warehouse Toolkit introduces many transformation strategies concerning validity, accuracy, consistency, uniformity, and redundancy in transaction processing. The goal is to ensure that the data placed in the data warehouse for querying and analysis is trustworthy. The concepts associated with the dimensional model and star schema architecture address why and how quality problems arise in the context of summarized business processes, and explains how longstanding practices in the handling of business events, financial transactions, and operational data adversely impact the usability of the DW/DBM lifeblood.

Quality problems arise when transaction processing systems are not designed with future needs for querying or analysis in mind. It is known that transaction processing system designers do not pay sufficient attention to the need for presenting a unified view of the organization's operations in immediately comprehensible, high level, insightful, time-relevant summary form. It is also known that quality problems can be resolved by a dimensional design and a star schema structure carried out at the operational data processing level. The ETL or ELT functions at the data warehouse level add additional techniques to resolve the remaining quality problems.

Data quality issues are often resolved in an ad-hoc, non-systematic manner. Detection and resolution methods can be identified for a range of common quality problems. We classify the solution methods according to three phases of ETL function: transformation to an operational data store, transformation to a data mart or data warehouse, and data access using query, view, or reporting language. As a basis for discussion about the method of analysis, we define some terms about what we mean by data cleaning and ETL. The purpose of ETL or ELT functions is to detect record and value errors and inconsistencies in relational data records. Data is cleansed when those problems are detected optionally, actions are taken to correct, correct, or reject them. "Data Cleaning" is a general term that encompasses a variety of data error classes, detection techniques, and correction methods. Data error classes group similar problems.

9.3. Loading Procedures

The data population or loading procedures must physically instantiate the data in the fact and dimension tables designed during the conceptual and physical

modeling phases. Three major ways have been proposed: single-load, multi-load, and scheduled-load.

The simplest procedure is to create and populate a data processing and data warehousing system at one time. For a new application, this "single-load" procedure can usually be invoked without difficulty because the star schema being populated is normally empty, and the "initial load" can be performed quickly, before the data warehousing and data processing system are put in production mode.

The second approach, the "multi-load" procedure, is designed to support the process of incrementally adding new data to an operational system. This process can occur on a scheduled basis for periods of dimensional data refresh, such as daily, weekly, or monthly, etc. The third alternative is to schedule load and refresh data on the star schema tables so that queries can still be processed, even if somewhat less efficiently during the refresh period. Each of the methods involves trade-offs, both technically and organizationally, that can impact on the entire data processing and data warehousing environment. The several of the capabilities of the loading processes, and the tools available today. For the business analyst, these differences need to be understood, and a decision made about which methods will be used as part of the architecture and the tool set used to implement this architecture.

10. Data Warehousing Concepts

The subject matter of this work centers on data processing and modeling. This is a very popular subject due to the prevalence and importance of data-driven information. Companies try to derive information out of the data that they store, for business enhancement and profit increase. Data-driven decision making encompasses factors like market segmentation, profitability assessment, budgets, expenses, etc., and aims to successfully position the company in the marketplace. Data cleaning and preparation, data processing and modeling are focused information retrieval techniques that combine knowledge from the business, knowledge from the field of computer science, and the right technology, to provide the information required. Within the above broad subject area, emphasis on dimensional design and star schema structures, aims at providing a potent and operationally efficient mechanism of information retrieval. Common to all approaches that derive decision-supporting information from data is Knowledge Discovery in Databases. KDD is the nontrivial process of identifying valid,

novel, potentially useful, and ultimately understandable patterns in data. A collection of tasks and techniques, data processing and modeling form the main parts of KDD. Of paramount importance are the stored databases, or Data Warehouses, that support the KDD process. A Data Warehouse provides its users an integrated, non-volatile, read-only collection of data focused on decision support. The data contained in a DW is presented in a dimensional format. Such a format allows users to very efficiently retrieve the data required in a decision-making process. Dimensional databases, or star schemas, are optimized for read-only operations. As the demand for data retrieval at all levels increases, the Data Warehouse becomes used more and more as the company database.

11. Role of OLAP in Dimensional Design

A few words may clarify the reason we mention OLAP so frequently in this work. The objective of data processing projects is generally to provide information and make it usable for business experts in their analysis and decision making tasks. OLAP systems, with their ease of use, powerful multidimensional analysis features and performance, are the preferred tools. However OLAP is not the only tool available and we have implied throughout this work that dimensional design also serves a much broader audience and much broader purposes. The dimensional model is not merely the logical structure for OLAP tools. A sound dimensional design provides for enterprise modeling and designing, graphical user interface and report design, and warehouse compounding and loading design. Furthermore, the multidimensional design has been used extensively in conceptual models for any kind of databases, even highly normalized ones. It provides users with a very usable means to express world semantics.

The way the dimensional model has been incorporated into OLAP implementations and special purpose tools is a great enhancement and contribution to the usability and capabilities of OLAP systems. OLAP enhancements to the dimensional design concepts, which we briefly summarize, are the following. They incorporate OLAP into conceptual model design and provide for tools for automatic generation of the logical model from the dimensional design. They enhance the ODL design with additional design rules for compound cubes, derived cubes and aggregate cube hierarchies. They implement schemas beyond the star schema model. Dimension hierarchies in the star schema are restricted to the owner dimension and they cannot support join conditions between dimension tables. The data contained in each dimension table of the star schema is accessible directly or indirectly by the fact table, on the basis

of the relationships defined in the hierarchy. The star schema model does not permit cyclic or non-conventional relationships between dimension hierarchies. There is also no semantic-based extension to star schemas beyond non-conventional relationships.

12. Case Studies of Star Schema Applications

Star schemas have been developed and refined in real-world applications for almost 30 years. They have arisen in many different industries and in almost all application areas. The software vendors, whose products support dimensional design and star schemas to some degree, come from various vertical sectors, including retail and e-commerce, telecommunications, education, utilities, transportation, healthcare, social services, marketing, and finance and banking. Some of these vendors concentrate on these industries; others cast a wider net with their solutions.

Star Schema Applications in the Retail Industry. Pretty much the first success in business intelligence was achieved in the retail sector in the early 1980s. The space designed for this initial success was sales in large retail companies. Despite the rise of e-business, such analysis of corporate sales in large retail companies remains today's most common application. Such analysis has become more sophisticated, more diverse, and cheaper over the years, but the essential application area remains the same: who buys what, when, how, and where. Such analysis is required mainly by the marketing, product management, sales force, promotions, and strategic planning departments of these companies, but it is also important to others, such as supply chain management and customer service.

One of the more exciting areas of current retail practice is the use of the Internet and related technologies to permit mass customization of product offerings. This development is at its genesis. However, the star-schema-based techniques developed for analysis of ordinary retail sales already seem to have been adapted for situational awareness in this new environment. The merits of this new thinking, however, cannot hide the fact that mass customization is still much less important than normal retailing by large chain stores.

12.1. Retail Industry

The term retail refers to an industry that sells products directly to consumers. At the end of the purchase process, the product is delivered to the customer and payment is completed. In the sale of some goods, there is no physical transfer

from seller to buyer. Initially, retail was understood as selling goods from premises. Currently, the retail sector is understood as all activities of selling goods and services to the economy's inhabitants for personal consumption, using the distribution networks. The consumer goods are most often physical products, but can also involve services.

Retail trade is defined by businesses whose main economic activity is the sale of goods and services. These businesses act as intermediaries between the production and wholesale distribution level and the final consumer, and complete the process of making products or services available to consumers in order to meet their requirements. In Poland, service retailing is an important and developing economic segment. The main parameters characterizing the retail sector include sales revenues in constant prices, the number of employee jobs, employee productivity, industry concentration ratio, and average consumer shopping basket amount. In Poland, retail sales growth in constant prices has been positive since 1995, whereas the phase of global recession in industry and service sectors in 2002/2003 was not applicable to the retail sector. The average size of the shopping basket in Poland as well as other EU countries indicates that consumer purchasing power has not been exhausted yet. The structural changes in the capital structure of retail enterprises are positively influenced by Poland's accession to the EU since May 2004.

12.2. Healthcare Sector

Healthcare systems are complex entities. They contain many biotic and abiotic components and have a multitude of heterogeneous data that describes both the systems' entities and the phenomena relating the entities. Countries and/or regions' healthcare systems must provide appropriate health-related services to all their population. Therefore, healthcare systems must include and sustain a considerable medical resource. The quality of the services offered by healthcare systems depends on the financial resources, on the personnel involved, and on the equipment used for diagnosis and cure. For various reasons, the multitude and complexity of the information that describes the entire healthcare sector make it extremely difficult to acquire the entire necessary data. Furthermore, the analysis regarding the quality of health services, the quality of the population's health, the costs of the services, the efficiency, and the performances of the healthcare systems have become actual practical concerns of institutional and economic importance. These aspects have led to the need for the development of star schemas.

The term star schema is used to designate one of the simplest forms to organize and model data in dedicated warehouses for OLAP and economic decision-

making systems. One of the first definitions of the star schema concept appeared in a paper that explained how a data warehouse, a core component of an information system, collects data from multiple operational sources inside and outside the company.

12.3. Finance and Banking

Banking and finance have a high abstract nature and need to combine models for organizing the visual explanation of conclusions based on quantitative analysis and for the representation of reports at different hierarchical levels. Mainly, reports are at high hierarchical levels and thus very abstract. Nevertheless, the purpose of the bank is to provide a safe place to pay for functions by receiving deposits or access to payment instruments. Therefore, it will sell services in exchange for fee income. This concept arises directly from accounting and the need to analyze and explain the significant financial statements. Budget revenues and expenses, statutes, and even staff personnel must be regularly assessed to review deviations. The bank concept as a service organization, possessing spending structures, demanding budgets admitting balances, explains that funds cannot be disbursed renewably unless there is certainty that they are reinvested in secure forms of placement and collection. Therefore, without a well-founded income budget with an acceptable high reserve, it will not be possible to expect deposits. The bank cannot be a construction and industry employee issuer and plauder, must be a serious credit institution and provide a good level payment instrument since it owns the fund for the stability of such paper as public.

Because the amount of information must be huge, it seems quite appropriate to model such a bank system in a multidimensional conception of accounting, and there are different data marts generally reflecting financial statement lines. Accounts contain vertical functions or notifying so that it is seen whether it allows or not to see the accounting processes that lead to economics results. Queries allow analyzing revenues, expenses, balances, and budgets by groups, per clients, and by agencies. Dimension tables can reflect budgets, agencies, clients, operations, and subjects. Budget accounting and expenses allow analyzing bank economics as a public organization. Center and agency accounting allow centralized detailed analysis.

13. Tools and Technologies for Dimensional Modeling

Dimensional modeling was gathering momentum in the 1990s. There was a push for dimensional data architecture creation as a part of large application systems to support data mining, online analytical processing, and drill-down visualizations. It has moved from being the domain of specialized high-performance vendors to being built into most mainstream database vendors. It was a major motivation behind the early dimensional modeling software tools becoming available, as users were looking for general-purpose data warehouse development tools. The IT staff needs tools that will simplify their warehouse building efforts, and business users need tools to assist them in their data explorations. There are vendors of generic but relatively unsophisticated data warehouse building tools, others with focus on one specialized area of activity, specialist software and services groups offering business tools for the warehouse, and more mainstream vendors offering a general-purpose data exploration and reporting tool.

Many data modeling tools treated one data model type as a variation of another. Tools were primarily entity-relationship data modeling tools. Others were more network and relational database record-based applications generator focused. These modeling tools had add-ons that plug directly into the export schemas of the modelers to convert their logical schema designs into wire scripts for a given database schema and to generate the tables and columns definitions from the data dictionary. Other tools had similar import scripts to further enhance, as-needed, the physical and logical schemas in their data modeling environments.

13.1. Data Modeling Tools

The data modeling tools for dimensional database design, unlike the contemporary visualization tools which allow easy exploration of data to understand past business operations and predict future runs, are not many. With the growing popularity and acceptance of dimensional databases due to the efficiency in handling business analysis processes, there are more demands for implementation of the dimensional modeling properties in tools to make the work of administrators easy. In this chapter, a few data modeling tools are presented, which use different techniques to achieve this goal. These tools already implement some of the techniques discussed in several chapters. Some of them are just data dictionaries, and a few are wizard-based tools. Though these tools considerably reduce the design time, they are not a substitute for the designer.

The designer still has to examine the final design to ensure the usability and true representation of the subject area.

Data is a collection of facts, meaning that is uncertain until joined with context. Data expands and becomes rich with meaning when it is modeled. Particularly true for business data, its structure and semantics become more congruent with intuitive understanding and use when it is modeled. Good modeling tools have the following properties: They make it possible to present raw data in a way that resembles how it may be seen, i.e. the presentation will help the discovery of data idiosyncrasies and realities behind the data; They allow some validation of the modeled data, to lessen the impact of its initial state on the presentation of the data; They help in reducing the complexity overload, hiding from users some modeling and performance principles; They allow managing structures for all data domains. Typical users are modelers and end-users or end-user groups.

13.2. Database Management Systems

In the early days of computerized data processing, DataBase Management Systems (DBMS) were used to store and retrieve transactional data. During the late 1970s to 1990s, using forward engineering techniques, the entity-relational methodology was developed. This methodology's aim was to help model both the data structures representing business functions as well as the dimensional structures, which were used for decision support. Over the years, this methodology has evolved into many DBMS products used for operational data. Nowadays, many of the remaining commercial operational transaction DBMSs are relational. However, we still find in the market products that are hierarchical or network based DBMSs. However, we forget that for some areas, like very large data or extremely rapid data access, these products were excellent. In addition, these operational DBMS products often include functionality like recovery from power failure, integrity constraints enforced by triggers, etc. These functionalities are quite difficult to implement in the dimensional design without having data redundancy.

An online transaction processing (OLTP) DBMS is very different from a decision support DBMS. Transaction processing environments involve a large number of very short transactions. A typical transaction consists of a read, update, delete, or insert on a small number of records at a time and executes in a fraction of a second. Alternatively, the transaction processing system may require the execution of a long query, which reads a large number of records at a time which runs on a batch-routine basis. Data in OLTP systems change frequently, and the relationship between these large numbers of records is crucial for the correct working and speed of the system.

14. Future Trends in Dimensional Design

Although the dimensional modeling process has remained relatively unchanged for several decades, it is now at the beginning of another renaissance. As the DW/BI industry continues to mature, there are both new technologies and old technologies becoming more mainstream that are changing the way we think about dimensional design. The next few years will see a fundamental change from appliance-like data warehouses based on hard-coded ETL processes tied to data marts to real-time data warehouses made up of many transactional data storelets coupled with semantic publishing and community-like data access to analytical and reporting applications. With the new technologies come new methods and themes of dimensional design. The appliance-like model is still valid because of its ease of use and stability; however, an appliance-like approach can only succeed when organizations are willing to sit and wait months or years for the technical experts and the IT organization to produce results. Companies are seeing and demanding results faster, and are looking to move BI from the back office to encompass the entire enterprise. The model that combines a balanced incomplete block design for dimensional analytics and reporting with the dimensional marketplace paradigm along with low-latency – and, in some cases, not-quite-real-time – data feeds to dimension tables will enable organizations of all types to develop a culture of data-centric decision support. This new culture empowers end users to execute cusp use cases that are specific to their analytical and reporting needs in a timely, data-informed manner.

15. Conclusion

The necessity of quality information and the speed at which it is delivered is increasing. The pressure on the data processing community is daunting. The industry has yet to realize that the best way is to build the structure of the data so that the data can be compressed to its lowest terms, reorganized and accessed quickly so that the customized reports can be processed to meet the demands of the data. There must be a redistribution of resources so that more resources can be utilized to build the structure of the data. Until that happens data processing will continue to be an expensive commodity. It is the opinion that the data processing community takes a large expenditure and closes its eyes while processing at a rate that must not equal what every financial analyst expects from the agency. The goal should be to build the structure of the data and what the business expert is looking for at the lowest possible data level. It is a proven fact

that knowledge driven analysis will no longer bother with low level data analysis of pie charts, bar charts, and other printed information. Knowledge and business driven decisions will be made based upon the lowest level of data access via queries and desktop tools.

The dimensional design concepts and star schema structure concepts are revolutionary in their approach. On the surface the entire design can be picked apart and found lacking. However, those who make it work will reap the rewards of timely data and be considered leaders in the data warehouse design movement. It is performance and timely data that will prevail above all else. This is more a vision of what the future of data is than a quantitative entity letting its theory be validated.

References

[1] Torabzadehkashi M, Rezaei S, HeydariGorji A, Bobarshad H, Alves V, Bagherzadeh N. Computational storage: an efficient and scalable platform for big data and hpc applications. Journal of Big Data. 2019 Nov 15;6(1):100.

[2] Zhang L, Stoffel A, Behrisch M, Mittelstadt S, Schreck T, Pompl R, Weber S, Last H, Keim D. Visual analytics for the big data era—A comparative review of state-of-the-art commercial systems. In2012 IEEE Conference on Visual Analytics Science and Technology (VAST) 2012 Oct 14 (pp. 173-182). IEEE.

[3] Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. Journal of big data. 2015 Feb 24;2(1):1.

[4] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.

[5] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.

[6] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.

[7] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.

[8] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.

[9] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.

[10] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.

[11] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.

[12] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.
[13] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[14] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[15] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[16] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[17] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[18] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[19] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[20] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[21] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[22] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[23] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[24] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[25] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[26] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[27] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[28] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical

Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.

[29] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).

[30] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.

[31] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.

[32] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.

[33] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.

[34] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.

[35] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.

[36] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.

[37] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.

[38] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.

[39] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.

[40] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.

[41] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.

[42] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.

[43] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.

[44] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.

[45] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[46] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[47] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[48] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[49] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 8: Big data: Governance, Security, and Compliance in Data Management

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

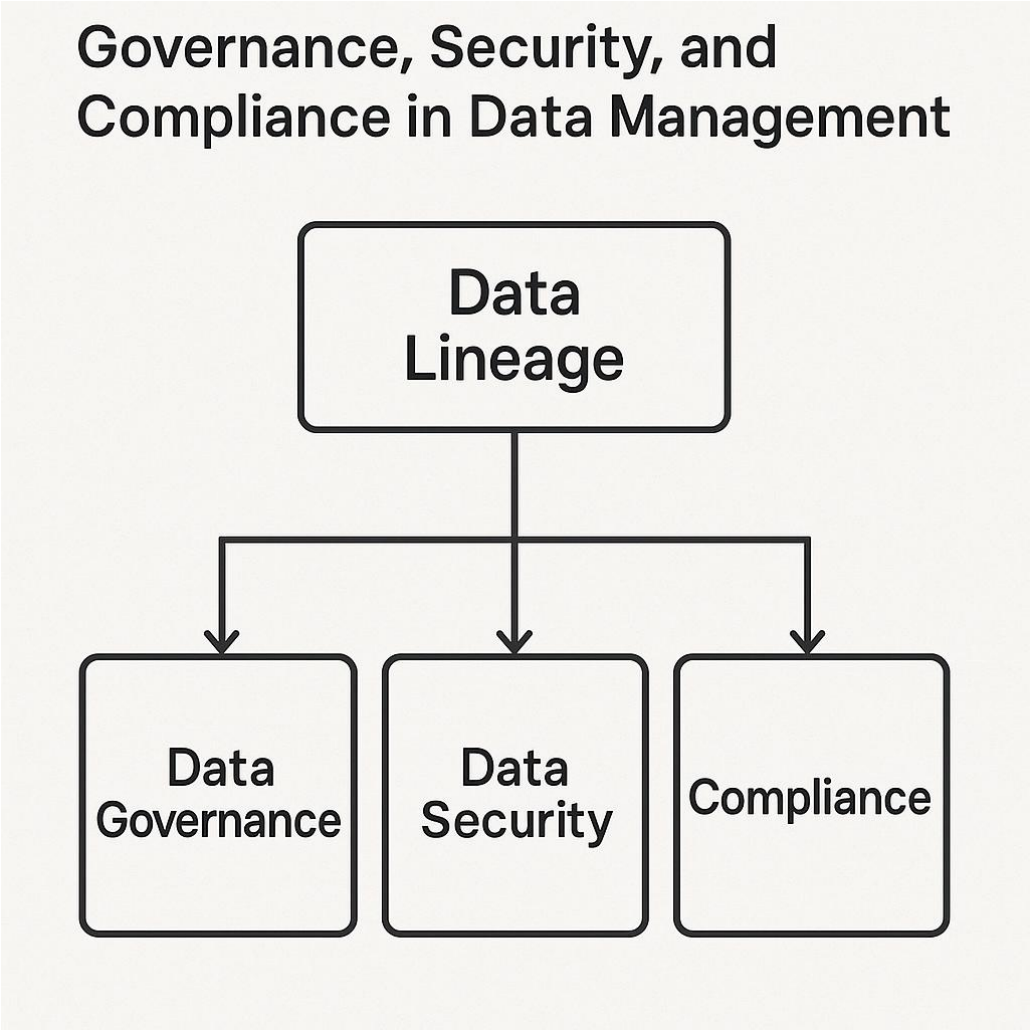
1. Introduction to Governance, Security, and Compliance

Big data is revolutionizing business. Data are created at a staggering rate with increasingly diverse formats. Data pipelines transform data into analytics. Modern organizations are moving blindingly fast to implement analytics systems and accelerate decisions, to produce results from machine learning, to monetize data, and to create value from multiple data sources [1,2]. In many cases, companies are wasting vast treasures of data. As they shift into high gear to leverage data, they are doing so without establishing the rules that provide needed guideposts to make data safe, effective, and meaningful.

What organizations need to achieve is a structure referred to as governance, security, and compliance (GSC) for their data platforms. Governance sets out the framework that guides effort, making data usable and meaningful for analysis, defining the "who, what, and why?" of data to provide clarity for their objectives. Security ensures the protection requisite for the exploration and analysis of data and the systems that store and process it. Compliance enforces and certifies that the structure of governance and the operations of security are being followed.

The GSC concept combines the three disparate elements, usually considered in isolation, into an integrated view. Each of the elements is explained individually, each is also presented from the perspective of analytics in machine learning and artificial intelligence, and finally the interrelationship between the three facets of governance, security, and compliance will be discussed, along with a four-step

approach for organizations to assist them in implementing or enhancing their GSC strategy. By mitigating risks and ensuring adherence to regulations, GSC closely aligns with the monetization of data. Data engines are revolutionary but need an integrated policy for them to operate and produce meaningful results.



2. Data Lineage

Many organizations struggle with issues related to data governance, security, data breaches, and compliance for not having a defined process for tracking corporate data. Data changes over time; its transformation from a primary stage into a finished process or an analysis is often a black box. Many data managers would appreciate help creating a data map for their organization with defined data

origins, flows, transformations, stores, and outputs, which can further be used in regulatory compliance to assist regulatory investigations. Numerous data management solutions track data lineage, build a corporate data map, and help answer complicated questions about where data came from, what happened to it, where it went, and how it can be trusted [3-5]. The goal of this section is to discuss data lineage – its importance and tools that track data lineage.

Data lineage is defined as the information that traces its origin and its movement through time, as well as the data model constructed to represent the app's data. Data lineage provides a long-term view of metadata, giving detailed hour-by-hour, day-to-day, week-by-week, month-by-month data. Metadata identifies the source of the data, the characteristics of the data, and the current owner of the data, but does not address history unless an organization has data that tracks changes to metadata. Metadata may include defined attributes for a student in an educational system. Key attributes might include key identifier, course, credit hours, GPA, student classification, financial obligation, culture, etc. Metadata may also track changes in the defined attributes for a student over time. In many organizations, a regulatory environment has made it mandatory to track the movement, usage, and transformation of sensitive and personal data.

2.1. Understanding Data Lineage

Data lineage refers to the tracking of the origins and subsequent movement, transformation, and realization of data throughout its lifecycle. In simple terms, it tracks and visualizes the flow of data throughout its collection, aggregation, and consumption across various environments, including how that data is ingested, transformed, and processed throughout the development, production, and post-production phases of assessing its quality using algorithms and methodologies [6,7]. Unlike data provenance, which describes metadata and refers to the original source of the data and how the data has been altered, which refers to the data’s transformations and why that data has been altered, typically through some sort of editorial process, data lineage includes digital forensics by tracking, analyzing, and visualizing the data’s journey, which is especially critical for CI/CD models where data is constantly being moved and designed upon. Further, data lineage is typically viewed as one-dimensional and often ignores who is responsible for the transformation, which often leaves open questions of spatial lineage with regards to lineage across local areas and within express committee guidelines.

Data lineage builds on the concepts of the data lifecycle and data flow by providing additional detail and instructions for after-the-fact actions. Understanding the relationship between data inventory and data lineage is also

important; the desired detail captured in data inventory could drive the data lineage. Data lineage consists of two types of lineage: upstream and downstream. Upstream lineage means what data contributed to that record or dataset. Downstream lineage means that record or dataset contributed to what other records or datasets [2,8-10]. With a rich data inventory containing links from records and datasets back to data sources, one can build the upstream lineage. With lineage links built in and pointing downstream, one can build the downstream lineage. In databases, the lineage refers to the compiled version of a program in contrast to the source version which represents its source.

2.2. Importance of Data Lineage in Governance

The paramount importance of understanding the origin of data is the bedrock on which the entire edifice of data governance and regulatory compliance rests. Without understanding where data come from, when they were created, whether they have passed through any transformations, where they are being consumed, who has authority over their accuracy and integrity, and through which systems their ongoing life is being tracked, the entire domain of data governance is a random walk, with every step being made in ignorance of the consequences. Concepts of data quality, information security, and accountability for privacy violations can, and should, be deeply interlinked with data lineage.

There is an ongoing culture clash between data governance and data science, which is at its heart a difference of opinion over the privacy implications of data provenance [1,11-12]. Proponents of data science argue that the value of having more and better data dwarfs the ethical and privacy implications of lacking lineage. However, while governing compliance to rules and policies across complex public institutions built up over decades is a time-consuming activity, it is often rendered necessary by the evolving competence and conscience of public representatives, such that the effort required is always in abeyance and ever-nearing completion. In contrast, the data science and analytics activities of, say, a bank, typically process many datasets every day, and the final impacts, or even intermediate consequences, are usually known or seen only by a small fraction of employees. Data governance and provenance tools must enhance compliance and lessen friction, to the benefit of the organization's employees, its data scientists, and other stakeholders such as customers and the wider public.

2.3. Tools for Data Lineage Tracking

The philosophy of data governance has traditionally a poor practice of securing the business needs analysis, often gathering organizationally irrelevant information. Therefore, data lineage analysis is needed to establish data accuracy,

integrity, and quality. Without the data lineage information, understanding the operational requirements of business data is difficult. To establish communication and trust in an organization, documentation of data lineage processes is essential in preventing and reducing data governance miscommunications [13-15]. Wrangling data lineage is an arduous and painstaking manual effort that places additional pressure on data teams, who are already grappling with a data environment that is growing exponentially larger and more complex. Instead of investing resources in tracking, maintaining, and confirming business rules for every dataset a consumer might touch, organizations will benefit from tooling that automates that work. By introducing central solutions that can track lineage all the way from origin data sources and protocols, organizations can scale this critical process and alleviate the burdens placed on business analysts [16,17]. By offering automatic and manual suggestions to business users to create and review business rules, these solutions can increase data integrity and accelerate the sharing of knowledge. They can also support exploration of company-wide data assets and improve productivity by allowing repeat users to expect and share the same meaning for terms used. Moreover, using AI-based natural language processing, the cognitive burden of authorship can be significantly reduced, increasing the number of business-friendly lineage comments. With machine learning algorithms that track usage on data assets, early tests in co-citation algorithms are promising in predicting business meaning. These intelligent suggestions can increase the velocity of building a foundation of organizational intelligence shared across revenue-generating functions and departments.

3. Data Cataloging

Data catalogs serve as libraries for all available data assets and the relevant information concerning them, being indexes or collations of what metadata is available within the organization and helping both humans and machines find and learn about available data to enhance future data-driven projects. Creation of data catalogs should be a shared responsibility between data owners, data producers, and data consumers. Data producers enrich the catalog with details about the data assets, data owners authorize the catalog data item information, and data consumers exploit the metadata in the catalog to accelerate their insight delivery. Well-crafted, well-maintained data catalogs are akin to a file plan in a document management system, providing a blueprint for where to find organizational knowledge, no matter where it's stored [12,18-20].

Data catalogs provide the capability to discover and understand data assets, validating the confidence in the data used for any analysis and speeding time to insight – ultimately supporting the business at large. The diverse data consumer base has varying requirements for how catalogs assist with these tasks. Data scientists need tool integration and metadata discovery capabilities to accelerate model-building tasks for data preparation and training. Business analysts and data explorers would like the results in a friendly interface that provides intuitive search and presentation and ultimately generates and validates queries to the organization's data assets. Data catalog tools support building trust in their data, which is a fundamental requirement for organizations to realize the full benefits of being data driven.

The Data Catalog service enables data producers and data consumers to share responsibility for documenting and describing an organization's data assets [21-23]. It maintains a centralized inventory of an organization's data sources – whether hosted or external, on-premises – classifying and enriching the details of the inventory items while automatically scanning, crawling, and cataloging findings on a periodic basis, as defined by the required cadence.

3.1. Overview of Data Cataloging

Data cataloging is the act of creating a data catalog – an organized inventory of data assets in the organization that enables data discovery – in which the relationships and other contextual metadata of data assets, and related data domain owners, stewards, and other stakeholders are recorded. Data catalogs help facilitate the work of data scientists, data engineers, and data analysts looking for trusted data assets to use for business intelligence, advanced analytics, and machine learning projects. A data catalog system connects the people involved in data management and data governance with the users who need data to do their work. The quality and usefulness of a data catalog is highly dependent on the availability and accuracy of the metadata used to create and populate it. Metadata includes basic data about the data asset like its name, description, type, format, and location, plus business lineage definitions and formal descriptions of the transformations applied to the data. The best data catalogs also import and maintain additional context including information on data quality issues, changes to the data and related assets, and access controls. Two different types of metadata are used to create a data catalog, technical and semantic, both of which are vital.

Technical metadata describes the properties of the data asset itself – how the data is physically constructed and extracted. Technical metadata for structured data assets includes the schema that defines the data asset. For dimensional or

hierarchical data, the technical metadata includes the attributes and members of the data dimension or hierarchy. For other kinds of structured data, the question "What does it mean?" is answered by semantic metadata, which creates a bridge between the data asset and the end user. Semantic metadata provides business lineage information, including how, when, and who defined the data asset, and formal descriptions of the transformations and business rules applied to the data. This information can be expressed as a data dictionary, but is often captured in business glossary tools that handle business terms, relations, synonyms, and parent-child hierarchies. While business glossaries are very useful, they do not provide sufficient information to act as a replacement for data dictionaries and cannot be the only source of semantic metadata.

3.2. Benefits of Data Cataloging

Data cataloging can be thought of as a data governance tool that enables companies to document their databases and datasets, and provides keywords/tools for easier search and discovery. However, a well-implemented data cataloging solution can do much more than just that [24,25]. In addition to keeping data useful, meaningful, and searchable, data cataloging also enables enterprises to: Provide stewards and stakeholders of the data, and the underlying technology, assets to enable quicker troubleshooting of failed data assets. Data are the new currency for most enterprises. Proper data cataloging, management, and governance of critical data assets makes information readily available, helping to deliver their stated vision of providing the essential information of the company to their employees and stakeholders for timely decision-making. Well-implemented data cataloging can also reduce the likelihood of misuse of data and help to implement Authentication and Authorization of data for the levels of data security and possibly compliance regulation.

Data cataloging is becoming an added storage cost, eaten into the overall cloud economics of IaaS providers. However, it is also a tool available to these providers for data protection and support of compliance regulations that adds to their overall service offering. And finally, data cataloging helps companies deliver real-time visibility into enterprise data initiatives providing ownership, data lineage, and business context to enrich enterprise data while easing the burden on data operations and engineering teams. Hence a focus on and investment in a data cataloging solution can benefit all stakeholders associated with an enterprise's data strategy.

3.3. Implementing Data Cataloging with Purview

A data catalog is a central metadata repository that helps data producers and consumers discover, understand, and leverage an organization's data at scale. A unified data governance solution helps organizations manage and govern their on-premises, multi-cloud, and software-as-a-service data. It combines an enterprise data catalog, data management, and data governance capabilities [26-28]. The Data Catalog enables data producers to annotate, tag, classify, and curate large amounts of data. Data Resource Management provides automated data mapping and management for data regulatory and compliance. Data Governance provides a centralized dashboard for data regulations and compliance.

Creating an account can be the first step to implementing data cataloging. An account is a resource that links various functions and capabilities. You can create an account using the portal, through templates, the CLI, the SDK, or other methods. An account can contain valid namespaces defining a scope to an account. The valid namespace is a storage account containing data, such as data lakes and data warehouses. Once the account is created, scanning is configured, and the Data Catalog is populated. To facilitate automated scanning, the account must have access permission to the namespace.

4. Data Classification

Organizations are required to safeguard regulated data in a manner consistent with its value and sensitivity. A sensible information risk management strategy starts with a solid understanding of data requiring protection and a plan for protecting it. Data classification plays a key role in protecting sensitive data. A data classification scheme is a means of sorting data repositories into levels or categories that designate sensitivity. A classification scheme provides companies a framework to ensure data protection is tailored to the sensitivity of the data. It helps identify ways to protect sensitive data that comply with the law and meet risk management objectives [29-31]. The intent of classification is to understand what data is stored in various repositories so it can be organized, managed, accessed, retained, and destroyed in accordance with its intended purpose and compliance obligations. This is a continuous process of data lifecycle management and engenders truth, accountability, and trust. A structured classification scheme can allow companies to sort through their data assets and employ the appropriate protection measures to meet privacy requirements. Data classifications can reduce compliance exposure by mapping out a company's sensitive data and granting access to that information to only those personnel and

parties who meet established security and confidentiality requirements. A structured classification concept can also reduce service requests, help drive cost alignment, provide a basis for identifying data owners and stewards, prioritize resource allocation, and allow us to identify training priorities better. Rule-based audits of access and usage can help us quickly assess and identify exposure risks.

4.1. Principles of Data Classification

Data classification is an important first step to map security and privacy controls to data management operations. Data classification defines how sensitive data is used and how it should be protected. Classification helps decision makers balance compliance with corporate governance and risk management, as well as provide transparency to customers that their data is secured according to expectations. Frameworks define control objectives that may apply to data [3,32,33]. While the frameworks specify control objectives for data management operations and systems hosting sensitive data, they do not define algorithms to classify data or classify data permanently.

Data classification is hard, due to the absence of concrete and deterministic rules to make classification decisions. Each organization will thus generate their own rules with their own algorithms, based on the data's context and the organization's mission. The fact that data classification is context dependent introduces many complications into the classification process. Not only do data change context over time, but the same dataset may belong to different classifications when viewed in different contexts. Determining when a dataset changes its context, moving from one classification to another, is a difficult task, as is understanding when possibly sensitive data is captured in a different organization's context. Also, data owners may be reluctant to classify datasets containing sensitive data in a way which makes security clear. For all these reasons, organizations typically apply trust-based or risk-based systems of data reputation, for those frequent cases that policies call for.

4.2. Classification Frameworks

A classification framework usually identifies a premise for how various items are classified and oftentimes, the degrees in which the item is classified. A particular framework identifies a range of classifications and concepts for both visualizing the who, what, why, and how of classification while also detailing the processes for actually classifying.

Although classification can be either on-demand or based on a priori properties of the item being classified, it is typically efficient to use a priori properties of an item or object to apply the classification, especially if the classification is to be

semi-automated [4,34-36]. There are two general types of frameworks for data classification; general and vertical. The general data classification framework can be used to develop a multitude of vertical frameworks for specific information needs, such as social networks, news articles, medical records, banking transactions, semantic web data, geography, and others. A vertical classification provides a specialization of a domain within a data type. While we can use one of the vertical frameworks that already have been developed, it is also possible for an organization to design its own vertical framework within these general types based on the items and services being provided.

The top-down approach provides important concepts for the general kinds of items being classified and the concepts contribute one or more characteristics of why items are classified. The physical data classification framework, the privacy and information security framework, and the international framework for the security of information technology resources have been developed for the general classification of data and application items. We can break the balance of general versus vertical into upside and downside, which reflects the general versus specific aspect. The upside to general classification is the "top-down" aspect; there is a wealth of experience from researchers and practitioners categorizing collections of items based on general principals and concepts. On the negative side, the downside of general classification is that there is not enough detail to be domain and problem specific.

4.3. Using Purview for Data Classification

As companies adopt a cloud-first strategy, the need for a unified and comprehensive governance framework becomes paramount. Data Governance is a framework that brings together people, processes, and technology to break down data silos and create a single source of truth [37-40]. Ensuring better data quality, availability, usability, and understanding of data helps organizations make sound business decisions based on accurate datasets. An integrated governance solution allows organizations to connect all their disparate information assets to provide a holistic summary of their data in the cloud, on-premises, and in on-premises environments.

This solution uses machine learning technology and set algorithms to classify data into relevant categories for information security management. Data Classification helps to identify sensitive and business-critical information in your organization's repositories. Data Classification automatically scans and tags information based on a customizable set of labels, both built-in as well as user-defined, and aids organizations in detecting potential compliance risks [4,41,42]. Data Classification uses advanced cognitive capabilities such as Optical

Character Recognition, linguistic detection, and so on, to apply sensitive info types to items in less common languages and that are in image formats. Data Classification also provides tracking and monitoring through a compliance portal. The content explorer and reports will show the progress and status of items scanned in the organization from different sources. Organizations can define jobs to run regular scans to stay updated with the information residing in their repositories and ensure possible compliance-related risks are mitigated.

5. Access Control Mechanisms

Access control governs who can view or use resources in a computing environment and is an essential element of data security, especially to protect information that is governed by compliance regulations. Access control covers a massive number of possibilities: Who can enter the building? Who can request a password reset? Who can print a sensitive report? Who can view, edit, or delete a particular row, column, or cell in a dataset? Who can view a specific dashboard? Who can run a specific dataflow? Who can create a specific pipeline?

Access control is typically layered to allow an organization to enforce security policies at a granular level that matches their needs, while not sacrificing usability and performance at a broader level [43-45]. The two most common layers for access control in data management relate to object ownership and group memberships. When an object is created within a namespace, the system assigns its ownership to that creator and gives them special privileges to determine the object's access rights and permissions. For example, when a dataset is created, it is assigned for ownership to the user that created it. The owner has permission to view the dataset's content, delete the dataset, and assign other non-ownership permissions to other users to view its content. Other access control privileges at a broader level may be implemented by assigning users to groups based on their roles or job functions. Then, the system can assign specific permissions to those groups relative to objects or classes of objects. A user can be assigned to a specific workspace role such as Member, Contributor, or Administrator, which determines their overall access rights to all the workspace objects, although typically at a more coarse-grained level, and then those workspace roles are used for access control for the entire workspace.

5.1. Understanding Access Control

Computer and data systems are usually shared and contain a wide variety of information, not all of it is intended for use by all users or groups of users.

Information systems provide some means of protecting data resources from unauthorized use or modification, but this must be deliberately considered and programmed into the system in some way. Access control mechanisms enforce an information security policy for a computer system and its data and programs, regulating who can use or access specific system resources and in what way. An access control mechanism enforces rules of sharing that the organization’s security policy defines. The services offered enforce rules that the organization defines about the conditions under which individuals may use or change data and programs.

Security policies require that the scope of enforcement be global—a mechanism must encapsulate all access attempts. This includes those from the network but also, in multiuser systems, those from other users on the same physical system. Access control mechanisms consist of two components: policy decisions and policy enforcement. The policy decision is made by a subsystem and may be based on attributes from any source, including user identities, objects being accessed, the roles of entities, request attributes, and so on. The enforcement mechanisms intercept resource requests. For I/O and networking operations, this is usually done at the lowest levels of the appropriate subsystems. These include the network software, file, and database system software for requests to read from a network connection, read from a disk file, or read from a database. For terminal operations and access to user memory, the enforcement must be at the operating system level.

5.2. Role-Based Access Control (RBAC)

5.2. Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) builds upon an authenticated user's identity and associates the identified user with one or more roles. The access control policy determines the operations allowed for the associated roles. A user's roles may be assigned by an administrator, automatically based on rules, determined dynamically, or some combination of these methods. User roles can change over time based on requests to the administrator, based on some operational conditions, or dynamically at each access. Roles in a RBAC system are descriptive of real organizational roles; for example, "accounting analyst" may be a role.

Because roles are so descriptive and typically shared by many users, RBAC mapping of users to roles is a very low effort process, particularly for new users. Also, because the number of distinct roles is usually small compared to the number of users, the list of roles is easily reviewed by administrators for

necessary changes or corrections. Further review is also manageable for conflicts as well, even though the simple rules for avoiding conflicts with roles prevent any automated check of the role-user membership. The first two reasons account for the popularity of RBAC within organizations and the possible simplicity of managing large numbers of users. Once established, however, the flexibility inherent in RBAC may lead to neglect, making it unnecessarily complex. Complexity may also result from simulations of discretionary access control or from implementing RBAC in systems that fundamentally do not use it.

5.3. Implementing Access Control in Microsoft Fabric

Access control in Microsoft Fabric is accomplished through a single role-based access control mechanism that is used across data engineering, data integration, data warehousing, data science, and business intelligence. Access to Fabric workspaces, as well as to the lakes and files that are stored in Fabric's data lake storage accounts, is made using Azure Active Directory. Only Azure AD identities and security groups can be added as members of a Fabric workspace. Different settings allow different levels of access to be granted to the workspace members, giving them the ability to do such things as create Fabric assets in the workspace, modify those assets, or modify workspace settings [9,46-47]. Workspace access determines who can work with which Fabric assets, like dataflows, datasets, reports, notebooks, or pipelines. File permissions take care of access control at a lower level for files or folders that are stored in the data lake. Workspace members need to have at least contributor access to the workspace to be able to perform data operations, like reading or writing a file, on a file stored in the data lake. If a file or folder is not specifically enabled for Fabric, it cannot be used from within Fabric’s assets. The data lake is configured by default to allow all workspace members to read and write Fabric-enabled files. Workspace access is set using Microsoft Fabric's central management tool, the workspace Experience page, or through dedicated RBAC API commands. Folder and file permissions cannot be modified using the Portal. However, users can add or remove Contributor access on any file or folder using storage clients without needing to resort to code.

6. Sensitivity Labels

6.1. Overview of Sensitivity Labels

Sensitivity labels for Microsoft Power BI are an instance of sensitivity labels add-on for Microsoft Purview that provides unified labeling for your Azure services.

This service offers an additional layer of information protection through a simple labeling system that is readily integrated into the native Power BI experience.

Sensitivity labels are created and configured in Purview, and applied to a per dataset basis. These labels are then visible within both the Power BI service and desktop clients. Although most user interactions associated with sensitivity labels are done in either Power BI or Purview, the labels themselves are managed only in the Microsoft 365 compliance center.

When a user publishes a labeled dataset to their PBI workspace, its sensitivity label is synchronized and becomes available for use within the Power BI service, but cannot be modified. The label acts as an owner-set property, which can only be changed by the original owner within the compliance center or Purview. The label is applied to said dataset only, and does not propagate downstream. However, downstream datasets can have their own labels applied to them independently.

When policies such as label retention rules are enforced, labeled datasets within Power BI are audited within the appropriate Microsoft 365 compliance center permissions. Because Power BI is a multi-tenancy service, compliance policies are checked through service principal calls to the respective customer's Microsoft 365 compliance center.

Sensitivity labels are typically used with datasets, but they can also be used with reports. Datasets and reports are labeled using the same configuration, and their use requires no additional consideration. The sensitivity label can be modified after the ownership transfer, and the label settings control what permissions are granted to the labeled content.

6.1. Overview of Sensitivity Labels

Sensitivity labels are a compliance feature that can be used to classify recommended content in apps and services as having certain levels of confidentiality. Sensitivity labels can enable protection actions such as encryption and specific rights. Optional content markings such as watermarks, headers, footers, and colors can also be made visible. Sensitivity labels can also enforce, recommend, or inform users about some policy-selected configurations for features such as DLP, Information Governance, retention policies, AIP Scanner, AIP Label APIs, or AIP Cmdlets. The required Azure AD Information Protection license for sensitivity labels that support encryption and integrated with other features is considered to be included in the suite license.

Sensitivity labels are usable in various enterprise applications and services. In SharePoint Online and OneDrive for Business, sensitivity labels can be applied to sites and document libraries, folders, and files. In Exchange Online, sensitivity labels can be applied to individual emails or to email transport rules, called mail flow rules. In Microsoft Teams, sensitivity labels can be applied to apps. In Microsoft 365, sensitivity labels for content are visible in tooltips and menus, and content can be filtered, searched, and sorted by sensitivity label. Sensitivity labels can be used as part of the ultra-high protection and governance strategy if available with an appropriate enterprise license, and by connectors available in AIP Scanner. Sensitivity labels can be customized with appropriate configurations to suit the organization and to satisfy the data governance, protection, and compliance needs. Sensitivity labels can also integrate with other services such as DLP, Information Governance, Conditional Access, Microsoft Dataverse, Microsoft Intune, Windows Information Protection, AIP, Microsoft Defender for applications such as Defender for Cloud, Defender for Office 365, Defender for Identity, and more.

6.2. Creating and Managing Sensitivity Labels

A Sensitivity label is a tagging construct available to Microsoft 365, Office 365, and Azure DevOps subscription customers who want to serve data protection and information governance needs of their organization. Sensitivity labels allow customers to classify and protect documents and emails for various organization. Unified labeling allows customers to protect their organization's documents. When documents which don't match organization's labeling and protection policies are opened by label users, unified label creates a unique label on the document to identify the organization which governs it.

With organizations that require tight security for specific documents, a single label may not provide the required security. Organizations that don't allow sharing of any data with users outside of their organization, or specific sensitive data, already have policies restricting such actions. But if an organization wants to allow mail flow and sharing of sensitive documents with external users, no label to allow that level of granularity exists. A label that allows users to share payroll documents but not account information would require custom cryptographic support and cooperation from all organizations that stored or opened classified documents. It would require significant collaboration and development effort among all organization that provide cryptographic products and services. For many organizations, customization involving a third party is a last resort. They want to provide users with reliable protection options with a minimum amount of variation and decision-making and without involving a third

party. Unity labels tightly couple classification and specialized sharing control and support such organizations' business needs.

6.3. Integration with Microsoft Fabric

Microsoft Fabric is the new cloud Native Data Platform from Microsoft, designed to unify the organization's data efforts and requires a solid framework for Data Governance, Compliance, and Protection. Data sensitivity is pivotal for any of those achievements and therefore should pervade all Microsoft Fabric components. This is the reason why data classification is integrated with Fabric to help discover and classify the sensitive data of data pipelines. Data Classification is integrated with Data Factory, Data Flow, and Data Warehouse.

Data Classification

While working with an integrated Data Factory, the user can define data classification rules based on data sensitivity. Those data sensitivity definitions will be used to classify the sensitive data being ingested in any Data Factory pipeline, Data Flow, or Data Warehouse Table. Data sources available for validation of data sensitivity definitions are Microsoft SQL Server on-Premises, Azure SQL Database, Azure SQL Managed Instance, Azure Blob Storage, Azure Data Lake Storage Gen 2, and Microsoft 365. Classification rules and the actual data classification will be maintained under data classification.

Data Factory

Data Factory is responsible for orchestrating the Data Ingestion, Data Movement, and Data Transformation Flows. Data Flow is part of Data Factory and is responsible for Data Transformation by executing Data Flows defined by the user. Fabric Data Factory is integrated with data classification allowing Data Factory users to define the Data Classification rules, which will be used by Data Factory, Data Flow, and Warehouse to classify sensitive data during the Data Pipelines execution. Those data sensitivity rules will be used to determine the sensitivity data definitions for data sources maintained under data classification.

7. Monitoring Data Pipelines

Converging enterprise data into a single corporate store that provides a unified view of the resources that a corporation possesses is the main goal of the enterprise data warehouse. Building such enormous corporate stores, known as enterprise data warehouses, necessitates a lot of money, effort, and the collection

of a lot of different kinds of data over many months or even years. The creation of a strategic data repository assists executive decision-making, or provides information that helps to guide the organization and form the basis for business strategy. It aids in achieving a competitive advantage by providing accurate, up-to-date, relevant information that is in line with corporate goals, complements and supports operational systems, is simple to navigate, is quick and easy to access, and presents data in clear, understandable ways through the use of an intelligent front-end tool. Data pipelines are important, as they transport the data to where data teams need them, possibly after a series of transformations, joins, and lookups. Monitoring data pipelines is crucial to understand if the data arriving at the destination is credible and ready to be consumed, and there are some topics within data monitoring that need specific attention. Data observability is the most constrictive and perhaps the most advanced aspect, and is defined as an organization's ability to fully understand the health of its data in order to meet privacy and compliance requirements.

7.1. Importance of Monitoring Data Pipelines

Enriching the combination of ingest, transformation, and storage of data allows data producers to share facts with data consumers, who explore them through dashboards and reports to assist data-driven decisions. The dependency of data consumers on the efficient and effective operation of the data pipeline that delivers trusted, timely, and significant insights on the metrics that matter creates an obligation for data producers to implement monitoring techniques that promote the handling of data pipeline issues quickly and mitigate the impact of negative effects. Reliability is the target of monitoring data pipelines. Monitoring is the process of implementing software and techniques to oversee and check the health, performance, and correctness of data pipelines.

Data monitoring in data management is comparable to network monitoring in network management in its properties and goals, but it is peculiar to the specifics of data pipelines in data management and has some unique guidelines due to its inherent unique challenges. Data pipelines can be long-running batch jobs that ingest, transform, and store data at infrequent intervals or real-time events in continuous streams. Batch processing is less complex, but its related monitoring is still hard because the unique properties of data can make errors difficult to detect and can allow a batch job to execute quickly but still produce bad output. Real-time processing is more complex and requires more effort because it has additional state management and timeliness challenges. Monitoring is complex due to the sheer variety and dynamics of the events in most data sources and the numerous transformations in most data pipelines. Unusual events may occur so

infrequently that humans may not realize their importance and tend to ignore them. It is critical to link data pipeline monitoring to business objectives and align it to the knowledge of business use cases and business metrics.

7.2. Best Practices for Monitoring

One of the first steps towards a successful monitoring strategy is to create a good baseline of metrics and alerts. Based on typical loads and transactions, establish tolerance thresholds for metrics that will form the basis for alerts. Avoid setting up threshold alerts on all metrics, as this will lead to alert spamming. Work with the data consumers and service level objectives users to determine alerts that are essential. Explore anomaly detection-based alerts to give context about unusual situations, such as what is unusual and why it is so. Alerts should be refined over time according to feedback on alerts that were created but ignored as not important.

Regularly check the health dashboards and surface-level exploration capabilities of your monitoring platform. Providing these capabilities to the data consumers will help alleviate the depths of the monitoring stack, as they are empowered to self-diagnose. Best practices for designing the health dashboards and exploration interfaces can easily be derived from the lessons learned from designing similar user experiences for business intelligence dashboards. Be sure to include aggregated overview information and aggregate exploration based on metadata such as time, data source, data set, and format. Use attention-mechanisms to highlight unusual segments of data. Include drill-down drill-through capabilities for inspecting segments more deeply and a guided experience to help users self-diagnose and self-correct issues that do not require escalation and input from the data teams.

7.3. Tools for Monitoring in Microsoft Fabric

Microsoft Fabric native solutions for monitoring its workflows offer two canvas options that introduce monitoring specific capabilities: the Pipeline canvas available in the Data Factory and the Orchestrator workspace available in the Data Integration. The primary difference is that the Pipeline canvas is focused on Data Factory pipelines, does not offer Data Integration specific activities, and is more centered around productivity while the Orchestrator workspace is generating orchestrator specific entities and contains all the integration activities. Using the Pipeline canvas provides an easy access point for Fabric users familiar with Data Factory. Both solutions leverage the same monitoring backend internally.

To optimize workload execution, Microsoft Fabric supports activity runs and job runs monitoring at different conductor levels. Jobs orchestrate the activity execution and contain execution details for the different compute engines available in Fabric. The activities contained in the jobs might be triggered by different mechanisms: manually executed, scheduled, or detected by events. For activities associated with service modes, which might be scheduled or event driven, the orchestration workflow allows to introspect all triggers or events that were previously detected for a specific date range.

Then job structures allow for simple identification of any issues as they contain at a high-level manageability unit the details of different activity runs for all the integration activities contained in it. The failure or retries information for the contained activity runs can be leveraged to identify any timing or data issues during the scheduling of the job and aimed to be reflected in the application workflows. Another interesting characteristic about jobs orchestration data is that it persistently stores activity run statuses. This allows users to identify or reflect historical changes in the orchestrated solution for activity runs that are executed asynchronously or executed as part of batch processes.

8. Securing Data Pipelines

Data pipelines underpin modern data management systems, especially where cloud services manage pipelines on behalf of organizations. Crucially, pipelines often ingest and transform sensitive and personal data, as well as outgoing data products, such as machine learning models, that may carry data leaks. Therefore, it is imperative to ensure that all steps of data pipelines are secure from actors who may maliciously modify, debug, or listen into a running pipeline. In this section, we focus on the security of data pipelines. Securing data pipelines is a specific case of securing systems that connect digital inputs to outputs. Closing the window for potential attackers needs to happen at multiple levels, ranging from securing storage buckets to querying protections, to secure transfer protocols using firewalls and encryption. Below, we first cover common threats to data pipelines and pipeline servicing infrastructure, and give an overview of existing research. Following that, we discuss countermeasures at multiple layers of a pipeline system, spanning by layer models of security, and security measures for cloud services. Finally, we discuss practical examples for those responsible for sensitive pipelines and exit with a commentary on future directions for research and practice to help secure pipelines. In summary, securing data pipelines is critical for defending organizations against attackers who may dusky

personal data ingested or originating from pipelines, or worse, transform it into harmful or weak models. Data management is therefore a community effort, where it is crucial to balance complicated security models with automated vertically integrated research technologies and practical tooling.

8.1. Threats to Data Pipelines

Introduction

Data pipelines are the lifeblood of the modern organization; they are not only responsible for moving the data required to power core business applications, but they also enable access to the vast wealth of data generated often from partnerships, collaborations, and the core systems of record. Unmitigated threats and vulnerabilities to these trusted pathways can pose risks to an organization's overall data integrity and security posture. Threats to data pipelines fit broadly into three categories: operational issues, data and service integrity issues, and general cloud security issues. Threats can manifest themselves at different layers of the stack within each of these categories: source application, data pipeline, data storage and target application. For business users, the risk of information loss and/or impact to live operation can have a significant business impact.

Operational Issues

For data pipelines serving analytic requirements, composed of batch jobs importing to a common execution and/or storage layer, operational issues like job failure and excessive processing latency can impact utility and be the source of high technical debt and cause business analysts to distrust enterprise data pipelines and resultant datasets. For data pipelines serving collaboration business needs, such as those powering collaborative applications like a feed, transactional email, recommendation, peer review, etc., the tail latency of individual item processing can cause a poor customer experience. These data pipelines are often close to a critical business operation and, on occasion, may even directly impact the result itself. For example, content feed postings have their legitimacy validated during a transaction; feed content is generated using a feed data pipeline; delayed posting validation can expose users to spams just before feed content shows up.

8.2. Security Measures for Data Pipelines

Starting with the target of security requirements stated in the previous section, we outline the building blocks of security in data pipelines. We cover security capabilities that data pipeline projects should support. Later, we will refer back

to these security capabilities while describing the architectures of popular data pipeline projects.

Since data pipelines ingest, transform, and route production data, usually involving high volume, velocity, and frequency, data pipeline projects must integrate tightly with the security mechanisms present in proper distributed systems. Both producers and consumers of data processed by the pipelines may belong to external organizations and may not have any trust relationship. In this case, the data flowing through a data pipeline may be sensitive, and its processing may be subject to regulatory compliance. Events generated by data pipelines through auditing may also be sensitive, making their protection necessary. Thus, the data pipeline must provide proper audit and monitoring of security events.

Data pipeline operations can involve sensitive credentials for accessing external data systems and unique topics or queues for high-value data being processed. So, protection mechanisms around tokens and encryption keys with fine-grained access controls need to be employed. It is important to protect the tokens being used to build connections from deployed components to external systems. The security measures must also apply to the code or configuration that is being pushed to the data management engines and worker processes to prevent unauthorized code execution.

8.3. Case Studies on Securing Data Pipelines

Various Data Security as a Service (DSaaS) implementations run by private data companies demonstrate the efficacy of secure data pipelines. We detail two examples in the following sections.

We explore the DSaaS implementation as an example of a hybrid secure data pipeline with centralized management and clustering-based secure data integration capabilities for large datasets. The Spatial-Data Service (SDS) is a large dataset DSaaS solution that allows sharing and accessing large datasets on demand. Built on top of an in-memory data grid, SDS uses access interfaces to load and store data. Data are loaded from a data producer site to SDS by an API-provided data loader. Unlike existing large dataset sharing solutions, which cannot support high-performance environments, SDS can load and store datasets at a high throughput and perform management, data sharing, and publication operations completely remotely. SDS provides the cloud service management system with APIs for automation of the entire life cycle of a spatial-data public cloud service.

We focus on the DSaaS implementation as an application-layer system with bottomless storage and global file sharing as a core to highlight security concerns

on the data pipeline application level and security measures for data confidentiality. Based on the DSaaS, the data exporting organization builds its data pipeline service to provide its authorized partners with the service to communicate confidential large files for internal or external use with remote data pipeline clients. The service will encrypt the files during upload and decrypt them after incoming transfer.

9. Compliance Frameworks

A compliance framework consists of the policies and procedures that ensure regulations, data protection laws, and corporate policies are followed. These compliance frameworks are built on foundational compliance requirements laid out in security and privacy standards. Review of these framework policies and guidelines is an essential process to ensure that company compliance policies align with the requirements set forth by regulatory standards.

Data is an increasingly valuable business asset, and laws have emerged to protect its use and privacy. Additionally, there are new government and industry regulations, as well as specific company guidelines, that apply to the data lifecycle and require organizations to take a proactive approach to address data purpose limitations, transparency, storage duration, location, protection measures, access restrictions, content removal, and audit rights. Any organization utilizing data must employ a compliance framework to ensure both company policies and government regulations are implemented and followed to mitigate risk.

Organizations must ensure that data is stored, processed, and consumed in compliance with the applicable privacy laws and regulations, including data residency requirements. To achieve this goal, organizations can implement data governance and compliance requirements by using tools, services, and features available. This framework is dedicated to helping organizations maintain compliance with the regulatory requirements and guidance frameworks applicable to their products and services. Through the extensive set of compliance resources available, organizations can find the required materials to implement the necessary processes and controls to ensure compliance with their corporate policies and regulatory obligations across their data and analytics solutions.

9.1. Overview of Compliance Frameworks

The content in this chapter is not Microsoft-specific but is extracted from our writing on Compliance Frameworks. Therefore, this chapter can be included as-is in the book. Compliance refers to the process of ensuring that an organization follows the rules of its legal, regulatory, and contractual environment. Compliance with regulations and laws has always been at the forefront of concern for Data Management professionals. With the increase in the volume of data shared across the globe for the purposes of business partnerships, financial transactions, and providing services through technology, the question of trust became critical. Data Partners need to have assurances that the appropriate controls are implemented by organizations that are entrusted with the storage, management, and processing of data. For many organizations, compliance certifications are required to meet basic business requirements for establishing trust with their partners and customers. Furthermore, legally binding privacy regulations place penalties on organizations that fail to demonstrate compliance with security, governance, and privacy laws. Organizations charge third-party compliance bodies to conduct assessments of their controls and the associated technology in order to provide a formal attestation of compliance.

During the past 20 years, organizations have established security frameworks that are considered best practices, and which organizations strive to comply with in order to obtain trust. These best practices include detailed recommendations on Governance, Security, Risk, Audit, and Operational controls for data services. Organizations interested in cloud services seek compliance with these best practice frameworks and commit to performing assessments against these frameworks at regular intervals.

9.2. Regulatory Requirements

The emergence of companies like Microsoft in the 2010s and the confusion and mistrust that resulted from the security and compliance implications of moving business processes and resources to the cloud led organizations dealing with large amounts of sensitive data to mandate burdensome requirements on cloud service providers. Most of those requirements demanded external auditors assess and issue audit reports according to attestation reporting standards. This audit-based compliance developed somewhat independently from both internal security and compliance processes and vendor security assessment processes.

During the early 2000s, various countries developed data privacy/protection laws to regulate the data practices of companies based within their borders. In the EU, the General Data Protection Regulation became effective in 2018 and created

strict data privacy requirements for organizations around the globe that process the private data of EU residents. In the early 2010s, the U.S. Congress initiated and quickly abandoned various attempts to increase the federal protection of personally identifiable information. However, some U.S. states, such as California, have developed their own data privacy laws. The California Consumer Privacy Act became effective in 2020, imposing strict privacy requirements on businesses in California. Globally, more than 150 countries have established data privacy laws, including Canada, Brazil, Japan, South Africa, and various other countries. Security and privacy compliance is a complex aspect of risk management.

9.3. Ensuring Compliance with Microsoft Fabric

Microsoft Fabric security offering supports compliance for the Microsoft Fabric services. With Microsoft Fabric, users can focus on the data analysis and insights aspect of their jobs instead of data governance issues like privacy, security, and compliance. Microsoft Fabric uses the same compliance and security foundations as Office 365 and Azure, which are trusted by organizations all over the planet. Some of the compliance-related features built into Microsoft Fabric that administrators can use are Compliance Manager, the Services Trust Portal, Azure Policy, Sentinel, and the Security Center.

Compliance Manager helps organizations meet compliance requirements with greater ease and convenience in an increasingly complex regulatory environment. These compliance requirements may be specific to an industry, for example, financial services or healthcare. The Compliance Manager also helps organizations meet global privacy and data protection requirements including but not limited to comprehensive national frameworks. Compliance validate configuration actions and manage risk through integrated monitoring and management capabilities. Identify and track compliance data, such as service trust documents and audit reports, while building and maintaining service-level agreements for products hosted in Azure. Sentinel uses security information event management and security orchestration automation response. The Security Center helps protect Microsoft Fabric services with an easy-to-use dashboard. Azure Policy helps manage extensions and configuration in Microsoft Fabric. Centralized policy management gives organizations control over their security and compliance posture.

10. Challenges in Governance and Compliance

The topic of governance and compliance in the data management domain encompasses challenges in organizational policy implementation and adherence, as well as data privacy and protection legislation compliance procedures. The nature of challenges in data management can be difficult to identify in clarity, because organization-specific policies for the management of data assets, such as ownership, documentation, and quality management, among others, are closely tied to external data privacy and protection regulations, that specify how organization employees and consumers interact, as well as where and how economic and organizational information should not be disclosed. However, with several new regulations enacted and enforcement becoming stricter, legislation adherence by organizations is more critical than ever. Further complicating the picture is the launch of several private organizations' crusades against what they consider to be coercive and nefarious data policies.

Although the importance of awareness about compliance with data-related policies and legislation throughout all stages of the data lifecycle is stressed, actually being aware and consequently avoiding the actions that one should or must not take can be a big hurdle to overcome for organizations. While all enterprises are employed by the key principles of related legislation and policies, being able to make the right decisions when faced with dilemmas is where organization success can be leveraged. For example, while employees at a financial institution are likely not allowed to disclose customer information and data to any or all other interested parties, sometimes, an agency may legally demand it in an investigation. Employees not being neither trained nor accustomed to dealing with such situations, notably those that may arise with increasing frequency, may make matters dangerous, both for the organization, often due to litigation, or for society as a whole.

10.1. Common Challenges Faced

Organizations today face a myriad of challenges in governance, security, and compliance in the context of data management. In simple terms, data governance promotes the equitable and responsible use of data, wherein security provides protection against unauthorized access, modification, and destruction, while compliance is adhering to data-related laws and industry-specific regulations. Recognizing relevant GSC problems, challenges, and solutions is a fundamental step in securing an organization's data management practices. In the lives of citizens, whose personal information is being actively monitored and shared, GSC failures can be catastrophic. In other cases, GSC demands and solutions

seem excessively burdensome and expensive. However, GSC problems are not going away. Rapid advances in data processing capabilities, plus increased interconnections via the Internet and cloud computing, often result in a data management environment that exacerbates the problems and challenges of ensuring effective GSC.

Despite the broad significance of GSC problems and potential solutions, robust guidance on how organizations should think about these issues is scant. While there are industry standards and frameworks in the areas of governance, security, and compliance, they are typically industry- or sector-specific, and written from a narrow or specialized perspective. Their limited scope often results in narrow treatments of questions involving the interdependencies among the pillars of GSC. This is important because risk management, and the corresponding stakeholder demands, is multifaceted. For example, enabling technology-driven innovation without reputational damage or serious regulatory penalties requires careful balancing of data access policies and procedures with robust data security measures.

10.2. Strategies for Overcoming Challenges

The challenges listed previously can hinder data governance and compliance levels in data management. However, organizations should strive to overcome them for improved compliance and data management levels. The following few strategies can help organizations overcome the challenges faced. Organizations should highlight the importance of data governance and compliance with tangible examples to overcome resistance from various departments such as business users or the IT department. They should explain how data governance and compliance help in the big picture, reward departments for collaboration, and model the behavior of proactive governance and compliance employees. Organizations should identify which policy exceptions could introduce security risks or contradictory provisions into the policy documents and disallow them. Again, organizations should strike a balance between data governance and compliance policies and organizational flexibility. For example, in a crisis where business users must process data outside of corporate premises, the IT department should not enforce endpoint security access restrictions that could hinder business users' work.

Organizations should make data governance and compliance simple for all departments involved. All data management departments such as data architects, data engineers, data scientists, business users, and the IT department should be consulted to create simple procedures that work for all departments involved. Furthermore, while data governance and compliance are continuous processes,

they do not need to be static. Organizations should avoid overly rigid procedures, checklists, and dashboards with mandatory approvals that inhibit innovation activities undertaken by business units. To solve the problem of unrelated policies, organizations can classify policies into categories and develop a dashboard template that presents only the instructions related to the selected categories. Organizations should extend data governance and compliance policies to include merged or acquired organizations well before the merger's completion. Data governance and compliance teams in charge of the merger should be involved in M&A activities early on.

11. Future Trends in Governance and Compliance

As more data is made available in the coming years, the demand for trustworthy, verified information will only increase. The cost of governance, risk, security, and compliance technology will decrease as more solutions arise and become more widely adopted in organizations of all sizes. Emerging technologies, such as A.I. and automation, will aid organizations in proactively managing risk and creating solutions based on alerting when data is outside of certain parameters, as opposed to merely alerting companies of potential risk. Additionally, there will be a continued shift towards third-party verification of data being secure and compliant. Many companies that have not been subjected to GRC audits are still in their infancy in terms of maintaining GRC, and there will be pressure to catch up. Expect to see heavier regulation in the space as more personal information and sensitive data is processed. Organizations that are A.I. driven and are risk aware will be paving the way for innovation and cost-effective security solutions, the goals of which are to improve confidence in being compliant and secure.

Some foresight into emerging technologies includes advances in the blockchain and quantum computing space. Blockchain technology enables a decentralized, incorruptible, and auditable record of any transactions that can provide real-time insight into transaction chains and can drastically reduce the time and cost associated with audits and document collection. Quantum computing has been publicly unveiling itself, but the actual implementation of quantum-powered computers and processes are currently still only available to large businesses or governments with the budget and existing businesses can leverage into quantum computing. Initially commencing quantum business processes will happen very slowly on a case-by-case basis, and there are understandable risk and security concerns. Once adopted and plans have been put into place, quantum business

processes will greatly reduce data processing time and costs while simultaneously enhancing risk management.

11.1. Emerging Technologies

Many existing Data Management services have already started to embrace some of the most popular and commonly requested technologies to deliver newer, better solutions to users. For example, Blockchain is being used to deliver guarantees for smart contracts, making use of trusted environments. Currently it is fairly complicated to prove that some code was running in a smart contract, given that there isn't a persistent state regarding which users are executing which code. The Trusted Execution Data Chain project works around this problem by creating a Blockchain that relies on Trusted Execution Environments for storing transaction history.

Most of the advanced data processing pipelines (batch or stream) also take advantage of Cloud computing engines for batch processing and stream processing. These engines allow users to rely on scalable and fault-tolerant systems without worrying about maintaining the infrastructure and building the resiliency logic themselves.

Currently, the most commonly requested AI offer is the Language Model as a Service. It was made popular by a product, now delivering an API for its advanced Language Model that will support other usage apart from just chat. At the same time other companies are investing in advanced Language Model for enterprise use, but using a closed technology. Other companies have already started democratizing the Language Model space with an inference API that integrates several state of the art Models.

11.2. Predictions for the Future

Although it is a little hard to predict with certainty, there are a couple of trends developing that will likely have a large impact the way we govern, secure, and ensure compliance to the principles of good data management. These trends are the rise of more data management layers above the operating system, the rise of a new type of operating system that acts as a federated data controller and manager for various data sources housed in various silos, and the processing of more data and more complex and sensitive data by more users across more types of computing systems. In essence, we believe enterprise data will likely become more centralized, formalized, and rules-based as the role of the enterprise architect develops tools for the enterprise developer to create more efficient solutions for the enterprise architect and end-user to realize.

Our prediction is that data management will move up the stack, especially with a rise of low code tools. However, at the same time, there will be demand from enterprise developers and data scientists for more governance and compliance capabilities in the tools that are allowing them to do their jobs. Therefore, a solution marketplace will likely emerge pushing compliance up the stack. However, there will also be the need for a new type of specialization within the Data Management field – the Role of Data Steward, given the enormous challenges that enterprises will be faced with security and copyright compliance both within the business and externally with other business partners. This role will be increasingly important, not just in being aware of internal data governance policies, but actually realizing and overseeing them in practice by enabling the use of specialized tools and model serving-scoring containers for knowledge workers who can be trained to perform specialized tasks but do not need the expertise and knowledge to build enterprise level tools.

12. Conclusion

This chapter presented a compact discussion on some relevant topics to consider when dealing with governance, information security, and regulatory compliance in terms of data management in organizations. Such topics become essential as organizations recognize the importance of their information assets and data as being essential to be properly managed. They may face some problems if they neglect almost 50 years of discussions on information security issues, or if they impose dire consequences to the transgressor in terms of data use, storage, and retention without at least discussing regulatory compliance, what is defined by laws and norms, including which consequences need to be imposed on transgressors. We intend to guide readers through the definition of governance policies, procedures, and assignments, to information security and predictive techniques, to discuss regulatory compliance about data privacy in terms of laws and norms focused on specific countries and regions.

We hope this paper contributes to the initial discussions on these areas: policy definition, information security mechanisms and predictive analysis, and regulatory compliance. In fact, at first sight, these are not often addressed with higher-level interest and scope in the data management or data science communities. Yet, data engineering, data science, business intelligence, big data, and analytics are rising fields. Their professionals must be aware of these issues and are also responsible. Therefore, discussions about data governance, information security, and regulatory compliance have become essential,

especially by guidelines and theoretical concerns. It can minimize future negative consequences for society.

References

[1] Jiang D, Chen G, Ooi BC, Tan KL, Wu S. epiC: an extensible and scalable system for processing big data. Proceedings of the VLDB Endowment. 2014 Mar 1;7(7):541-52.

[2] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.

[3] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.

[4] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.

[5] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.

[6] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.

[7] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.

[8] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.

[9] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.

[10] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.

[11] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.

[12] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.

[13] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).

[14] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.

[15] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[16] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[17] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[18] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.
[19] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[20] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[21] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[22] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[23] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[24] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[25] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[26] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[27] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[28] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[29] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[30] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[31] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.
[32] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.

[33] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.
[34] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.
[35] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[36] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[37] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.
[38] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.
[39] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.
[40] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[41] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[42] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[43] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[44] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[45] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[46] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[47] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.

Deep Science Publishing, 2025
https://doi.org/10.70593/978-93-7185-151-0

DeepScience
Open Access Books

Chapter 9: Big Data in Finance, Healthcare, and Retail Industry

Swarup Panda
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

1. Introduction to Big Data

The notion of data is not new as any object, event, or entity can be represented by data in the physical as well as the virtual world. Today, with access to corporate databases and near-infinite storage support, a tremendous amount of data is available almost effortlessly [1-3]. Companies have become increasingly aware of the usefulness of data, which actually is a byproduct of routine operations. Many organizations are putting in place systems and infrastructure to capture such data and be better equipped to face and work with the inherent challenges of managing big data. Therefore, we first explore what exactly is big data? What are its characteristics? How is this different and special compared to traditional data and systems? How some of the buzzwords around big data such as the data-fication of society came to life, and more importantly, how industries are beginning to change, and leverage the power and potential of big data. Big data is generally characterized in terms of the V's, and currently, there is neither a complete list of the V's nor, in many cases, agreement upon the specific names that describe characteristics. Some common ones are volume, variety, velocity, veracity, and value. Data deluge is good only if it translates into special, meaningful information, which can help society take important decisions, commercial or otherwise. Hence, the increasing focus on the 'Big Information' concept and how it can help advance industries, and increase the quality of life.

In this chapter, we primarily focus on large sets of historical transaction data available in structured forms, usually stored in relational databases. With the rapid advancement in data storage technologies, we are witnessing the possibility

of sustaining, and more importantly, making sense of huge mountains of data currently sitting in enterprise warehouses, and across diverse locations and sources, all across the virtual world. We also briefly describe areas and permissions in which new, unconventional forms of data are attracting considerable interest and are likely to make an impact.

2. The Importance of Big Data in Modern Industries

"'Big data' combines two popular topics: data and the contemporary interest in using it to enhance productivity. For every company that has gradually made progress in working through the three phases of data economics—ad hoc reporting, management dashboards, and predictive analytics—I can point to at least another three that still struggle with the first phase of developing data warehouses [2,4,5]. Despite this uneven progress, it's hard to deny that the Internet has transformed business and the world into a giant laboratory, fuelling the growth of both data, the new fuel of the economy—and the new revolutionary discipline of data science. The modern influx of data puts at the disposal of companies and countries the most powerful of tools available to enhance productivity, and this promises to be the greatest fuel of growth for the long-undertaking global economic recovery.

Experiences have been reported in the past about the application of advanced decision-making technology to areas such as telecommunications, banking, transport, and urban development. With regard to banking and finance, these applications have mainly focused on decision problems related to asset and equity management. On the other hand, the healthcare industry is growing and becoming mature in the implementation of technologies for monitoring the vitality of patients remotely [6-8]. Such technologies constantly relay the responses and physiological parameters of patients to computers that are able to compare them with pre-configured alarms and alert physicians in charge anytime there are anomalies to be dealt with. Finally, in logistics and retail, the integration of big data with data science methods can foster impactful changes and business improvement."

3. Big Data in Finance

The global financial services industry has been one of the earliest adopters of technology, deploying massive computing resources to handle high-volume financial transactions, create detailed financial models to manage portfolio risk, and provide services for regulatory compliance. Financial institutions have increasingly begun to adopt new technologies to increase efficiency, provide better services, and improve risk management [9,10]. Customers are becoming increasingly digital, mobile, and complex with evolving preferences and expectations, and financial institutions have to provide them with innovative solutions built to suit their needs. With intense competition and opportunities from nontraditional players emerging, financial service firms need to invest in customer-centric business models. The sheer agility of fintech firms that can quickly target specific customer segments and provide shipping velocity, transparency, and a better experience poses more than a risk to incumbents. They are building partnerships with large financial institutions and tech companies to introduce radical changes to the industry.

Big data solutions allow financial institutions to overcome challenges faced due to high data variety and volume. By leveraging data from different online and offline sources, financial institutions today can target customers in a more personalized way. This is increasingly enabling the industry to solve for the data silos that have made it difficult to identify cross-selling opportunities and provide customers with cheaper and faster product solutions [11-13]. This chapter includes a literature review on predictive analytics in finance, with potential future research directions. The tools and techniques of data analytics are also illustrated through a case study in retail banking. The case describes how a bank used predictive analytics to optimize customer campaigns relating to the cross-sell of consumer credit, in the retail bank product area.

Fig 1. Importance of big data

3.1. Overview of Financial Data Analytics

Data analytics primarily revolves around predicting what is going to happen next in order to allow the organization time to react. This prediction is achieved using historical trends and correlating them with other readily available sources of data. Financial forecasting, budgeting, and modeling are the most common varieties of predictive analysis employed by organizations. Predictive analytics considers data that show the significant influence on the past behavior to obtain relevant future predictions. Small events that cause minor variation in historical data may drive substantial future prediction volatility as businesses operate on very narrow margin rates. Banks deal with diverse risks that require them to maintain discipline in their lending policies and also demands implementation of a disciplined approach toward financial planning. The overarching objective of financial planning is to maximize shareholder wealth. Banks have to concentrate

on sectoral growth while retaining reasonable profitability within a risk framework. Their profitability is based on managing the balance of profitability relative to risk effectively and maintaining a relatively low rate of non-performing loans [2,14-17]. Predictive analytics techniques increase forecasting accuracy for all users in the organization, with a particular emphasis on the areas. The results can influence bank strategy because it is the unique banks that conquer their individual markets and extend their area of influence. The prediction has a high impact on working capital policies, should be driven by the predictions of any empirical state of the art predictive analytics techniques, and influence production and sales topics. As banks venture into more sophisticated areas of business, the use of better forecasting should be based on the assumption that the banking institution is involved in a local competitive market.

3.2. Risk Management and Fraud Detection

Regulatory and customer demands for sophisticated fraud detection coupled with large amounts of communication, transaction, and historic behavioral data make big data analytics an appealing solution for fraud detection. Several companies view prevention of card-not-present fraud, transaction risk modeling, and detection of money laundering and other suspicious activity as top priorities. By analyzing patterns in historical transaction data from both fraudulent and valid customers, banks can successfully identify probable credit card theft [9,18-21]. A particular area of concern in financial services is securing non-face-to-face transactions, which require a bank to validate a new payment method before it can be used without challenge. An example would be payments initiated from a newly-set up account to buy recently posted luxury items, especially if sent to a vendor not especially known for quick shipping, or to an unusual country, where sellers typically ship to local buyers. The risk sensitive nature of transactions also contributes to banks taking a varied and selective approach to protecting and securing the electronic payments ecosystem. Traditional payment types, including debit card purchases and ATM transactions, are not typically subject to risk decisioning at the time of the transaction. After all, most debit card purchases utilize a personal identification number, making them no more susceptible to fraud than the busy bank branch transaction.

Merchant category and transaction details also affect the risk decisioning process. For example, some cards and banks view airline transactions as low-risk while others consider them high-risk, so those decisions are made in the aggregate. Using big data analytics, banks will aggregate credit and debit transaction amounts by merchant and merchant category as an additional way to automatically identify an application as suspect. Credit card accounts tend to

show a particular, if not similar, spending history, including both typical number and dollar amount. A transaction that differs outside of that range could be flagged for questioning. A bank will also compare the purchase amounts of accounts in those various categories, particularly with stolen amounts in the prior period. While no risk model can guarantee that the right decision will be made at the right time, a proper risk decision model will ensure that the best decision possible can be made at the time of the transaction.

3.3. Customer Insights and Personalization

Customer relationship management has been of utmost importance historically for banks. It is driven by the transaction-based insight banks possess on their customers and also what they wish to achieve through banking relationships. Banks aspire to trigger customer loyalty through personalized experiences, custom products, unique services, and consistent engagement. The best way to build loyalty is to develop an emotional relationship, and personalization is a key improvement area to achieve this [22,23]. Advancements in analytics capabilities offer banks the ability to effectively implement customer-centric strategies that encourage increased customer satisfaction and loyalty, thereby enhancing business profitability.

Banks now possess multiple functions that drive customer engagement across the entire organization from marketing and sales to product management and customer service. Data can come from multiple sources, including account data, sales data, survey data, customer support data, compliance data, competitive data, risk data, and product portfolio data. Banks can leverage predictive analytics on these multiple data types to assess customer needs around product functions and features, derive insights into customer preferences and product usage, address product portfolio management, and answer customer queries with respect to market developments. Predictive analytics enables banks to identify the likelihood of campaigns being successful, segment customer bases based on buying patterns, modularize products and services for customization, and learn variables that influence product acceptance.

There are four aspects to personalized engagement – the recruitment of customers, the retention of existing customers, the growth of customers with respect to value maximization, and the communication with customers. Banks can personalize engagement by using predictive analytics to segment customers based on life stage, behavior, and needs [24-26]. Then they can suggest relevant products, trigger offers leading to more cross-selling, upselling, and cross-channel selling. They can implement policies for enticing customers to respond to marketing discounts/offers at optimal levels, select customers more likely to

switch to a competitor bank, and enhance customer service through multi-channel integration at key touchpoints. Deploying artificial intelligence along with predictive analytics and design-thinking principles can lead to the development of captivating and unique personalized experiences for customers.

3.4. Case Study: Predictive Analytics in Banking

With the growing acceptance of predictive analytics where structured and unstructured data are adopted for predictive models that are used to support business decisions, numerous success stories have started appearing in various domain areas. Within the finance sector, predictive analytics has been widely accepted for its big data insights. Within the banking sector, predictive analytics is being increasingly accepted for its ability to provide business insights and thus taking business decisions, particularly in areas like sales, marketing and customer relationship management [27,28]. Predictive models in banking attempt to answer various problems such as forecasting bank failure, classifying good and bad loans, acceptance or rejection of loan applications, bankruptcy prediction, customer retention, and so on.

Banks have rich sets of data available with themselves such as customer's financial history, loan and deposit balances, age, income, marital status, credit rating, existing liabilities, financial ratios, and usage of other bank products. The data collected over certain number of years can be used to model life consisted and diverse portfolio of customer business and product lifecycle variables that models the relationship between the variable of interest and other potentially influencing factors in the model for analyzing and predicting customer behavior, for decision making, customer relationship and profitability evaluation apart from providing the future status of the portfolio. Predictive analytics is being widely accepted as a support document for use by line management in decision making. In the last few years there has been one of the biggest revolutions in the area of retail banking, mostly due to the entry of private banks amongst increasing competition and information technology revolution. Predictive analytics is increasingly being acknowledged as the most powerful strategic tool in retail banking, used to gain insight of customer preferences for the effective and efficient sales and marketing for the customized product and service offerings to the customers.

4. Big Data in Healthcare

The healthcare industry has a typical Big Data problem in terms of data volumes, velocities and variety. Healthcare is an increasingly data-rich environment and the interoperability of disparate data sources can be of high clinical value. Key data sources include electronic health records, medical imaging and sensor continuous monitoring data, social media, genomics, and biomedical literature. Technological advances in areas such as the internet of things and AI have the potential to spur the health data revolution, making real-time streaming of diverse data sources possible every moment for each person [19,29-31]. Healthcare data variety ranges from very structured data such as demographics and lab results to semi-structured and unstructured data such as clinical free text in notes and medical imaging. Driven by the growing legislative focus on the exchange of patient data, there is growing opportunity for Health Information Exchanges that aggregate clinical data across healthcare providers and companies that are building data aggregation capabilities.

Within the healthcare industry, there is an increasing trend to have providers invest in companies that provide digitization services that make old data digital. Tying together these disparate sources and transforming the raw data posed by long tail medical codes into useful information is a daunting challenge motivated by the range of efficacy that different treatments and the appropriate interventions can have on diverse patient populations. In addition to rare fatal interventions, different medication timelines have widely varying outcomes across the population with respect to efficacy, side effects and cost for conditions such as depression, bipolar disorder, and psychoses.

4.1. Healthcare Data Sources and Types

Advances in technology, the Internet of Things (IoT), and digitalization have transformed healthcare systems, making available unprecedented amounts of data from many different sources. Important types include clinical and administrative data collected during daily hospital operations from electronic health records, health information exchanges, remote health monitoring, radiology images, personal electronic devices, business services and finance, treatment efficacy from clinical trials, and data in public health databases, including those on diseases and risk factors, deaths, and healthcare access and utilization [32,33]. Electronic health records, for example, are the digital version of a paper chart, providing information regarding the condition and treatment history of each patient—including patient demographics, diagnoses, medications, immunization dates, allergies, lab test results, and radiology images—collected

through contact with various healthcare providers involved in care. Health information exchanges, in turn, allow doctors and hospitals to access and securely share a patient's vital medical information—wherever and whenever the patient is treated—to make informed decisions about care.

The amount of data available from clinical and administrative sources is so large that it is often referred to as big data. The differentiating characteristics of big data related to clinical and administrative data are the large size, complexity, and variety of the data, due to the diverse types and sources, the sensitivity of the data, and the increasing speed of data generation, which can make data rapidly pass the point of utility. As is common in many application domains, the size and complexity of the data have created challenges related to how best to store, manage, and analyze the data for effective decision making [34-36].

4.2. Improving Patient Outcomes with Data

A report stated hospital costs consumed 30% of the US gross domestic product; of this amount, $700 billion was spent on nonproductive activities. For example, about $75 billion was spent on excess hospital readmissions. Using data to identify at-risk populations and design interventions can accomplish a triple aim: strengthen the patient-physician relationship, avoid negative patient experiences, and achieve short- and long-term budget effects. Data are being used to help healthcare providers improve their patients' overall health and individual outcomes in various ways, including by predicting which patients will develop health problems like diabetes; identifying the patients who are not likely to comply with treatment regimens, thus leading to poor outcomes; and creating guidelines to help practitioners choose the most effective interventions or therapies.

With the release of innovative healthcare initiatives, healthcare payers and providers are harnessing the power of data to improve outcomes for both the patient and the healthcare organization. Predicting clinical outcomes, such as hospital readmissions, can help guide treatment plans for patients at risk for insufficient recovery and long-term function [37-40]. Many other healthcare services, not limited to hospitals, can use data analytics to identify patients who will incur high costs due to adverse medical events. Hospitals are busier than ever, with more than 36 million annual visits, and up to 20% of these hospital discharges may require readmission. Within a month of discharge, another sizeable proportion of patients are likely to visit the emergency room with similar issues, causing vital resources to be diverted and decreasing the standard of care.

4.3. Operational Efficiency and Cost Reduction

Healthcare organizations are now interested and focused on investing in Big Data analytics to help them streamline their operations, reduce costs, and cut fraud and waste. These companies create vast amounts of data and information, collect and store them at a high velocity, at a low price, from internal and external sources, coming in a number of different formats and present it on a day-to-day basis. These massive amounts of stored data are purposefully transformed into information intelligence to make operative strategic decisions [41-43]. Making improved short- and long-term decisions can help reduce the resources and costs organizations use while hastening the path for medical innovations to ease the pace of discovery and reduce their time to digital deployment.

The decryption and analysis of the large data assets generated in the healthcare domain can assist reluctant hospitals and healthcare organizations to advance a scientific understanding of the pathways of wellness, to object directions for prevention. Additionally, intelligent detection of inefficiencies today can systemically help to make those efficiencies operate in fact. Reducing logistics, supply chain, and management issues can help achieve the ultimate goal of healthcare: to provide better care to patients using fewer resources while ensuring both patients and healthcare providers appropriate the maximum benefit to be derived from positive health outcomes. Analyses of large data assets derived from the historical actions of patients and hospitals in the province of health support large facility-based hospital systems in making healthcare practice more adequate. The possible examined returns rewards in narrowing practice, decreasing costs, and advancing mitigation of medical mistakes [28,44-47].

In factoring all the benefits into decision making, hospital administrations and integrated health systems have a better opportunity to shift resources toward the most effective treatment and improve care delivery and practice. This prospect might lead connectivity and ease policies to have a tilted effect on the least-linked subjects and heighten area margins without impacting payback and quality. Data-driven decision making facilitates current and planning in a rational approach. In so doing, digital and other forms of body structuring promotion-supported data decryption, managerial reaccreditation, and spot boosting may result in enhanced performance of hospitals and integrated systems, both digitally and in quality.

4.4. Case Study: Data-Driven Decision Making in Hospitals

Hospitals encompass a plethora of departments that offer acute, emergency, inpatient, or outpatient care. Multiple departments are engaged in providing specialized care; for example, a labelled dataset in the hospital may involve

available hospital beds of a specific type in each department. A decision support system should actively interact with hospital users, who might be doctors, nurses, bed managers, or even patients. The data-driven decision support uses advanced data analytics methods, fluidly handles unpredictable chaos, and respects doctors' intent. Hospitals are large, complex systems open to external termites: small bugs could cause a big change in the future.

Applying data analytics to advance hospital performance is quite popular with many examples readily available [48,49]. Data-driven decision support could also help hospitals during natural calamities. Empirical differences among and within Florida's counties warn decision-makers who would like to invest in hospital infrastructure not to trust the simple average, although statewide results could be advantageous to make the right overall choice. After all, data-driven decision making could be considered a modern buzzword that everyone wants to believe in. Data-driven hospital decision support should thereby not be understood as outsourcing decisions entirely to machines.

On the other hand, decision support based on big hospital data is a very serious matter as the consequences of wrong models or policies may be disastrous and life-threatening and not just economically costly. Warning signals address this illusion of easy decision making with an avalanche of research papers addressing exciting applications.

5. Big Data in Retail

Big Data has had an enormous impact on the retail industry. Intensive use of sensors, transactions, text/audio/video/images, and mobile apps allows retailers to create large and complex datasets that shed light on every aspect of the retail business: consumer online-offline behavior, footfall, product sentiment and reviews, buying patterns, social media discussions, stock levels, supply chain status, location traffic, delivery times, return rate, breach of security, etc. When analyzed and interpreted with sophisticated tools and domain-specific algorithms, these datasets reveal insights into customer experience, engagement, and satisfaction (both positive and negative), sales forecast, product shift, fulfillment network optimization, outage detection, brand loyalty, and new-product adoption, resulting in improved bottom lines [3,50-52]. In short, the Big Data phenomenon is changing the paradigm of how the retail business functions. Exploration and Explanation of Outcomes: In Retail, understanding consumer behavior is fundamental to the success of the business. For many retailers, a

critical area of focus is improving customer engagement and satisfaction with marketing messages that are pertinent to their consumers, and timing and offering thoughtful recommendations regarding products that consumers wish to buy. As retailers twine together their extensive datasets on consumer interactions (both digital and offline), preferences, purchases, and sentiment with others' online, app, social, and review data, they can create comprehensive consumer profiles. A very important part of Big Data is unstructured data in the form of reviews, social discussions, blogs, and so on. Used properly, this data provides granular insights into consumers.

5.1. Consumer Behavior Analysis

In contemporary times, consumers have become exceedingly reliant on technology. Shopping itself has evolved as a technology-driven act with users depending on devices and their capabilities all through the shopping experience, from searching for the product to making a purchase and even beyond. However, traditional shopping has not faded away entirely; a vast majority of consumers continue to visit brick-and-mortar stores to make purchases. Their shopping experience is dependent on external factors and individuals such as the appearance of the store, product placement, sales promotions, and salespersons. Since the inception of the retail industry, businesses have sought to analyze the behavior of consumers for varied reasons, but mostly to increase sales and identify successful promotional strategies to influence consumers. Nevertheless, it wasn't until the advent of big data technology, along with the rapid adoption of social media, smart devices, and wearables, that consumers started generating a quantifiable influx of data. As individuals interact with devices and technology more than ever, they also furnish retailers with extensive information on their preferences and inclinations, which results in large volumes of complex data being generated.

The availability of such a vast amount of data has been a catalyst for new analytical methodologies that can glean insights from this data. It has also helped researchers develop a more comprehensive understanding of both in-store and online shopping behavior. Retailers handle both online and physical channels simultaneously; therefore there is a necessity for more extensive consumer behavior analysis that incorporates the two retailing locations. This also elaborates on the need for cross-disciplinary research in the area. Incorporating cross-cultural analysis is another requirement that researchers should incorporate into their work. In addition, retailers also manually conduct observational studies to better understand consumer behavior, for instance, by analyzing how consumers interact with their products in their display windows; whether

consumers follow or react to billboards or advertisements; or whether consumers utilize the product once they purchase it and how.

5.2. Inventory Management and Supply Chain Optimization

Inventory management has always been an important part of every retail business. The efficient management of a store's inventory is crucial for the smooth functioning of that store. Efficient inventory management can help a business to increase its sales, lower its overall inventory costs, and save time. With the advancements in big data technologies, retail businesses have now started using big data and analytics to optimize its inventory. Using big data, retailers analyze the foot traffic in stores to answer questions, such as how many consumers enter the store each day, how many visit the store on weekends, and how many visit the competitor's store. These answers help them to predict how much inventory they need for the store so that customers do not leave empty-handed. Retailers can then adjust the merchandise accordingly. As for the supply chain part, the management of a supply chain has become extremely complicated. There are a large number of players involved, including suppliers, manufacturers, distributors, retailers, sales agents, and customers. There is an increased pressure on suppliers and manufacturers to increase the efficiency of the supply chain. The increased complexity has lowered the overall efficiency of the supply chain. Traditionally, supply chain demand has been a projected expectation. As quick access to data related to the demand has become easily available, retail businesses are now trying to optimize the functions involved in the supply chain by eliminating bottlenecks, forecasting demand, anticipating seasonal spikes, using dynamic pricing, and increasing supply chain collaboration using big data.

5.3. Personalized Marketing Strategies

Personalized marketing aims to tailor products and marketing campaigns to individuals based on available customer information. The increased involvement of technology in almost all aspects of our lives has also resulted in a proliferation of data related to our online presence. Customers' search preferences, social media presence, buying history, bills, and bank transactions all contain useful information about their likes and dislikes which can be used to provide personalized, relevant, context-based marketing messages to the customer. Online retailers have proven the success of personalized marketing through their recommendation engines. It is interesting and informative to believe that a significant percentage of web customers did not mind providing their information to retailers who offer customized deals.

The recommendations made by these online retail leaders are based on models or collaborative filtering and are primarily limited to recommending products to customers. However, other companies have also started to take it a step further in their customer engagement analysis and have initiated several offline efforts to create more personalized messages for customers. For example, some companies have started implementing geo-marketing for their retail stores. There are cases reported about how close a company can actually get to the customer with their marketing strategies. Locally owned shops have also been seen to engage in hyper-targeting and request referrals or talk to consumers favorable to the store for their marketing means. The success of these marketing strategies depends on the accuracy of prediction models; thus, organizations should closely monitor these campaigns for their effectiveness over time.

5.4. Case Study: E-commerce Analytics

E-commerce platforms gather vast information through interactions with customers and vast numbers of products that get streamed in and out every minute. Managing this dynamic environment requires very timely decisions, and understanding the content of the input data clearly can reap tremendous advantages for differentiating companies. Such data and analytical needs will differ broadly across the different functional areas of website architecture, online marketing, and product optimization. Moreover, every online retailer must have certain fundamental activities in place. The e-commerce industry is comprised of organizations that do business over the internet.

Using analytics, data aggregators can collect and provide relatively easily available business metrics, such as number of unique visitors, number of page views and cross-page click streams. These metrics can facilitate the making of decisions in areas such as site design, media planning, new product introductions and website mining. Furthermore, organizations can utilize data mining techniques to apply log data to understand multi-session user behaviors, content recommendation personalized services or website fault detection. The Internet is designed for information processing with high degree of interactivity and possibly without geographic restrictions; thus, it is well suited for service industries that have highly perishable output of intangible services.

E-commerce products and services are more and more information products and services such as online advertising, e-stores, online banking, and e-insurance. This chapter addresses the use of analytics and data mining techniques to promote website design, to advertise, sell and serve customers better, and to enhance the clarifying services and product design or improve decision making ability. Our focus is on gaining operational insights for specific functional marketing,

product, customer services activities or cross-customer activity coordination and optimization.

6. Challenges of Implementing Big Data Solutions

Big data has a lot of promise for improving decision-making, predicting the future, and uncovering new insights in business. However, there are several fundamental challenges associated with big data solutions. Companies might not have thought about these issues when using traditional business intelligence solutions with smaller data sets. The consequences of failure for big data initiatives can be staggering. Big data projects are even more challenging than typical IT projects. Big data solutions are likely to be the ‘biggest’ ever attempted, in terms of sources, volume, complexity, technology, timeframe, organizational involvement, and so on. Big data projects are also more likely to fail than regular IT initiatives. We discuss some of the difficult challenges that must be solved to achieve big payoffs from big data.

Data privacy and security concerns are heightened for big data compared to traditional analytics. The sheer amount of data associated with big data can minimize organizations' concerns about data privacy or security at the level of individual customers. Users do not expect the same lack of accountability that a smaller data subset or a single point of access would provide. As a result, both organizations and consumers have increased sensitivity to authenticity and accuracy for data presented in the aggregate. Indeed, any modifications to data privacy or security protocols made by a company that owes customers a fiduciary duty to protect their information can have significant implications. A breach of data privacy or security requirements can result in lost customers or a serious tarnishing of a product's brand label.

Integrating big data solutions with legacy systems can be an obstacle to adopting big data in practice. Most organizations already use IT infrastructure and data systems to transfer, act on, or even analyze existing data, such as historical models or trend analyses in decision support systems. Adjusting these infrastructures and systems and retraining employees to respond to and take advantage of new data from new sources can take considerable time and resources. The presence of legacy systems can also deter organizations from trying to use big data to change the way they make decisions.

6.1. Data Privacy and Security Concerns

Data privacy and security issues have become paramount with the implementation of Big Data, and they concern all stakeholders. Starting with users, exposing highly personal data may result in the occurrences of negative or even horrific events, such as highlights in different news reports regarding companies. Cyber-attacks on these companies, among others, are an indication that securing user data is not easy. In the digital age, protecting user data privacy is resulting in regulations at both national and industry levels. For example, in the European Union, an agency is responsible for network and information security, and it released a Good Practice Guide for Threat Modeling. Another regulation to add more data privacy and security is the General Data Protection Regulation. In the U.S., the Health Insurance Portability and Accountability Act protects patient health information. A specific rule of this regulation defines the data privacy and security of the protected health information. Similar data privacy and security regulations are being established in other countries as well.

While regular data security breaches could expose all user data, it is even worse for Big Data breaches, in the sense that even anonymization techniques will provide low privacy guarantees with Big Data datasets. This is the case due to the easily accessible auxiliary datasets on user or patient data in different Internet sites, including social networks, which make it easy to deanonymize data in these datasets. As a result, it is easy for either cyber-criminals or Big Data authorized users to target and steal personal information from the public, including medical, financial, or shopping preference information. Cybercriminals can even go a bit further, and after passing some of the security barriers present, either physical or electronic, they can corrupt entire organizations systems and steal information.

6.2. Integration with Legacy Systems

Companies have spent large amounts of capital investing in legacy systems over the past few decades. These systems have been central to meeting key business needs and have grown sophisticated over time in supporting critical enterprise operations. Yet, as organizations look to leverage big data technologies, they often find a gap between their existing legacy systems and the new big data solutions. Some of the advanced services offered by big data technologies sit on top of traditional business propositions already supported by legacy systems. Many use these systems for extracting intelligence that helps manage the business. For example, banks use their legacy transaction systems to help detect fraud. Retailers use inventory systems to help optimize product availability and reduce inventory carrying costs. Manufacturers use product design and production planning systems to help manage product quality and reduce costs.

As enterprises seek to enhance their operations with new technologies, they want to use their investment in legacy systems together with the lots of possibilities that new solutions can offer.

Big data systems are designed with different architectures, query execution methods, and storage mechanisms than those of legacy systems, and integrating the two can be a challenge. Missing is the sophisticated and nuanced data query capability that these enterprise systems support. For operational big data questions, these legacy systems capture a lot of critical information over a long time horizon spanning hundreds or thousands of different data fields associated with different customer or product attributes or specific business operations. Beyond querying, these legacy systems are used for providing enterprise information in a continuous manner, and what is gathered is maintained in an authoritative manner, for use by many different functions or units across the enterprise. Work that is important to the business is performed continuously, using enterprise initiatives for updates, the processes also serving while helping to ensure data quality. Without any advanced integration capability, companies often feel compelled to simply build special purpose interfaces among the various data repositories being tapped for information about customers, products, or business processes. Such a piecemeal effort can become problematic, creating a fragile web of one-off interfaces that take time to build and soon grow obsolete.

6.3. Data Quality and Management Issues

Many organizations are struggling to connect many different systems and formats of varied, sometimes questionable quality—but the sheer volume of data flowing from those channels allows the application of Big Data tools that can quickly vet for accuracy and usability some of the extremely large datasets problematic to work with just a few years ago. The challenge involved isn't a matter of size, then; it's the management of that data—what it means, how it can be treated, and how it fits in a larger regulatory structure. A unified understanding of data in terms of semantics and business context will go a long way toward addressing many of the problems that stand in the way of robust, responsible decision-making from large analytical platforms. The importance of Semantic Reasoning is significant. Reasons to integrate may be as simple as geographic or timing considerations that would make it more practical to analyze multiple large datasets together rather than separately.

Creation of better tools and panels for monitoring change and current performance measures along with Big Data can keep those datasets useful, thus making them easier to integrate with newer data as old assumptions and models associated with older data categories require updating. It is important to

remember that the quantity of data is not intrinsically valuable, especially as organizations are beginning to realize that past data only have limited bearing on future performance; moreover, significant investments are being made in predictive analytics tools. Analysis of larger, historical datasets can pinpoint those times and types of events where earlier predictions would have warranted frank sentiment from the original data about how unlike other changes this new development seems to be.

7. Future Trends in Big Data

As Big Data continues to develop and grow both in use and importance across a range of industries, major technological trends promise to increase the range of applications and the impact of data. This chapter discusses some of the major areas of growth in Big Data.

Artificial Intelligence and Machine Learning Recent years have witnessed a tremendous acceleration in the fields of Artificial Intelligence and Machine Learning. This has been due to several trends, including the availability of large amounts of data, many of the advances in algorithms, and the availability of cloud computing with its large amounts of cheap computing power. Algorithms like deep learning have produced amazing results in vision applications, such as image recognition and some speech applications. Machine learning is now being widely adopted for non-Traditional AI applications, like recommendation engines and bid management for Internet advertising. For big data, machine learning increasingly must be provided at scale and in real time, meaning that the data must be provided as the machine learning models are being trained, with no batch processing delays while models are trained ahead of time.

Real-time Data Processing A major trend in data is the requirement that it be actionable in real-time. One of the boundaries of Big Data is that it frequently is processed in batches. While this processing is done, the data becomes stale and less useful. Businesses often require updates on a 24/7 basis. Today this includes all electronic commerce applications, where stock levels and pricing must always be current. This also includes social media applications where users expect to see only recent comments. Many other business applications are beginning to require online processing, including operational risk management and financial transaction analysis. There are advances in technology, both in terms of hardware and in software, which should allow more data to be processed in real-time.

7.1. Artificial Intelligence and Machine Learning

The term Artificial Intelligence (AI) refers to the capability of a machine to imitate intelligent human behavior. The term itself was coined sometime during the summer of 1956, and it actually means that machines display a certain kind of intelligence that is not dissimilar from human intelligence. AI systems use various concepts of computer science, neural networks, natural language processing, robotics, and Machine Learning (ML). AI and ML today do not realize the expected results because of chip shortage and ethical standards related to their use. But new trends related to privacy, security, sustainability, and fairness that are reshaping consumer opinion are expected to complete the remaining pieces of the puzzle. In particular, interpretable ML, in order to build high-performance productive databases, AI fairness, AI ethics, and considerate AI seem to have an important role in AI and ML developments.

Databases, indeed, have the duty to embody into their design such concepts in order to be the background of AI and ML experience for consumers and to assume the related consequences of their development. From an implementation perspective, we base our research on a probable collaboration of new hardware paradigms, operating systems, programming languages, and new machine-oriented engines in order to allow an efficient and productive functioning of ML/AI algorithms. To this perspective, we propose to use special-purpose logic nanoparticles for high-performance data processing. Emerging developments provided by quantum computing, neuromorphic chips, innovative optical technologies, and human brain simulation would be included as prescriptive techniques for FPGA-oriented neural networks.

7.2. Real-time Data Processing

As previously mentioned, big data is not only about the volume of data generated or the tools available to process it. Today's companies are increasingly interested in processing and analyzing high volume, often unstructured data streams containing real-time events where the value of the information changes with time. Classic big data processing tools cannot provide the required sub-second processing latencies to offer real-time insights. On the other hand, high throughput, low latency data analysis platforms are very much in demand but still require efforts in configuring them both for speed and convenience.

This demand cannot be underestimated. For example, consider the financial services industry: about 10 million transactions worth around $860 billion happen daily in the US alone. Even if 99 percent of the transactions are processed in less than 1 second, in the worst case scenario, a $43 million transaction can

take a full second to process. As a result, the financial services sector is devoting considerable resources to identify those few transactions that are most likely to signal fraudulent activity. It is equally important to share this information in real time with other related entities, such as the recipient's bank, in order to minimize the loss incurred to those involved. If a transaction is initially flagged as suspicious but not cancelled within a short period of time, the recipient's bank is left with the task of verifying its legitimacy after the fact. It then has to rely on help from external sources if, for example, the credit card used by the recipient was issued by a bank based in another country.

7.3. The Role of Cloud Computing

At present, cloud computing is considered as the best environment for storing and processing big data, providing the most notable characteristics such as scalability, pay-as-you-go model, and manageable IT resources. A broad range of cloud computing models are proposed and launched in practice, e.g., Infrastructure-as-a-Service, Platform-as-a-Service, and Software-as-a-Service. Infrastructure-as-a-Service provides virtual physical machines or distributed computing infrastructures hosted in data centers to users according to their needs of computational resources without explicit regulation. Platform-as-a-Service offers a platform enabling users to develop, deploy, and run applications without the complexity of building and maintaining the infrastructure typically associated with developing and launching web applications. Software-as-a-Service is exposed by third-party providers by hosting applications and making them available to users on the Internet.

Though minimally effective for less intensive tasks, cloud computing offers a potentially cost-effective solution by deferring and consolidating the costs associated with a particular task. Cloud computing can shift big data processing tasks—including data away—while allowing externally developers to focus more upon the application logic. Consequently, cloud computing can cover the resource gap in big data processing and support larger and more complex applications. Furthermore, issues relating to security, data access latency, and development lifecycle can diminish the prospect of these new infrastructure. Despite the complaints about security in this storage paradigm, offloading the data management task into a third-party cloud provider eventually increases the Big Data accessibility to a broader range of users and developers.

8. Comparative Analysis of Case Studies

The intersection of big data with humanities and social sciences has prompted the creation of new applied fields such as digital finance, computational healthcare and retail analytics. While research in these areas has grown rapidly, much of the focus has been on proofs-of-concept. Consequently, research in these areas lacks in-depth case studies. We have filled this gap by investigating three case studies in the finance, healthcare and retail sectors of industry. In this section, we summarize lessons learned from these case studies.

While the case studies are about specialized topics, they exhibit common themes as well. Big Data Solutions combine computational methods for processing, analyzing and visualizing various types of data streams with domain knowledge about how these methods can improve various HSS problems. In addition to technological requirements such as massive parallelism, scalability and fault tolerance, the deployment of Big Data Solutions should address realistic business considerations such as efficiency, operationality and business value. Furthermore, specialized domain knowledge is essential both for transforming research prototypes into operational systems and effectively using Big Data Solutions for decision support. Though data-centric Big Data Solutions can be potentially useful in every HSS domain, they may only be of limited use unless they meet the aforementioned business and domain requirements. In what follows, we summarize insights from each of the three sectors of industry. Each summary proceeds as follows. First, we discuss key takeaways from the case study, particularly in terms of the business and domain requirements for Big Data Solutions. Next, we discuss lessons learned about transforming research prototypes into operational systems and using these systems for decision support.

8.1. Key Takeaways from Finance

This chapter investigates the opportunities and challenges associated with big data in the finance industry by studying three diverse and disparate cases. We find that big data technologies and their applications are transforming the finance industry along three levels – social network analytics are transforming how risks are assessed, visual analytics tools are enhancing trading pattern recognition, and predictive modeling is reshaping the methods through which professional traders and private investors predict stock movements. The intersection of these three technological enhancements has the potential to democratize finance, providing tools for more individuals to compete with well-resourced investment banks and traders. However, this same democratization is providing institutional investors

and hedge funds with the same big data tools that individual retail investors are gaining access to.

Given the aforementioned challenges, we conclude with lessons learned, opportunities available, and further research proposals. The finance industry has always been strongly data-driven; however, it has typically used smaller, structured datasets to analyze stock price movements and computerized algorithms to determine buying and selling behavior. In contrast, new tools, techniques, and platforms are enabling the industry to collect, curate, visualize, and analyze very large and unstructured datasets to identify and predict price movements for stocks and other assets in the financial market. These datasets include news articles published worldwide, international and domestic social media websites, and corporate press releases. These big data sources are being employed, often in real-time, to supplement and enhance the price-sensitive data curated daily by stock exchanges.

8.2. Lessons Learned from Healthcare

At first glance, one might assume that healthcare would be particularly well-suited to the incorporation of big data tools, approaches, and techniques: both the willingness of government and patients to invest in collection of extensive amounts of patient data, as well as the desire of researchers to study the complex relations between patient data and diseases, would seem to make the introduction of advanced forms of statistical analysis little more than a logical step in the natural evolution of the field. In addition, the involvement of entire sectors of the economy in the provision of healthcare, and the presence of sophisticated financial mechanisms such as insurance, would further lend support to such advanced analysis being a natural and important aspect of the domain. Nevertheless, although there have been a number of successful applications of big data techniques to healthcare, our research reveals two major sources of difficulty in incorporating data-enabled decision making and the decision support that it brings.

First, healthcare is characterized by the presence of heterogeneous patient populations that often exhibit very different underlying behaviors and business processes, eschewing any standardized analysis and decision-making regime, and causing attempts to fit a tidy analytics framework to fail. Decision-making in healthcare normally does not follow a standardized, sequential procedure that allows for guideline-based provision of care; rather, decision-making may be invoked deterministically or stochastically at very specific times and not at others in very different decision trees generating care for patients. More importantly, these differ across patients being treated. This has resulted, in a number of cases,

in high-profile failures of naive predictive analytics to derive optimum clinical decision regimes for specific patient classes.

8.3. Insights from Retail

In retail, big data facilitates a deeper understanding of customers' preferences in product categories, price sensitivity, information content and timing of promotions, geo-temporal shopping patterns, etc. In turn, these insights help design better new products, optimize pricing and promotion, and improve marketing effectiveness and store operations such as product placement, product availability, and staffing. However, protecting data privacy of millions of individual consumers remains a huge challenge for retailers.

We first study the online environment, where an astonishing 15% of the global retail sales was made in 2021. Capable of collecting rich shopping behavior data, e-retailers optimize the consumer experience using recommender systems powered by big data analytics. Although there are many commercial implementations of recommender systems in use, research literature is still active and insightful. We first consider two-stage recommender systems called “recommend-then-rank” systems. In this setup, a small number of items is suggested to consumers based on collaborative filtering. Depending on consumers' clicks on the suggested items, a ranking of the items is determined using a ranking model. Two-stage recommender systems are popular in practice because collaborative filtering can only work at scale for low-dimensional feature representations of items, which publishing houses can afford. The success of these recommender systems hinges on the performance of the ranking models.

We also discuss assortment optimization, which refers to the problem of selecting product variants to stock in each store, to strike a better balance between satisfying demand and avoiding waste during the selling season. The challenge is that demand is influenced by competition and occurs at the store/local market level, while the optimization model needs to be implemented at the supply chain level. Items requested by consumers may differ when they shop online without physical constraints, or brick-and-mortar stores with finite shelf space.

9. Conclusion

The beginning of the 21st century saw the advent of a new era in virtually all sectors. As jobs in factories were increasingly replaced by machines, humans began to work in offices; and since much of this work involved managing and

processing information, the availability of computers made such work easier and faster, considerably decreasing the need for human work. Over time, more and more work came to be done online, by consumers and businesses. Because of this, soon the possibility of collecting information on all activities became a real one. This would lead to data being generated on an unprecedented scale, leading to the creation of a new concept: big data.

As big data becomes more and more common, and is also generated in larger and larger volumes, industries are investing in analyzing techniques to exploit it. This is resulting in the increasing automation of tasks, as well as a positive impact on the bottom line of many companies. In certain industries such as finance, the capabilities of big data are being assembled to build systems that use it to automatically and continuously monitor financial markets, buying and selling securities at very high speeds. In healthcare, there are systems that analyze data from large sets of patients in order to help physicians make better treatment decisions. At retail companies, information generated by consumers when searching for products is combined with social media information to design better marketing campaigns. The trend is for the involvement of big data systems in industries to only increase in the coming years.

References

[1] Ajiga D, Okeleke PA, Folorunsho SO, Ezeigweneme C. Methodologies for developing scalable software frameworks that support growing business needs. Int. J. Manag. Entrep. Res. 2024;6(8):2661-83.

[2] Salloum S, Dautov R, Chen X, Peng PX, Huang JZ. Big data analytics on Apache Spark. International Journal of Data Science and Analytics. 2016 Nov;1(3):145-64.

[3] McPadden J, Durant TJ, Bunch DR, Coppi A, Price N, Rodgerson K, Torre Jr CJ, Byron W, Hsiao AL, Krumholz HM, Schulz WL. Health care and precision medicine research: analysis of a scalable data science platform. Journal of medical Internet research. 2019 Apr 9;21(4):e13043.

[4] Barga R, Fontama V, Tok WH, Cabrera-Cordon L. Predictive analytics with Microsoft Azure machine learning. Berkely, CA: Apress; 2015 Aug 19.

[5] Abourezq M, Idrissi A. Database-as-a-service for big data: An overview. International Journal of Advanced Computer Science and Applications. 2016;7(1).

[6] Demirkan H, Delen D. Leveraging the capabilities of service-oriented decision support systems: Putting analytics and big data in cloud. Decision Support Systems. 2013 Apr 1;55(1):412-21.

[7] Nothaft FA, Massie M, Danford T, Zhang Z, Laserson U, Yeksigian C, Kottalam J, Ahuja A, Hammerbacher J, Linderman M, Franklin MJ. Rethinking data-intensive science using scalable analytics systems. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 631-646).

[8] Baldominos A, Albacete E, Saez Y, Isasi P. A scalable machine learning online service for big data real-time analysis. In2014 IEEE symposium on computational intelligence in big data (CIBD) 2014 Dec 9 (pp. 1-8). IEEE.
[9] Talia D. A view of programming scalable data analysis: from clouds to exascale. Journal of Cloud Computing. 2019 Feb 11;8(1):4.
[10] Sandhu AK. Big data with cloud computing: Discussions and challenges. Big Data Mining and Analytics. 2021 Dec 27;5(1):32-40.
[11] Panda SP. Artificial Intelligence Across Borders: Transforming Industries Through Intelligent Innovation. Deep Science Publishing; 2025 Jun 6.
[12] Selvarajan GP. Leveraging SnowflakeDB in Cloud Environments: Optimizing AI-driven Data Processing for Scalable and Intelligent Analytics. International Journal of Enhanced Research in Science, Technology & Engineering. 2022;11(11):257-64.
[13] Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of parallel and distributed computing. 2014 Jul 1;74(7):2561-73.
[14] Dai HN, Wong RC, Wang H, Zheng Z, Vasilakos AV. Big data analytics for large-scale wireless networks: Challenges and opportunities. ACM Computing Surveys (CSUR). 2019 Sep 13;52(5):1-36.
[15] Panda SP. Augmented and Virtual Reality in Intelligent Systems. Available at SSRN. 2021 Apr 16.
[16] Shivadekar S, Kataria DB, Hundekar S, Wanjale K, Balpande VP, Suryawanshi R. Deep learning based image classification of lungs radiography for detecting covid-19 using a deep cnn and resnet 50. International Journal of Intelligent Systems and Applications in Engineering. 2023;11:241-50.
[17] Rane J, Chaudhari RA, Rane NL. Data Analysis and Information Processing Frameworks for Ethical Artificial Intelligence Implementation: Machine-Learning Algorithm Validation in Clinical Research Settings. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:192.
[18] Elshawi R, Sakr S, Talia D, Trunfio P. Big data systems meet machine learning challenges: towards big data science as a service. Big data research. 2018 Dec 1;14:1-1.
[19] Berisha B, Mëziu E, Shabani I. Big data analytics in Cloud computing: an overview. Journal of Cloud Computing. 2022 Aug 6;11(1):24.
[20] Yang A, Troup M, Ho JW. Scalability and validation of big data bioinformatics software. Computational and structural biotechnology journal. 2017 Jan 1;15:379-86.
[21] Jannapureddy R, Vien QT, Shah P, Trestian R. An auto-scaling framework for analyzing big data in the cloud environment. Applied Sciences. 2019 Apr 4;9(7):1417.
[22] Ranjan R. Streaming big data processing in datacenter clouds. IEEE cloud computing. 2014 May 1;1(01):78-83.
[23] Panda SP. The Evolution and Defense Against Social Engineering and Phishing Attacks. International Journal of Science and Research (IJSR). 2025 Jan 1.
[24] Shivadekar S, Halem M, Yeah Y, Vibhute S. Edge AI cosmos blockchain distributed network for precise ablh detection. Multimedia tools and applications. 2024 Aug;83(27):69083-109.

[25] Mohapatra PS. Artificial Intelligence and Machine Learning for Test Engineers: Concepts in Software Quality Assurance. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:17.
[26] Wu J, Rohatgi S, Keesara SR, Chhay J, Kuo K, Menon AM, Parsons S, Urgaonkar B, Giles CL. Building an Accessible, Usable, Scalable, and Sustainable Service for Scholarly Big Data. In2021 IEEE International Conference on Big Data (Big Data) 2021 Dec 15 (pp. 141-152). IEEE.
[27] Saif S, Wazir S. Performance analysis of big data and cloud computing techniques: a survey. Procedia computer science. 2018 Jan 1;132:118-27.
[28] Ramakrishnan R, Sridharan B, Douceur JR, Kasturi P, Krishnamachari-Sampath B, Krishnamoorthy K, Li P, Manu M, Michaylov S, Ramos R, Sharman N. Azure data lake store: a hyperscale distributed file service for big data analytics. InProceedings of the 2017 ACM International Conference on Management of Data 2017 May 9 (pp. 51-63).
[29] Potla RT. Scalable machine learning algorithms for big data analytics: Challenges and opportunities. J. Artif. Intell. Res. 2022;2:124-41.
[30] Hu H, Wen Y, Chua TS, Li X. Toward scalable systems for big data analytics: A technology tutorial. IEEE access. 2014 Jun 24;2:652-87.
[31] Mrozek D. Scalable big data analytics for protein bioinformatics. Computational Biology. 2018.
[32] Mohapatra PS. Artificial Intelligence-Powered Software Testing: Challenges, Ethics, and Future Directions. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:163.
[33] Panda SP. Relational, NoSQL, and Artificial Intelligence-Integrated Database Architectures: Foundations, Cloud Platforms, and Regulatory-Compliant Systems. Deep Science Publishing; 2025 Jun 22.
[34] Chandramouli B, Goldstein J, Quamar A. Scalable progressive analytics on big data in the cloud. Proceedings of the VLDB Endowment. 2013 Sep 1;6(14):1726-37.
[35] Bharti AK, NehaVerma DK. A Review on Big Data Analytics Tools in Context with Scalability. International Journal of Computer Sciences and Engineering. 2019;7(2):273-7.
[36] Pandey S, Nepal S. Cloud computing and scientific applications—big data, scalable analytics, and beyond. Future Generation Computer Systems. 2013 Sep 1;29(7):1774-6.
[37] Chowdhury RH. Cloud-Based Data Engineering for Scalable Business Analytics Solutions: Designing Scalable Cloud Architectures to Enhance the Efficiency of Big Data Analytics in Enterprise Settings. Journal of Technological Science & Engineering (JTSE). 2021;2(1):21-33.
[38] Panda S. Observability in DevOps: Integrating AWS X-Ray, CloudWatch, and Open Telemetry. International Journal of Computer Application. 2025 Jan 1.
[39] Shivadekar S. Artificial Intelligence for Cognitive Systems: Deep Learning, Neuro-symbolic Integration, and Human-Centric Intelligence. Deep Science Publishing; 2025 Jun 30.
[40] Wang X, Guo P, Li X, Gangopadhyay A, Busart CE, Freeman J, Wang J. Reproducible and portable big data analytics in the cloud. IEEE Transactions on Cloud Computing. 2023 Feb 15;11(3):2966-82.

[41] Miryala NK, Gupta D. Big Data Analytics in Cloud–Comparative Study. International Journal of Computer Trends and Technology. 2023;71(12):30-4.
[42] Demirbaga Ü, Aujla GS, Jindal A, Kalyon O. Cloud computing for big data analytics. InBig data analytics: Theory, techniques, platforms, and applications 2024 May 8 (pp. 43-77). Cham: Springer Nature Switzerland.
[43] Yilmaz N, Demir T, Kaplan S, Demirci S. Demystifying big data analytics in cloud computing. Fusion of Multidisciplinary Research, An International Journal. 2020 Jan 21;1(01):25-36.
[44] Singh D, Reddy CK. A survey on platforms for big data analytics. Journal of big data. 2014 Oct 9;2(1):8.
[45] Panda SP, Muppala M, Koneti SB. The Contribution of AI in Climate Modeling and Sustainable Decision-Making. Available at SSRN 5283619. 2025 Jun 1.
[46] Mohapatra PS. Artificial Intelligence-Driven Test Case Generation in Software Development. Intelligent Assurance: Artificial Intelligence-Powered Software Testing in the Modern Development Lifecycle. 2025 Jul 27:38.
[47] Panda S. Scalable Artificial Intelligence Systems: Cloud-Native, Edge-AI, MLOps, and Governance for Real-World Deployment. Deep Science Publishing; 2025 Jul 28.
[48] Muppala M. SQL Database Mastery: Relational Architectures, Optimization Techniques, and Cloud-Based Applications. Deep Science Publishing; 2025 Jul 27.
[49] Warren J, Marz N. Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster; 2015 Apr 29.
[50] Babuji YN, Chard K, Gerow A, Duede E. Cloud Kotta: Enabling secure and scalable data analytics in the cloud. In2016 IEEE International Conference on Big Data (Big Data) 2016 Dec 5 (pp. 302-310). IEEE.
[51] Panda SP. Securing 5G Critical Interfaces: A Zero Trust Approach for Next-Generation Network Resilience. In2025 12th International Conference on Information Technology (ICIT) 2025 May 27 (pp. 141-146). IEEE.
[52] Rane J, Chaudhari RA, Rane NL. Data Privacy and Information Security in Deep Learning Applications: Risk Assessment and Patient Safety Protocols for Big Data Analytics. Ethical Considerations and Bias Detection in Artificial Intelligence/Machine Learning Applications. 2025 Jul 10:54.