International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Real-Time Big Data Stream Engineering for Smart Logistics Optimization

Sathiri Dhanaraj

Independent Researcher, India

dhanrajsathiri@gmail.com

ABSTRACT: The logistics and supply chain domain is a hotspot for Internet-of-Things (IoT) applications and Big Data. Logistics processes are increasingly instrumented, resulting in the generation of massive amounts of data. For humankind, this is an opportunity to achieve sustainability and efficiency increases through the development of Smart Logistics. The Big Data stream processing field offers tools to process data in near real-time and/or at large scales. Yet, the requirement for Smart Logistics is not merely the processing of Big Data streams, but rather the development of a Smart Logistics Operations System that continuously adapts to new information and subsystems. Such an application demands high availability, scalability, reliability, fault tolerance, and the ability to process streams of data in real-time, allowing learned knowledge to be applied instantaneously.

Two key decisions must be made for building a stream system that supports these same hard requirements in terms of implementation and solution development. The processing unit, implemented in the Cloud or at the Edge, must guarantee the lowest possible delay, while the use of Microbatch or True Streaming Processing should provide the best response time feasible, without sacrificing, at least at the design stage, reliability and fault tolerance. A third concern, occurring in the Data Ingestion layer of the Smart Logistics stack, is the heterogeneous nature of the data sources and their integration into a common Near Real-Time stream for the application. Finally, the development of a Smart Logistics Operations System relies on true Real-Time Analytics that naturally support decision-making, since optimal strategy changes are executed and orchestrated in the Streams.

KEYWORDS: Real-time big data stream processing; Supply Chain; Logistics Optimization; Inventory Management; Fleet Management; Traffic Data; Routing; Decision Making; Real-Time Feedback; Reinforcement Learning; feedback System; Emerging Trends; Supply Chain Automation.

I. INTRODUCTION

Logistics drives the economy. More than $10 trillion was spent in 2021 on the movement and storage of goods, constituting 12.5% of the given year's global GDP. Supply chain disruptions incurred over $650 billion in losses in 2020 alone. Logistics is becoming increasingly smart, harnessing IoT, AI, and other such technologies to fortify supply chains. AI uses technology to deliver solutions that facilitate understanding, forecasting, decision-making, and process automation—at unprecedented scales. Consequently, real-time data streaming is key to building systems adapted to the dynamic nature of logistics, enabling analysis as soon as relevant events happen.

The development of real-time data streaming at enterprise scale must address reliability, scalability, and fault tolerance, and establish a software architecture that closely mirrors the end-to-end process and covers all layers of Big Data processing in line with standard paradigms. Solutions to Big Data engineering challenges encountered in major logistics hubs (ports, airports, warehouses, and distribution centers) are proposed, focusing in particular on inventory optimization, fleet sizing, vehicle routing, and estimated time of arrival (ETA) calculations. The objective is to lay the foundations for intelligent real-time decision-making regarding logistics operations—from orchestration to reinforcement learning—driving automated optimization and greater efficiency. All proposals are backed by empirical data collected in real-time and represent a major step toward a Smart Logistics vision.

1.1. Background and Significance

The continuous exchange of always-on information and data in cyberspace is creating new opportunities to enhance real business processes in the physical world. Digital technologies enable organizations to operate in a smart and rapid-manner, helping to maintain competitiveness. Information Technology continues to revolutionize supply-chain processes and logistics operations with more real-time demand and supply data and higher levels of integration and

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

collaboration among trading partners. Yet day-to-day operations remain reactive when demand or supply changes. Information systems used for execution are not able to exploit these data pools for routinized decision support. Analytics and visualization tools produce useful info-needs for execution. Seamless integration of data, analytics and operations can optimize time-sensitive tasks and create positive efficiencies.

An ideal solution integrates operational data and analytics and then uses analytic insights to orchestrate dedicated operational actions. Data from internal, external and competitor sources augment knowledge and operational data. Optimization of inventory levels, stock locations and fleet capacity minimize execution costs. Inventory levels and locations of stock and fleet minimize execution costs. Logistics is a fundamental strategic aspect of every company addressing the global market by affecting revenue, costs and customer satisfaction. Each day throughout the globe millions of shipments are moving in real time but little or no decision is taken to improve the management of these shipments. A continuous-streaming view of shipment data enables short-term analysis and decision automation for defined processes.

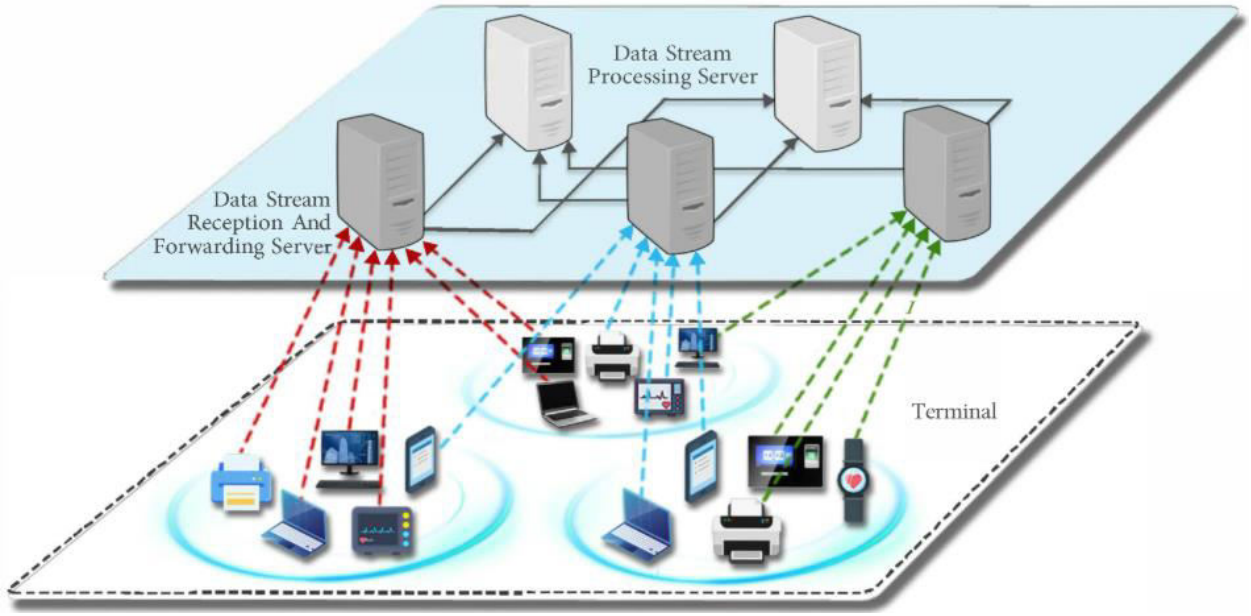


Fig 1: Real-Time Data Stream Transfer System in Edge Computing of Smart Logistics

1.2. Research design

Today's society is characterized by a demand for services and consumer goods being delivered in ever-shorter time periods. With the growing trend of companies putting primary effort into their core competencies, logistics service providers take over more and more logistics functions, including transporting, warehousing, and delivering goods on time. Some of the major logistics trends are the opening of new global markets through emerging economies, the battle for sustainability, and the rush to improve service quality while simultaneously lowering costs.

The optimization of the logistics and supply chain operations that support trade and commerce is a difficult task because these processes are faced with a multitude of conflicting objectives and structures, changing environmental conditions within a dynamic business setting, and complex interactions. Classical batch-mode data-processing techniques that have been used in the past often fall short of providing timely information to enable real-time synchronization of supply, distribution, transportation, and service processes by means of feedback control loops. Consequently, logistics and supply-chain managers cannot act in a truly anticipatory and proactive manner. Data-driven decision making and process automation have been considered possible only at the tactical or strategical decision-making level.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Equation 1: End-to-end latency decomposition (step-by-step)
Define end-to-end latency for an event (sensor reading, shipment status, GPS ping, etc.) as:

$$L_{\text{e2e}} = L_{\text{ingest}} + L_{\text{net}} + L_{\text{queue}} + L_{\text{proc}} + L_{\text{sink/act}}$$

Meaning of each term

- L_{ingest}: device→gateway serialization, protocol overhead, parsing
- L_{net}: network RTT / one-way propagation (edge usually smaller)

Real-Time Big Data Stream Engin…

- L_{queue}: waiting time due to bursts / peak loads
- L_{proc}: compute time for operators/models

$L_{\text{sink/act}}$: write to DB/queue + trigger action/orchestration

II. SCALABILITY, RELIABILITY, AND FAULT TOLERANCE

Real-time Big Data Engineering for Smart Logistics Optimization The architecture must provide strong levels of fault tolerance, reliability, and exactly-once processing, combined with high throughput and low latency. Big data stream processing systems can easily process millions of records per second, while low latency capabilities support strict service-level agreements (SLAs). With its distributed nature, a cluster can handle increasing loads whenever needed. However, this area still deserves more attention as providing fault-tolerance without compromising throughput can be hard. This holds especially for a logistics digital twin where these properties must remain valid through a peak load period like Black Friday involving thousands of ships and autonomous vehicles. Preventing any data from getting lost and ensuring that all virtual messages Flow e.g., for control-state change take effect are mandatory requirements. Thus, an adequate set of strategies for peak load handling and for preserving the solution's validity and correctness in these dense star and snowflake subgraphs must be foreseen.

Any distributed stream processing system in a logistics digital twin can benefit from a set of fault-handling support mechanisms, which provide reliability guarantees for the processing of virtual messages in the Flow. High reliability and fault tolerance levels can be achieved in-stream-processing systems by combining mechanisms for both state and data-forwarding redundancy and enabling exactly-once processing. In addition, such guarantees can be provided without incurring high overhead on throughput and latency when mixed with proper resource provisioning. The combination allows for balancing high reliability requirements with efficient resource use.

A robust stream-processing architecture for a logistics digital twin must reconcile three often competing objectives: strong fault tolerance, exactly-once processing semantics, and sustained high throughput with low latency. In peak scenarios such as Black Friday—where thousands of ships, trucks, and autonomous vehicles simultaneously generate telemetry and control-state updates—the system must scale horizontally across distributed clusters while preserving strict service-level agreements (SLAs). This requires mechanisms such as state replication, checkpointing, partition-aware load balancing, and backpressure control to prevent data loss or overload propagation. Exactly-once guarantees are particularly critical for control-state change messages, as duplicate or missed events could lead to inconsistencies between the physical and virtual environments. Furthermore, dense star and snowflake subgraph structures within the digital twin—representing hubs, ports, and interconnected fleets—intensify communication patterns and state dependencies, making correctness preservation under peak load more challenging. Therefore, the architecture must incorporate adaptive scaling, resilient state management, and deterministic processing strategies to ensure validity, consistency, and reliability without sacrificing performance, even under extreme operational stress.

A distributed stream processing system within a logistics digital twin must incorporate robust fault-handling mechanisms to ensure dependable and consistent operation. By integrating both state redundancy—through techniques such as checkpointing and replicated state management—and data-forwarding redundancy, the system can maintain continuity even in the presence of node or network failures. Enabling exactly-once processing semantics further guarantees that each virtual message in the flow is processed without duplication or loss, preserving data integrity across complex logistics operations. When these reliability mechanisms are combined with thoughtful resource provisioning, including dynamic scaling and workload-aware allocation, high levels of fault tolerance can be achieved without significantly compromising throughput or latency. This balanced approach ensures that stringent reliability requirements are met while maintaining efficient use of computational and network resources, which is essential for real-time, data-driven logistics environments.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

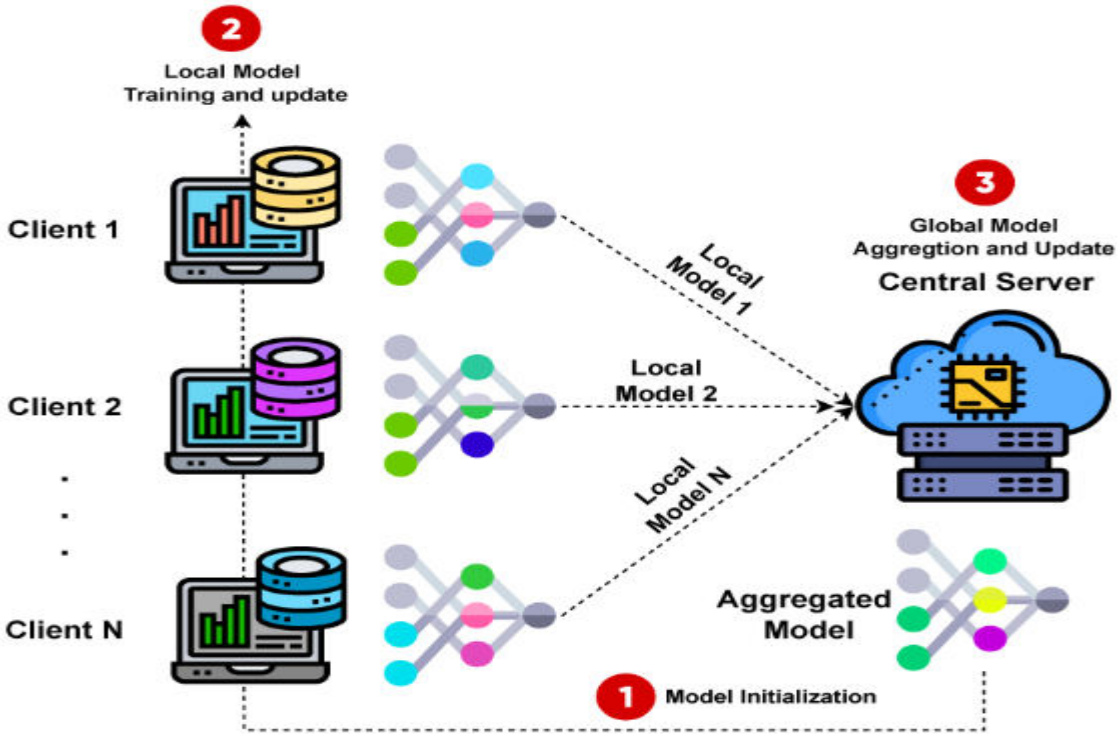


Fig 2: A Fault-Tolerant Scalable

2.1. Exactly-Once Processing and Fault Handling

Despite their large scale, big data streams are often less fault tolerant than conventional databases. Several mechanisms exist for exactly-once processing semantics, to ensure a consistent view of the input data and prevent the storage or streaming of duplicates. Replicated input sources can compensate for temporary outages, as can microbatching. Amazon Kinesis handles failures and maintains exactly-once delivery of analysed data through its data lake architecture. Other providers rely on a more indirect approach: they set checkpoints in the stream flow, and in case of failure rewind the entire stream sequence to a previous checkpoint. Exactly-once processing semantics can also be achieved simply through idempotent operations, though this method requires that the stream consumer be written specifically up to such standards.

As confirmed during the practical implementation of several internal projects, the cloud-based services offered by Microsoft Azure allow automatic input data protection, including these exactly-once processing mechanisms. However, in cases of planned maintenance, such as controller hot swapping, without redundant controllers, the Microsoft and Snowflake Services are unable to switch for operation temporarily with capacitive redundancy and bypass the working data quality measures. Such issues are actually common to any operation at this level, any supplier would require exit operational redundancy to hot-swap controllers for EtherCAT nodes. Adaptation is achieved by translating whichever service is being affected during planned maintenance from providing product supply flow analysis, while design of the redundancy is limited to the monitoring and alarming systems whose redundancy has been defined for ambient disturbances and NOT for project dependent design constraints.

2.2. Scalability Strategies for Peak Loads

Adequate scalability may be achieved by strategic data distribution and the use of data grids for replication but at increased complexity and cost. When velocity and volume peak, however, there is no substitute for sheer high quality resources. In such situations, temporary over-provisioning may be required. Elastic cloud infrastructures offer a means of accommodating this, but the public cloud is usually too slow and inelastic by the time peak loads are met. Energies are wasted routing sensitive data flows over long optic-fibre links to cloud data-centres. Thus operators of fast-traffic streams must organise ‘cloudlets’, meaning clusters of server resources just a fraction of RTT away from the edge-router that handles their high-throughput demand. When these caches can mirror at least the hot data sets in the public cloud it becomes viable for them to replicate flows to the main cloud on a longer timescale, observing TTL-based

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

policy-control directives. When trunk networks become congested and long-latency because of backup and cloud-scaling flows, local clouds may be switched-off altogether. Also, to support afterwards on-line-rolling upgrades and community-formation of shared-state paths for any networks of flows failing golden traffic engineering, deeper caching strategies can be activated dynamically because long-distance temporal locality is re-enabled. The attempt is to fall into a snare of wasting excessive compute and/or storage resources for this hot-tearing region of supply-chain architecture.

Furthermore, the live-show of continuously-controlled, multi-dimensional traffic-management can provide the operators of a scale-able edge-cloud data-stream enrichment service with detailed information about the behaviour of packets and about packet bursts. Dynamic queue-sizes provide on-the-fly fine-adjustments, alerting man or machine to pollution affecting performance when behaviour diverges significantly from common.

Equation 2: Microbatch latency (step-by-step)
Let Wbe the waiting time from event arrival until the batch closes.

- If arrival time is $U \sim \text{Uniform}(0, T)$, then waiting time is:

$$W = T - U$$

- Expected waiting time:

$$\mathbb{E}[W] = \mathbb{E}[T - U] = T - \mathbb{E}[U]$$

- For uniform U, $\mathbb{E}[U] = T/2$. Hence:

$$\mathbb{E}[W] = T - T/2 = T/2$$

So expected microbatch latency becomes:

$$\mathbb{E}[L_{\text{e2e}}] \approx \underbrace{T/2}_{\text{batch wait}} + \underbrace{L_{\text{fixed}}}_{\text{ingest+proc+sink}}$$

III. ARCHITECTURAL PATTERNS FOR STREAM PROCESSING

The Cloud Computing paradigm encompasses end-user devices, Cloud services, and Data Centers complemented by various layers: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). In addition to these standard layers, the edge computing paradigm introduces computational power in close proximity to end-user devices. The Edge-Cloud collaboration combines these two paradigms to operate streaming applications closer to data sources. Its horizontal scaling ability is especially relevant when many devices need to transmit data simultaneously during events like natural disasters, or when peaks of Internet-of-Things data traffic occur depending on the season or day of the week.

The majority of popular Stream Processing systems adopt a micro-batching approach (e.g. Apache Spark Stream and Apache Flink) while others (e.g. Apache Kafka, Apache Storm, Amazon Kinesis) enable true streaming. These two types of systems show different processing latencies and resource consumption profiles. As such, the appropriate selection between them relies on application-specific requirements. However, these cloud services cannot provide the very low latency required by certain applications, like detecting anomalies in video surveillance. As a result, a growing number of services are being deployed in network edges to minimize their latency and offer acceptable user experiences.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

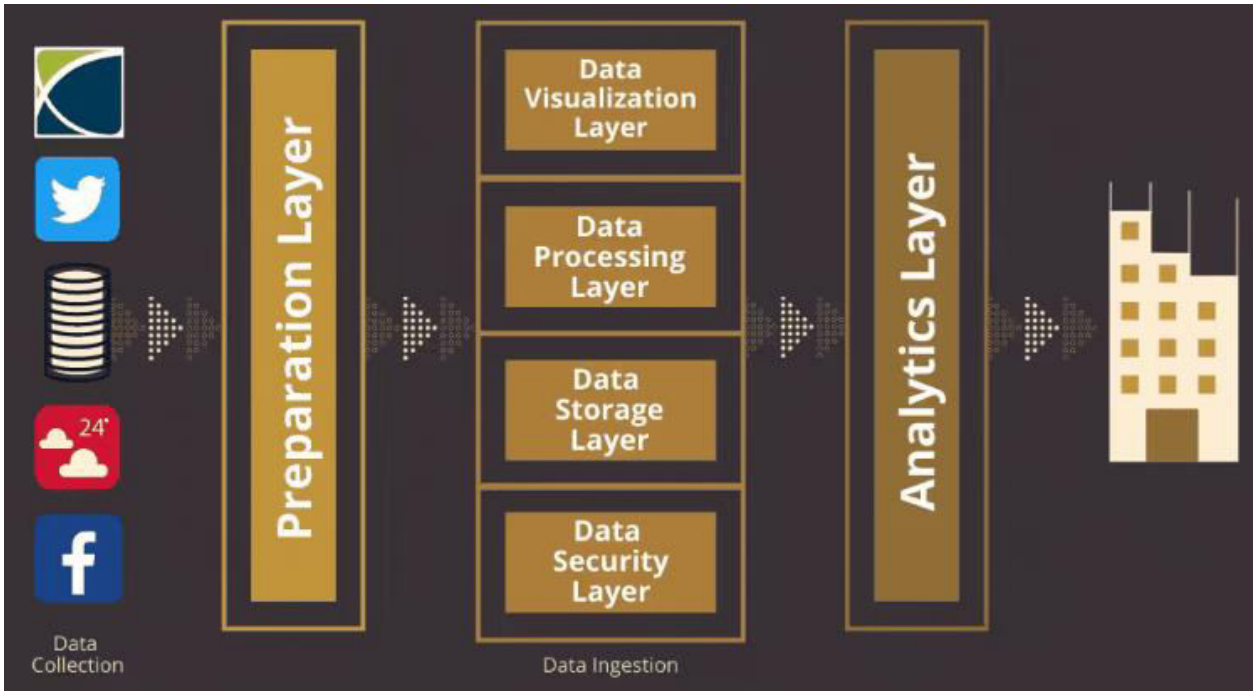


Fig 3: Data Streaming Architecture

3.1. Edge-Cloud Collaboration

A continuously evolving urban environment creates vast amounts of data from the activities of people and vehicles. Transport logistics is one area of significant data evolution. Road constructions advertisements and warning or information signs provide useful information for logistics operations. Delivery service companies seek to optimize last mile deliveries in metropolitan areas. Supply chains require timely deliveries and produce large amounts of data for decision support. Slowing down of processes pushes supply chains to realize more changes in real time. Stock replenishment in retail requires quick stock-taking operations in every facility.

Smart logistics optimizes these business objectives by using various technologies and by exploiting real-time data streams. Real-time data are produced by several processes such as GPS tracking weather information road conditions and social media. These changes encourage enterprises to achieve real-time monitoring and support of logistics data. However data generation is not sufficient. How to ingest and process data streams in real time and how to realize real-time decision support and automation are open research questions.

Cloud computing provides infrastructure and platform services that simplify and accelerate development of information systems. Computational resources storage space software and services are provided as services. However the cloud paradigm is not always the most suitable way to access services in every situation. Edge services are placed closer to users improve response time reduce network load and fulfill local service requirements. The increasing amount of data and connections require not only new computational power but also new services at the edge of the cloud. Increasing amounts of user data will be generated at the edge but only a limited part of the data will be sent to the cloud. Edge services process user data manage local network nodes and achieve quick service responses. Cloud-based data processing remains essential especially for historical analysis and long-term business monitoring.

3.2. Microbatch vs. True Streaming

Microbatch processing is often the best choice for processing data streams, yet it also alters the streaming model. The primary motivations for microbatching are simplified operational management (e.g. resource provisioning or autoscaling is conducted in larger time periods), support for interactive queries during processing, limited guarantees on fault tolerance and data loss (in comparison with truly streaming solutions), and cloud costs. Award-winning systems such as Apache Spark Streaming, Apache Flink, Samza and Storm can execute near real-time stream processing tasks at a very large scale. By adopting microbatch processing, it is possible to automatically direct data flows through the stream processing system and provision the required resources.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Microbatch processing imposes latency requirements on the duration of the batches, which in turn limits the real-time interactivity with data flows. Higher frequencies of data flow ingestion and processing result in shorter batch periods, and synchronous queries can return results based on the most recently completed batch or microstep. Data flows may need to be rerouted through different processing routes (for example, for analysis of anomalies) more frequently. However, the time required to complete an interactive query might still be higher than the desired latency for certain applications.

Equation 3: Inventory optimization: Newsvendor model (full derivation)

- Demand random variable $D$with CDF $F(d)$
- Choose order quantity Q
- Underage cost per unit (stockout penalty): c_u
- Overage cost per unit (leftover/holding): c_o
- Costs:
- Shortage amount: $(D-Q)^+ = \max(D-Q, 0)$
- Excess amount: $(Q-D)^+ = \max(Q-D, 0)$

Expected cost:

$$C(Q) = c_u\, \mathbb{E}[(D-Q)^+] + c_o\, \mathbb{E}[(Q-D)^+]$$

Use these facts:

1. $\left(D-Q\right)^+ = \int_Q^\infty \mathbf{1}\{D > x\}\, dx$

Taking expectation:

$$\mathbb{E}[(D-Q)^+] = \int_Q^\infty \mathbb{P}(D > x)\, dx = \int_Q^\infty (1 - F(x))\, dx$$

Differentiate w.r.t. Q:

$$\frac{d}{dQ}\mathbb{E}[(D-Q)^+] = \frac{d}{dQ}\int_Q^\infty (1-F(x))dx = -(1-F(Q))$$

2. Similarly:

$$\mathbb{E}[(Q-D)^+] = \int_{-\infty}^Q \mathbb{P}(D < x)\, dx = \int_{-\infty}^Q F(x)\, dx$$

Differentiate:

$$\frac{d}{dQ}\mathbb{E}[(Q-D)^+] = F(Q)$$

Differentiate cost:

$$C'(Q) = c_u \cdot (-(1-F(Q))) + c_o \cdot F(Q)$$
$$C'(Q) = -c_u + c_u F(Q) + c_o F(Q) = -c_u + (c_u + c_o)F(Q)$$

Set $C'(Q) = 0$:

$$-c_u + (c_u + c_o)F(Q^*) = 0$$
$$F(Q^*) = \frac{c_u}{c_u + c_o}$$

IV. DATA INGESTION AND INTEGRATION IN SUPPLY CHAINS

Data data ingestion in supply chain systems require attention in several aspects. Of those, heterogeneity is a primary challenge, as internal and external supply chain actors constantly generate massive amounts of data, integrating, storing, processing, and analyzing them in a coherent way can require substantial time. In contrast to the appliances actuation time (control/feedback loop), that can be considered reaction time (simple logic) for executing an operation such as opening/closing a valve, switching the light on/off or stimulating the user. This strongly depends on the completeness of input data coming from the data ingestion and data integration pipelines that could have some latency or may not contain data for triggering the operation.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Heterogeneous streams from actors can vary drastically in provenance, granularity, format, frequency, and nature (may belong to different modalities, paradigm: images with metadata, text, and numeric code). Furthermore, their quality may serially or concurrently (convergence versus divergence purposes) deviate from the predetermined—globally or locally (actor)" mitigating mechanisms should be implemented in the analytics (data-driven and model-driven) to allow the decision-making process to take into consideration also data quality. Data quality is mostly related to performance sustainment (monitoring function). Nevertheless, it is still important to create data quality logs for retro-analysis.

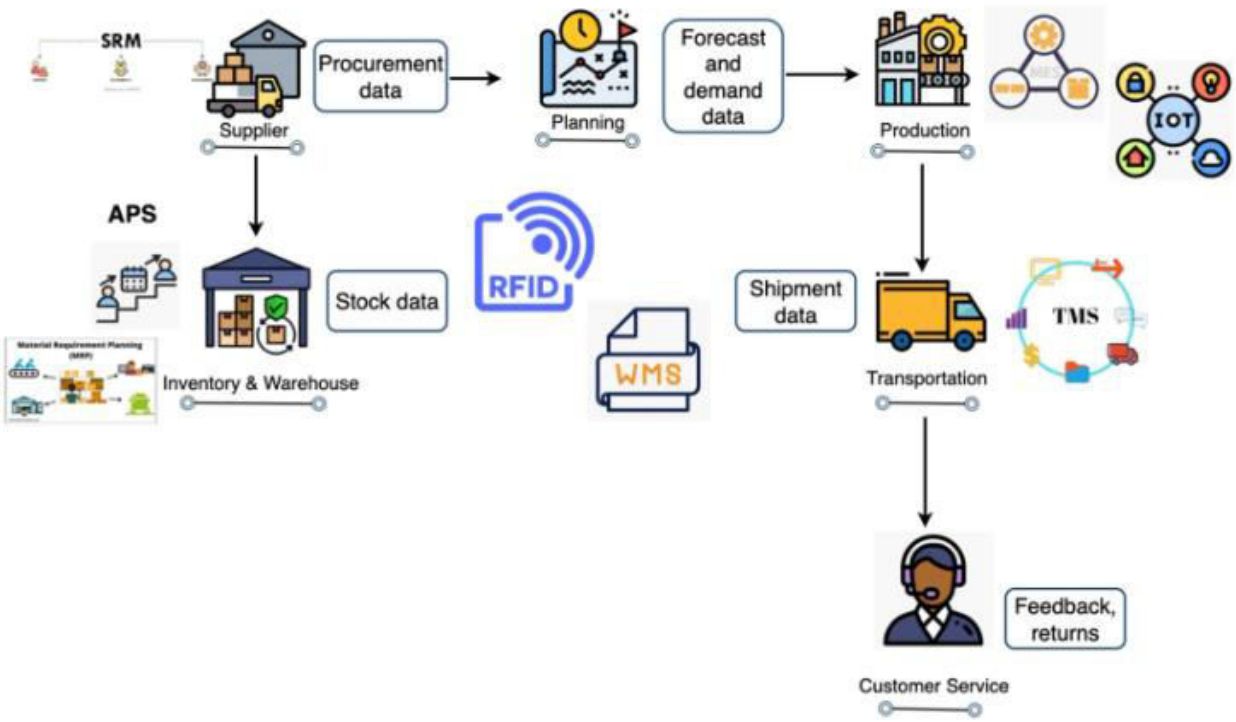


Fig 4: Data Management in Smart Manufacturing Supply Chains

4.1. Heterogeneous Data Sources

Data feeding the models comes from different data providers and needs to be integrated for analysis and prediction. For example, sensors on trucks provide predictive information about their normal operation, while external sources, such as meteorological data, allow better estimations of the ETA of trucks. The sensor data and external data can then be processed together to provide more realistic models.

An important challenge is data integration from different data providers and with distinct data formats. Cognitive video analytics, deep learning methods, and detection algorithms provide information for decision-making in a wide range of logistics problems. Automatic queue detection assists companies with touchless refueling procedures. Combined with external traffic data, road and weather conditions, truck-dependent cooling unit sensors and forecasts, ETAs can be predicted with greater accuracy. Merging heterogeneous sources and technology-agnostic data can improve prediction efficacy.

4.2. Data Quality and Provenance

Ensuring the quality of a smart logistics data pipeline is essential for all preparatory, operational, and maintenance phases. Outliers and noise introduced in the early stages may have an adverse impact on all types of analytics. Data quality, which is often not recognized as a problem for analytics on stream data, is therefore of utmost importance. Even for real-time analytics that are executed on data streams while they are continuously coming to the system, QA has to be implemented, as poor data quality might trigger a series of operational activities based on misleading analytics results. On one side, the user must take care of removing untrustworthy instances from the pipeline as soon as possible, while on the other side, a good-level data quality (e.g., correctness and consistency) should definitely improve the accuracy of analytics and, consequently, of all types of operational decisions that rely on these analytics.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Another critical issue is that of data provenance and trust in analytics results. Data provenance can be defined as the complete history of a data item, including the sources from which it is drawn and all processing operations that have been applied to it. Maintaining such information for all data in the pipeline is essential, especially when the results of key analytics are disseminated to decision makers in real time. Using low-quality or untrustworthy data for operating the system may cause severe problems. For this reason, it is also advisable to keep track of which instances of the data have been used for computing which results, to which decisions they were related, and, possibly, to store assessments about the trustworthiness of the various data sources.

Equation 4: Fleet sizing + delivery tradeoffs (MIP form)

- $x \in \mathbb{Z}_{\geq 0}$: number of owned vehicles (fleet size)
- $y_t \geq 0$: outsourced capacity purchased at time t(TNC/3PL)
- $I_t \geq 0$: inventory level
- $s_t \geq 0$: unmet demand (lost sales/backorder)
- $u_t \geq 0$: delivered units

Inventory balance:

$$I_{t+1} = I_t + \text{replen}_t - u_t$$

Demand satisfaction:

$$u_t + s_t = D_t$$

Capacity:

$$u_t \leq \underbrace{\alpha x}_{\text{owned capacity}} + y_t$$

Minimize expected total cost:

$$\min \sum_t (c^{\text{own}} x + c_t^{\text{tnc}} y_t + h I_t + p s_t)$$

V. REAL-TIME ANALYTICS FOR LOGISTICS OPTIMIZATION

Enabling new operational smartness and competitive advantages in logistics operations requires context-aware, operationally relevant, and real-time analytics. Such analytics can be identified based on actual data streams, predefined operational decisions or external factors that differ from the normal operating conditions and thus require special actions in near or real time.

Real-time analytics should enable a better decision-making process during routing by integrating support for inventory and fleet optimization, ETA estimation, identification of hotspots for dynamic re-routing, and prevention of disruptions based on learning from past operations. Reinforcement learning is particularly attractive since it allows feedback loops based on actual executed actions and their evaluation by the users to seamlessly improve operational policies over time.

5.1. Inventory and Fleet Optimization

Retail inventory and fleet are usually optimized periodically, based on stochastic demand forecasts generated by statistical methods. Large retailers use single- or multiple-location versions of the classical newsvendor model, combine it with stochastic demand forecasts, and employ dynamic programming to sequentially solve it for short time horizons. Many retailers offer same-day delivery using a mixture of owned and outsourced (3PL) fleets. Fleet optimization needs to take inventory, delivery times, and Transportation Network Companies (TNCs) prices into account. Inventory placement, fleet size, and delivery times (low versus high) are interrelated trade-off decisions. Cargo risk and delivery times can be integrated into a mixed-integer model to reduce demand loss by determining the Tweel curve for inventory risk.

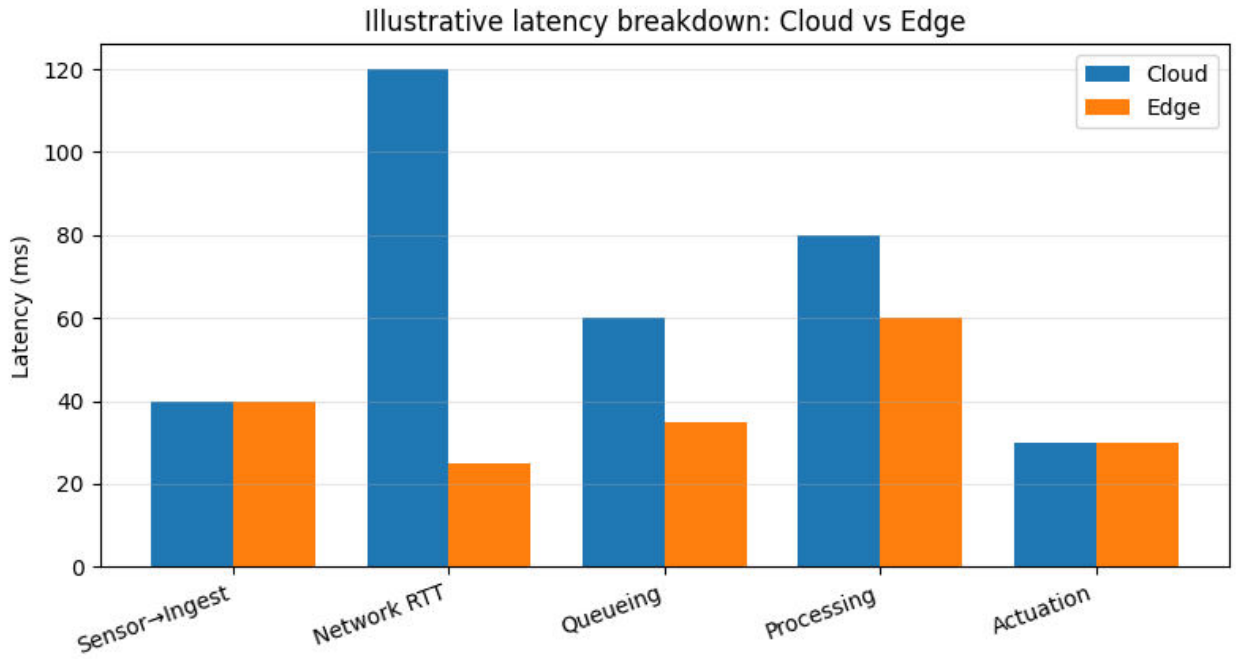
Web-based companies such as Amazon and Alibaba have established the B2C model as a product delivery benchmark. To remain competitive, other retailers must offer the same-service level. Their management relies on a mix of "Just In Case," "Just In Time," and "Fast Fashion" strategies. The combined use of external logistics service providers (LSP) networks, TNCs, immigrant labor, and metropolitan proximity allow to meet customers' expectations for low delivery fees on low-cost products.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021



5.2. Dynamic Routing and ETA Estimation

Dynamic vehicle routing problems (DVRP) entail planning the fleet routes to deliver goods in a cost-effective manner. Logistic managers must consider various constraints—limited vehicle capacities, time windows for pick-ups and deliveries, service durations, and connectivity in terms of the graph paving the roads. In addition, preventive strategies accommodate delays by minimizing the probability of a missed delivery time window, while reactive strategies respond to unexpected delays in transit time with secondary optimization routines. Also, deterministic traffic conditions can be relaxed as the corresponding travel-times become uncertain or undergo abrupt change, leading to a stochastic DVRP or a time-dependent DVRP.

The ETA of each shipment is a critical piece of knowledge in smart logistics. NoSQL data streams with multiple perturbations are processed for real-time ETA estimates of each shipment. The learned ETA models are maintained in parallel to test any ETA predictor during training. ETAs are the expected future arrival-time (TFAT) at any depot; TFAT for non-depot locations at the anticipated time of passing through the nearest depot. These are critical for proactive disruption management, enabling collaboration with the e2e customer prior to missing delivery deadlines.

Equation 5: Flow constraints (step-by-step)

Each customer visited exactly once:

$$\sum_{k}\sum_{i} x_{ij}^{k} = 1 \forall j \in \text{customers}$$

Vehicle flow conservation:

$$\sum_{i} x_{ij}^{k} = \sum_{m} x_{jm}^{k} \, \forall j, \forall k$$

Capacity:

$$\sum_{j} d_j \left(\sum_{i} x_{ij}^{k} \right) \leq Q_k$$

Time windows:

Let a_j arrival time at node j, service time s_j, window $[e_j, l_j]$:

$$e_j \leq a_j \leq l_j$$

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

$$a_j \geq a_i + s_i + t_{ij}(a_i) \text{if } x_{ij} = 1$$

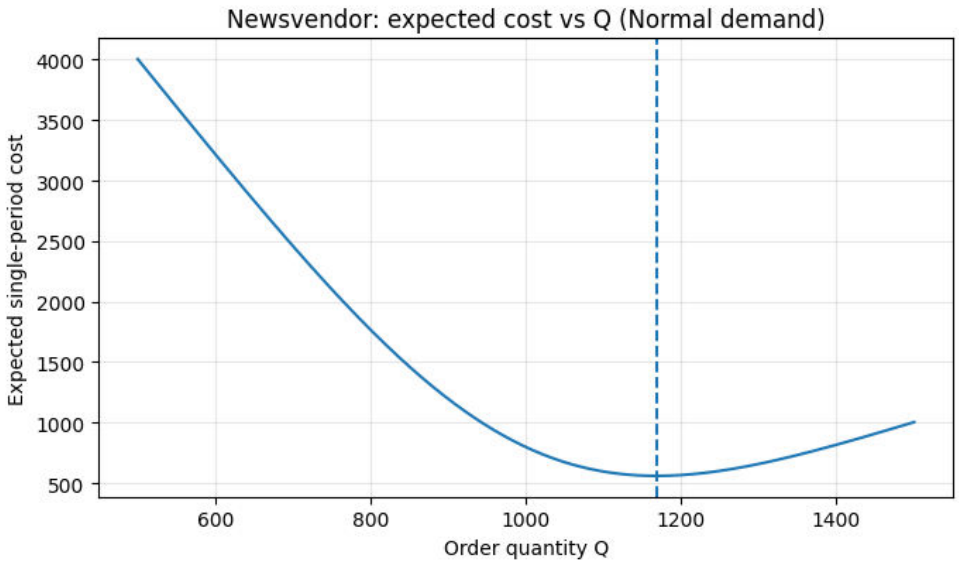
VI. DECISION MAKING AND AUTOMATION IN REAL TIME

To provide value to decision-making in logistics, real-time analytic- and data-processing systems must support automation by quickly orchestrating operational actions. Oftentimes, Stream Processing systems use Virtual Operators to represent analytic modules with action effects; they actually do not process stream data — their only task is to select actions among the available ones based on threshold-parameter queries. For example, if the current network traffic exceeds a threshold, the Shaping Traffic Control operator generates a set of actions to alleviate congestion by means of Traffic Shaping and/or Traffic Rerouting. Data-driven automation is a currently hot trend in stream processing. One possible solution is to use Complex Event Processing (CEP) capabilities and event patterns to trigger selected actions. However, deploying a control strategy by means of actions is not enough; closed-loop control is important for automatically tuning the control strategy based on feedback from the environment. Frequent high network jitter inducing shaping-action effect is one aspect of a timely-control strategy, whereas rodent congestion level is another aspect of a closed-loop solution.

As a consequence, providing an automatic mechanism to tune stream rules by using a Reinforcement Learning (RL) algorithm and a feedback-loop concept is another relevant research topic. RL is a feedback-learning mechanism in which the agent selects actions to be performed in the environment in order to maximize the expected long-term reward signal. In the stream-processing context, a Reinforcement Learning Control (RLC) module is responsible for controlling the internal state of a Stream Processing (SP) application in the course of real-time operation, adapting action thresholds based on the received feedback. In this way, the thresholds that fit well with the current stream-dynamics conditions can be selected in real time.

6.1. Orchestration of Operational Actions Operational actions, whether for logistics and supply chain management, telecommunications, financial trading, or any other area where decisions need to be taken in real time, have a long time horizon and span several interacting parts of the system. In the case of logistics optimization, available data streams support the real-time exploration of optimized inventory-transportation plans directly linked to inventory replenishment plus routing actions that switch vehicles and modes at stops, such as pipelines and warehouse transshipment hubs served by freight vehicles or drones.

This observation suggests the utility of an orchestrator that will trigger such long actions in their own time (detection of flows of bulk goods to minimize average shipping cost in supply chains, sensors for the warehouse replenishing flows, setup of transport vehicles, planes, ships for aggregated goods flow bursts, …). Orchestration is a reactive use of streams with the orchestrator describing the dependency relations with delays of the actions rather than of the stream, with orders defining the input conditions to take the actions, dedicated functions implementing plausibility checks, and action switching interlocked with successful feedback of the action execution.



International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

6.2. Feedback Loops and Reinforcement Learning in Streams

As already mentioned, traditional BI processes neglect real-time situations, hindering feedback loops. However, the supply chain is a complex system that is not directly predictable, and users aim at optimal, not arbitrary, business decisions. Automation is desirable yet risky; any decision can generate stochastically unfit results. Both areas require data refreshment without traditional batch-processed history or control feedback. Additionally, it is recommended to insert a routing layer in front of the stream data mart to orchestrate several operational actions. That layer takes inputs from the physical flow of goods and the virtual flow of data and supports strategic supply chain actions. Based on data-driven predefined policies, the decision engine makes automated operational actions for transportation, stocks, services, procurement, and distribution. Human intervention is an exception for abnormal occurrences or for executing actions that a machine cannot determine due to the missing components in its learned model. Reinforcement learning techniques in streaming environments can also be applied for routing prediction, improving real-time processes when transporting data and allowing for closed-loop behavior.

The objective of decision automation is to assist supply chain users. Users can be managers that need decision support, intelligent agents either operating in a game or controlling a simulation, or any other entity. They generally want to apply an a priori built knowledge base to optimize the outcome of actions. Such knowledge represents specific cases. Users aim at optimal processes and can afford to incur costs and have low-priority time constraints. Therefore, their opportunities can be stochastically unfit for real-time processes. Real-time BI methods that ignore hot data relation dependencies and only cover the necessity of behavior change to maintain a sampling environment have been proposed. Real-time reinforcement learning for newsvendor problems with cost-sensitive classification loss has been conceived, as newsvendor is a central role in supply chains. The processes of users not only inspect the state of the systems.

VII. CONCLUSION

Looking to the future, streaming big data engineering will provide key services to emerging systems for logistics supply chains. Logistics networks are becoming digital twins forming during planning, execution and optimization cycles. The Big Data paradigm has recently expanded to include many systems dealing with large—often unbounded and continuous—streams of data: Sensor Networks, TELERIGs, Spatio-Temporal Data Management. Scalability problems, delivery time-critical requirements and the need for on-line, Ai-driven internal/external feedback—as Computer Systems become more Automating Decisions—pose new challenges to existing supporting infrastructures, System Interaction Operations and Data Stream Engineering, in particular.

Driving system operation—achieved via various levels of automation or orchestration—requires the execution of operational actions: activating or migrating a resource, reassigning a tour or flight, modifying an order, etc. Many of these actions can be automated or aided by machine learning (ML) algorithms. Automating prediction and classification is key, yet it is usually not sufficient; automating the choice of decisions may be more complex, as answers are not always stored or easily derived from past decisions. Reinforcement Learning (RL) has emerged as a powerful paradigm to resolve such issues. In a streaming-acquisition environment, a Q-function answers: “Which is the best action to take in order to maximize the long-term payoff?” Its main novelty lies in the stepwise notation: the function is computed as actions are performed, acquiring knowledge in loops of trial-and-error, thus integrating learning directly into processing.

Aspect	Microbatch	True streaming
Latency	Bounded below by batch interval (≈T/2 avg wait)	Record-level; bounded by processing + network
Fault tolerance	Often simpler via batch retries/checkpoints	Requires continuous state + checkpointing
Exactly-once	Easier with microbatches and idempotent sinks	Achievable with checkpoints + transactional sinks

Table: Microbatch vs True streaming comparison

7.1. Emerging Trends

In the automotive sector, the widespread adoption of electric vehicles (EVs) is generating huge amounts of high-quality data from numerous sources. This data can be used for traffic prediction and real-time navigation to avoid congested areas and help other EVs reach their destinations. Moreover, for fleet operators, the optimization of charging stations and the real-time prediction of charging behaviors will improve the management of electricity and reduce operation costs. Logistical optimization of the supply chain and a better understanding of fleet management in maritime and aviation transportation are also relevant topics. There is certainly an ongoing efforts to introduce a real-time capability

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

to operations, also referred as "continuous operations". In the logistics domain, being able to react in real time to changes and incidents in the supply chain is a prerequisite to the realization of smart logistics. The gradual introduction of data science, machine learning, and artificial intelligence is indeed facilitating real-time actions.

Yet the requirements for systems and software to support real-time logging and smart logistics go beyond those for traditional stream processing. Supporting business operations in real time requires not only the capability to execute and react in real time, but also the continuous orchestration of all operational actions, together with feedback loops that learn from operations and improve proposals and predictions over time. Continuous orumble learning from data streams has the potential to introduce a new level of automation in decision making.

REFERENCES

[1] Al-Issa, Y., Ottom, M. A., & Tamrawi, A. (2022). Security challenges and solutions in eHealth cloud computing. Journal of Healthcare Engineering, 2022.
[2] Avinash Reddy Segireddy. (2022). Terraform and Ansible in Building Resilient Cloud-Native Payment Architectures. International Journal of Intelligent Systems and Applications in Engineering, 10(3s), 444–455. Retrieved from https://www.ijisae.org/index.php/IJISAE/article/view/7905
[3] Aldboush, H. H., & Ferdous, M. Building trust in fintech: An analysis of ethical and privacy considerations in the intersection of big data, AI, and customer trust. International Journal of Financial Studies, 11(3), 90. https://doi.org/10.3390/ijfs11030090
[4] Kothapalli Sondinti, L. R., & Syed, S. (2022). The Impact of Instant Credit Card Issuance and Personalized Financial Solutions on Enhancing Customer Experience in the Digital Banking Era. Universal Journal of Finance and Economics, 1(1), 1223. Retrieved from https://www.scipublications.com/journal/index.php/ujfe/article/view/1223
[5] Davuluri, P. N. (2020). Improving Data Quality and Lineage in Regulated Financial Data Platforms. Finance and Economics, 1(1), 1-14.
[6] Rongali, S. K. (2020). Predictive Modeling and Machine Learning Frameworks for Early Disease Detection in Healthcare Data Systems. Current Research in Public Health, 1(1), 1-15.
[7] Armbrust, M., Das, T., Davidson, A., Ghodsi, A., Or, A., Rosen, J., Stoica, I., Wendell, P., Xin, R., & Zaharia, M. (2021). Delta Lake: High-performance ACID table storage over cloud object stores. Proceedings of the VLDB Endowment, 13(12), 3411–3424.
[8] Gottimukkala, V. R. R. (2020). Energy-Efficient Design Patterns for Large-Scale Banking Applications Deployed on AWS Cloud. power, 9(12).
[9] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50–58.
[10] Chava, K., Chakilam, C., & Recharla, M. (2021). Machine Learning Models for Early Disease Detection: A Big Data Approach to Personalized Healthcare. International Journal of Engineering and Computer Science, 10(12), 25709–25730. https://doi.org/10.18535/ijecs.v10i12.4678
[11] Babcock, J., Chaudhuri, S., & Das, G. (2004). Dynamic sample selection for approximate query processing. Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, 539–550.
[12] Vadisetty, R., Polamarasetti, A., Guntupalli, R., Raghunath, V., Jyothi, V. K., & Kudithipudi, K. (2021). Privacy-Preserving Gen AI in Multi-Tenant Cloud Environments. Sateesh kumar and Raghunath, Vedaprada and Jyothi, Vinaya Kumar and Kudithipudi, Karthik, Privacy-Preserving Gen AI in Multi-Tenant Cloud Environments (January 20, 2021).
[13] Bifet, A., & Gavaldà, R. (2007). Learning from time-changing data with adaptive windowing. Proceedings of the 2007 SIAM International Conference on Data Mining, 443–448.
[14] Muthusamy, S., Kannan, S., Lee, M., Sanjairaj, V., Lu, W. F., Fuh, J. Y., ... & Cao, T. (2021). Cover Image, Volume 118, Number 8, August 2021. Biotechnology and Bioengineering, 118(8), i-i.
[15] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.
[16] Aitha, A. R. (2022). Cloud Native ETL Pipelines for Real Time Claims Processing in Large Scale Insurers. Available at SSRN 5532601.
[17] Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. Mobile Networks and Applications, 19(2), 171–209.
[18] Dwaraka Nath Kummari. (2022). Fiscal Policy Simulation Using AI And Big Data: Improving Government Financial Planning. Kurdish Studies, 10(2), 934–945. https://doi.org/10.53555/ks.v10i2.3855
[19] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.
[20] Gadi, A. L. The Role of Digital Twins in Automotive R&D for Rapid Prototyping and System Integration.
[21] Das, T., Zhu, A., Li, S., Narayanamurthy, S., & Bhat, P. (2013). Distributed and fault-tolerant streaming computation in Spark. Proceedings of the ACM Symposium on Cloud Computing, 1–12.

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

[22] Siva Hemanth Kolla. (2022). Knowledge Retrieval Systems for Enterprise Service Environments. International Journal of Intelligent Systems and Applications in Engineering, 10(3s), 495–506. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/8037
[23] Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. Communications of the ACM, 51(1), 107–113.
[24] Paleti, S. (2022). Financial Innovation through AI and Data Engineering: Rethinking Risk and Compliance in the Banking Industry. Available at SSRN 5250726.
[25] DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., & Vogels, W. (2007). Dynamo: Amazon's highly available key-value store. Proceedings of the 21st ACM Symposium on Operating Systems Principles, 205–220.
[26] Sriram, H. K., ADUSUPALLI, B., & Malempati, M. (2021). Revolutionizing Risk Assessment and Financial Ecosystems with Smart Automation, Secure Digital Solutions, and Advanced Analytical Frameworks.
[27] Dwork, C. (2008). Differential privacy: A survey of results. Proceedings of the 5th International Conference on Theory and Applications of Models of Computation, 1–19.
[28] Keerthi Amistapuram , "Energy-Efficient System Design for High-Volume Insurance Applications in Cloud-Native Environments," International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE), DOI 10.17148/IJIREEICE.2020.81209
[29] Elmagarmid, A. K., Ipeirotis, P. G., & Verykios, V. S. (2007). Duplicate record detection: A survey. IEEE Transactions on Knowledge and Data Engineering, 19(1), 1–16.
[30] Dwaraka Nath Kummari,. (2022). Machine Learning Approaches to Real-Time Quality Control in Automotive Assembly Lines. Mathematical Statistician and Engineering Applications, 71(4), 16801–16820. Retrieved from https://philstat.org/index.php/MSEA/article/view/2972
[31] Fader, P. S., Hardie, B. G. S., & Lee, K. L. (2005). "Counting your customers" the easy way: An alternative to the Pareto/NBD model. Marketing Science, 24(2), 275–284.
[32] Inala, R. (2022). Engineering Data Products for Investment Analytics: The Role of Product Master Data and Scalable Big Data Solutions. International Journal of Scientific Research and Modern Technology, 155-171.
[33] Davuluri, P. N. (2020). Improving Data Quality and Lineage in Regulated Financial Data Platforms. Finance and Economics, 1(1), 1-14.
[34] Meda, R. Enabling Sustainable Manufacturing Through AI-Optimized Supply Chains.
[35] Ghemawat, S., Gobioff, H., & Leung, S. T. (2003). The Google file system. Proceedings of the 19th ACM Symposium on Operating Systems Principles, 29–43.

[36] Varri, D. B. S. (2022). A Framework for Cloud-Integrated Database Hardening in Hybrid AWS-Azure Environments: Security Posture Automation Through Wiz-Driven Insights. International Journal of Scientific Research and Modern Technology, 1(12), 216-226.
[37] Yandamuri, U. S. (2021). A Comparative Study of Traditional Reporting Systems versus Real-Time Analytics Dashboards in Enterprise Operations. Universal Journal of Business and Management, 1(1), 1–13. Retrieved from https://www.scipublications.com/journal/index.php/ujbm/article/view/1357
[38] Gottimukkala, V. R. R. (2022). Licensing Innovation in the Financial Messaging Ecosystem: Business Models and Global Compliance Impact. International Journal of Scientific Research and Modern Technology, 1(12), 177-186.
[39] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer.
[40] Vadisetty, R., Polamarasetti, A., Guntupalli, R., Raghunath, V., Jyothi, V. K., & Kudithipudi, K. (2022). AI-Driven Cybersecurity: Enhancing Cloud Security with Machine Learning and AI Agents. Sateesh kumar and Raghunath, Vedaprada and Jyothi, Vinaya Kumar and Kudithipudi, Karthik, AI-Driven Cybersecurity: Enhancing Cloud Security with Machine Learning and AI Agents (February 07, 2022).
[41] Hellerstein, J. M., Haas, P. J., & Wang, H. J. (1997). Online aggregation. Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, 171–182.
[42] Garapati, R. S. (2022). Web-Centric Cloud Framework for Real-Time Monitoring and Risk Prediction in Clinical Trials Using Machine Learning. Current Research in Public Health, 2, 1346.
[43] Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. Proceedings of the 2008 IEEE International Conference on Data Mining, 263–272.
[44] Amistapuram, K. (2022). Fraud Detection and Risk Modeling in Insurance: Early Adoption of Machine Learning in Claims Processing. Available at SSRN 5741982.
[45] Davuluri, P. S. L. N. (2021). Event-Driven Compliance Systems: Modernizing Financial Crime Detection Without Machine Intelligence. Journal of International Crisis and Risk Communication Research , 339–354. https://doi.org/10.63278/jicrcr.vi.3636

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

[46] Meda, R. (2022). Integrating Edge AI in Smart Factories: A Case Study from the Paint Manufacturing Industry. International Journal of Science and Research (IJSR), 1473-1489.
[47] Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., & Shahabi, C. (2014). Big data and its technical challenges. Communications of the ACM, 57(7), 86–94.
[48] Segireddy, A. R. (2020). Cloud Migration Strategies for High-Volume Financial Messaging Systems.
[49] Khatri, V., & Brown, C. V. (2010). Designing data governance. Communications of the ACM, 53(1), 148–152.
[50] Amistapuram, K. (2021). Digital Transformation in Insurance: Migrating Enterprise Policy Systems to .NET Core. Universal Journal of Computer Sciences and Communications, 1(1), 1–17.
[51] Goutham Kumar Sheelam, "Semiconductor Innovation for Edge AI: Enabling Ultra-Low Latency in Next-Gen Wireless Networks," International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI: 10.17148/IJARCCE.2022.111258
[52] Nagabhyru, K. C. (2022). Bridging Traditional ETL Pipelines with AI Enhanced Data Workflows: Foundations of Intelligent Automation in Data Engineering. Available at SSRN 5505199.
[53] Lahiri, M., & Venkatasubramanian, S. (2013). Robust record linkage. Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, 101–112.
[54] Avinash Reddy Aitha. (2022). Deep Neural Networks for Property Risk Prediction Leveraging Aerial and Satellite Imaging. International Journal of Communication Networks and Information Security (IJCNIS), 14(3), 1308–1318. Retrieved from https://www.ijcnis.org/index.php/ijcnis/article/view/8609
[55] Leskovec, J., Rajaraman, A., & Ullman, J. D. (2014). Mining of massive datasets (2nd ed.). Cambridge University Press.
[56] Rongali, S. K. (2022). AI-Driven Automation in Healthcare Claims and EHR Processing Using MuleSoft and Machine Learning Pipelines. Available at SSRN 5763022.
[57] Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7(1), 76–80.
[58] Meda, R. (2021). Digital Infrastructure for Predictive Inventory Management in Retail Using Machine Learning. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), DOI, 10.
[59] Lin, J., Kolcz, A., & Szymanski, B. K. (2012). Large-scale machine learning at Twitter. Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, 793–804.
[60] Sheelam, G. K. Power-Efficient Semiconductors for AI at the Edge: Enabling Scalable Intelligence in Wireless Systems. International Journal of Innovative Research in Electrical, Elec-tronics, Instrumentation and Control Engineering (IJIREEICE), DOI, 10.

[61] Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., & Byers, A. H. (2011). Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute.
[62] Vadisetty, R., Polamarasetti, A., Guntupalli, R., Rongali, S. K., Raghunath, V., Jyothi, V. K., & Kudithipudi, K. (2021). Legal and Ethical Considerations for Hosting GenAI on the Cloud. International Journal of AI, BigData, Computational and Management Studies, 2(2), 28-34.
[63] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. Proceedings of the International Conference on Learning Representations, 1–12.
[64] Ramesh Inala. (2022). Cross-Domain MDM Integration Using AI-Driven Data Governance: A Case Study In Financial Technology Architecture. Migration Letters, 19(2), 280–304. Retrieved from https://migrationletters.com/index.php/ml/article/view/11982
[65] Montoya, D. Y., Neto, A. M., & da Silva, A. S. (2016). A survey of entity resolution in big data. Journal of Big Data, 3(1), 1–22.
[66] Aitha, A. R. (2021). Optimizing Data Warehousing for Large Scale Policy Management Using Advanced ETL Frameworks.
[67] Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, 1–7.
[68] Varri, D. B. S. (2022). AI-Driven Risk Assessment and Compliance Automation in Multi-Cloud Environments. Available at SSRN 5774924.
[69] Zaharia, M., Das, T., Li, H., Shenker, S., & Stoica, I. (2012). Discretized streams: Fault-tolerant streaming computation at scale. Proceedings of the 24th ACM Symposium on Operating Systems Principles, 423–438.
[70] Segireddy, A. R. (2021). Containerization and Microservices in Payment Systems: A Study of Kubernetes and Docker in Financial Applications. Universal Journal of Business and Management, 1(1), 1–17.
[71] Zhai, C., & Massung, S. (2016). Text data management and analysis: A practical introduction to information retrieval and text mining. ACM & Morgan Claypool.
[72] Kolla, S. H. (2021). Rule-Based Automation for IT Service Management Workflows. Online Journal of

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 5, Issue 6, November- December 2022 ||

DOI: 10.15680/IJCTECE.2022.0506021

Engineering Sciences, 1(1), 1–14. Retrieved from https://www.scipublications.com/journal/index.php/ojes/article/view/1360
[73] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, 135–146.
[74] Uday Surendra Yandamuri. (2022). Cloud-Based Data Integration Architectures for Scalable Enterprise Analytics. International Journal of Intelligent Systems and Applications in Engineering, 10(3s), 472–483. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/8005
[75] Goutham Kumar Sheelam. (2022). Reconfigurable Semiconductor Architectures For AI-Enhanced Wireless Communication Networks. Kurdish Studies, 10(2), 1027–1040. https://doi.org/10.53555/ks.v10i2.3867
[76] Batarseh, F. A., & Yang, R. (2019). Federal data science: Transforming government and society. Academic Press.
[77] Gottimukkala, V. R. R. (2021). Digital Signal Processing Challenges in Financial Messaging Systems: Case Studies in High-Volume SWIFT Flows.
[78] Bhasin, H., & Bhatia, P. (2020). Clickstream data mining for web analytics and customer behavior modeling: A review. ACM Computing Surveys, 53(6), 1–34.
[79] Rongali, S. K. (2021). Cloud-Native API-Led Integration Using MuleSoft and .NET for Scalable Healthcare Interoperability. Available at SSRN 5814563.
[80] Davuluri, P. N. (2020). Event-Driven Architectures for Real-Time Regulatory Monitoring in Global Banking.
[81] Abedjan, Z., Golab, L., & Naumann, F. (2016). Profiling relational data: A survey. The VLDB Journal, 24(4), 557–581.
[82] Garapati, R. S. (2022). AI-Augmented Virtual Health Assistant: A Web-Based Solution for Personalized Medication Management and Patient Engagement. Available at SSRN 5639650.
[83] Dwaraka Nath Kummari. (2022). AI-Driven Audit Frameworks For Enhancing Compliance In Modern Manufacturing Systems. Migration Letters, 19(S8), 2150–2177. Retrieved from https://migrationletters.com/index.php/ml/article/view/11912
[84] Davuluri, P. N. Event-Driven Compliance Systems: Modernizing Financial Crime Detection Without Machine Intelligence.
[85] Baesens, B., Van Vlasselaer, V., & Verbeke, W. (2021). Fraud analytics using descriptive, predictive, and social network techniques: A guide to data science for fraud detection (2nd ed.). Wiley.
[86] Varri, D. B. S. (2021). Cloud-Native Security Architecture for Hybrid Healthcare Infrastructure. Available at SSRN 5785982.
[87] Barocas, S., Hardt, M., & Narayanan, A. (2019). Fairness and machine learning. fairmlbook.org (Book manuscript).
[89] Sriram, H. K. (2022). Advancements in Credit Score Analytics using Deep Learning and Predictive Modeling Techniques. Available at SSRN 5255128.
[90] Ajayi, A. A., Igba, E., Soyele, A. D., & Enyejo, J. O. Enhancing digital identity and financial security in decentralized finance (DeFi) through zero-knowledge and blockchain solutions for regulatory compliance and privacy. Journal of Digital Security and Forensics, 2(1).
[91] Yandamuri, U. S. (2022). Big Data Pipelines for Cross-Domain Decision Support: A Cloud-Centric Approach. International Journal of Scientific Research and Modern Technology, 1(12), 227–237. https://doi.org/10.38124/ijsrmt.v1i12.1111
[92] Inala, R. Advancing Group Insurance Solutions Through Ai-Enhanced Technology Architectures And Big Data Insights.
[93] Aljabre, A. (2019). Cloud computing security in healthcare. Journal of King Saud University – Computer and Information Sciences, 31(1), 10–18.
[94] Kolla, S. K. (2021). Architectural Frameworks for Large-Scale Electronic Health Record Data Platforms. Current Research in Public Health, 1(1), 1–19. Retrieved from https://www.scipublications.com/journal/index.php/crph/article/view/1372
[94] Blumenstock, J. E., & Kohli, N. Big data privacy in emerging market fintech and financial services: A research agenda. arXiv preprint arXiv:2310.04970.