ADVANCE'2026 25-28 March 2026,, SC, Brazil

Proceedings of the 13th International Workshop on ADVANCEs in ICT Infrastructures and Services

25-28 March 2026

Federal University of Santa Catarina

Florianópolis, Brazil

Table of contents

Foreword

It is a great pleasure to welcome you to ADVANCE 2026: the 13th International Workshop on Advances in ICT Infrastructures and Services, held this year in Florianópolis, Brazil. ADVANCE continues to serve as a dynamic forum where researchers, practitioners, and engineers from academia and industry come together to share insights, exchange ideas, and discuss emerging trends and future directions in ICT infrastructures, networking, and distributed services.

Since its first edition in 2012 in Canoa Quebrada (Brazil), organized with the support of IFCE Aracati, the ADVANCE workshop has grown into an international event hosted across multiple continents. The second edition was held in Morro de São Paulo (Brazil), followed by Miami (USA) in 2013, Recife (Brazil) in 2014, and Évry Val d'Essonne (France) in 2015. In 2017, the workshop took place in Santiago (Chile), supported by UEVE/Paris-Saclay and the Universidad de Chile. Subsequent editions were held in Cape Verde, Cancún (Mexico), and, during the global pandemic, online with the support of the University of Zaragoza (Spain) and University College Cork (Ireland). The workshop returned to an in-person format for its 10th edition in Brazil, supported by UFC and IFCE-Fortaleza. More recently, ADVANCE was hosted in Hanoi (Vietnam) in 2024 and in Sophia Antipolis (France) in 2025, supported by Université Côte d'Azur and UNIFACS University. Each edition has been made possible thanks to the strong commitment of local academic partners and the active engagement of the research community.

The 2026 edition features six technical sessions, including 12 full papers and 8 short papers. The program also includes a keynote lecture by Prof. Jonice Oliveira (Federal University of Rio de Janeiro, Brazil), addressing crowd dynamics in large-scale events, urban challenges, and the role of social network analysis in promoting well-being among underserved populations. In addition, an invited talk by Prof. Mauro Antonio de Oliveira (Federal Institute of Education, Science and Technology of Ceará, Brazil) will explore *Artificial Intelligence Datacenters: Opportunities and Threats*, examining how the global race for AI infrastructure creates both significant development opportunities and new risks of technological dependency, particularly for countries in the Global South.

We would like to thank all the authors who submitted their work, as well as the participants joining us in Florianópolis. Our sincere appreciation goes to our distinguished speakers and to the members of the Technical Program Committee for their careful evaluation and selection of the contributions. We are especially grateful to our colleagues at the Federal University of Santa Catarina (UFSC) for their dedication and support in organizing this edition.

We hope that ADVANCE 2026 provides a stimulating, collaborative, and rewarding experience for all participants.

Prof. Paulo Nazareno Maia Sampaio and Prof.Edson Tavares De Camargo
ADVANCE 2026 TPC Co-chairs

An Evaluation of Device-to-Device Communication and URLLC Service in 5G Networks

Edson T. de Camargo
Federal Technology University of Paraná (UTFPR)
Toledo, PR, Brazil
Software/Hardware Integration Lab (LISHA) – Federal University of Santa Catarina (UFSC)
Florianópolis, SC, Brazil
edson@utfpr.edu.br

Antônio Augusto Fröhlich
Software/Hardware Integration Lab (LISHA) – Federal University of Santa Catarina (UFSC)
Florianópolis, SC, Brazil
guto@lisha.ufsc.br

Abstract

Device-to-device (D2D) communication allows devices to exchange messages directly over a cellular network without always requiring a base station as an intermediary. In 5G networks, D2D communication is particularly appealing for vehicular-to-everything (V2X) communication because it can meet the ultra-low latency and high reliability requirements of the URLLC (Ultra-Reliable Low Latency Communications) service. However, meeting URLLC requirements remains challenging even years after its definition and the deployment of 5G networks. This article investigates and evaluates D2D communication in the context of the URLLC service using the Simu5G simulator. The study presents the main 3GPP standard definitions for D2D and URLLC communication and evaluates the D2D and URLLC features implemented in the simulator.

CCS Concepts

• **Networks** → **Network simulations**.

Keywords

Sidelink, URLLC, PC5, 5G, device-to-device.

1 Introduction

Vehicle-to-Everything (V2X) communications enable a vehicle to interact with other vehicles and nearby elements, such as road infrastructure, signage, pedestrians, and cyclists, with the primary goal of making driving and travel safer, smarter, and more comfortable [12].

Device-to-device (D2D) communication refers to direct communication between two user equipment (UE) without routing through the base station (BS) or network core. D2D is a key feature of cellular networks, particularly 5G networks, also known as 5G New Radio (5G NR), as it allows communication between vehicles (Vehicle-to-Vehicle, or V2V) bypassing cellular infrastructure to provide faster and more reliable transmission of critical information. This direct connection is often called Sidelink (SL) and uses the interface known as PC5. The PC5 interface is designed to enable SL communications in scenarios both within and outside the UE coverage area using Proximity Services (ProSe) [8]. Much of the progress in SL has been due to the role of 3GPP in the context of 5G NR networks.

Among the features that make 5G NR networks attractive for V2X is their Ultra-Reliable and Low-Latency Communication (URLLC) service, which supports critical applications such as autonomous and connected vehicles, industrial automation, remote control, and virtual reality. The goal of URLLC is to reduce latency to less than one millisecond and achieve reliability greater than 99.99%. Although D2D and URLLC are different concepts, they can be combined to create more robust and efficient communication systems [23]. Because D2D communication does not involve the core network, there are fewer communication hops, resulting in lower latency and a reduced probability of packet loss.

However, meeting the demanding latency and reliability requirements for URLLC in D2D remains a challenge [16, 23, 20]. Maghsoudnia et al. [16] argue that it is unclear whether and how URLLC can be achieved, requiring a holistic, system-level perspective to address all inherent bottlenecks. Yan and Jerome [23] state that research on URLLC and Sidelink is still limited, especially for applications requiring minimal latency and very high reliability. By adjusting parameters such as numerology, modulation and coding scheme (MCS), and MAC layer scheduling, the authors claim to have achieved a configuration capable of meeting stringent URLLC requirements, demonstrating the potential of V2X Sidelink communication in the 5.9 GHz band. However, they note that challenges persist in achieving URLLC under certain conditions, such as busy channels, where system performance may be compromised. Therefore, developing solutions to increase reliability and reduce latency is essential to maintain high performance in demanding communication environments.

In this context, simulation tools that accurately model the new mechanisms and technologies in 5G NR are essential for researching and evaluating proposals and improvements that address the communication requirements of emerging services [15]. Some simulators, such as LENA-5G [14], focus on implementing lower-layer protocols, while Simu5G [17] is a system-level simulator notable for its autonomous 5G architecture, independence from LTE networks, and integration with vehicle simulators like Veins[1]. However, simulators often do not readily provide the necessary implementations to meet standards. Additionally, the available documentation is not always clear or detailed, which hinders new implementations and delays progress in related research and the development of new solutions.

This article investigates and evaluates Sidelink communication and the URLLC service using a scenario implemented with the Simu5G simulator, a leading tool for 5G network simulation. Simu5G is developed as a library for the OMNeT++ simulation framework. The simulator models the data plane of the Radio Access Network (RAN) and the network core. Its features include a 3GPP-compliant

[1] https://veins.car2x.org/

network protocol stack, physical layer transmission with realistic channel models, and support for 5G NR Release 16. The results show the end-to-end latency in a Sidelink communication scenario, highlighting the simulator's capabilities for D2D and URLLC communication.

The remainder of this article is organized as follows. Section 2 provides an overview of Sidelink, URLLC, and related work. Section 3 describes the evaluation scenario. Section 4 presents the results, and Section 5 offers the conclusion.

2 Sidelink, URLLC and Related Work

Sidelink refers to the standardized technology that enables direct communication between UEs without data passing through the network [2]. URLLC is one of the main services of 5G NR, primarily intended for critical applications [1]. Table 1 summarizes the evolution of URLLC and Sidelink in the 3GPP releases [12, 23]. Release 15 introduced 5G NR and established the fundamentals of URLLC, such as numerology and the ability to schedule packets with durations shorter than a full slot (mini-slots). Release 16 marks the introduction of Sidelink in 5G NR, which was previously defined only for 4G LTE [4]. Release 17 expands and refines URLLC for space networks and low-cost devices. For Sidelink, Release 17 consolidates the Proximity Service (ProSe), a set of functionalities that allow a UE, for example, to discover other devices enabled for direct communication in its vicinity. Release 18 incorporates machine learning into the URLLC service and enables Sidelink for unlicensed bands. The following subsections describe the main features of Sidelink and URLLC.

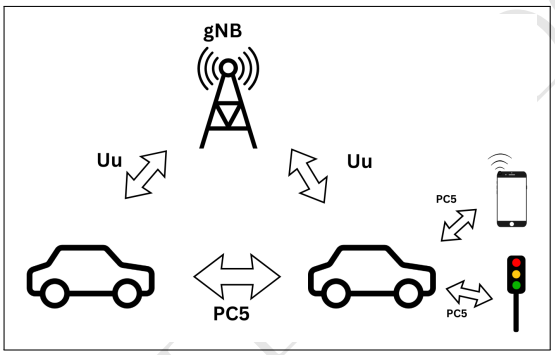

Figure 1: Sidelink Communication Architecture.

2.1 Sidelink V2X

Sidelink communications occur via the PC5 interface, while Vehicle-to-Network (V2N) communication, both uplink and downlink, uses the Uu interface, as shown in Figure 1. There are two communication modes: mode 1 and mode 2 [2]. These modes define the selection of subchannels via the PC5 interface and correspond to modes 3 and 4 of LTE V2X. However, Sidelink in LTE supports only broadcast communications, while 5G NR V2X supports broadcast, groupcast, and unicast sidelink communications.

Table 1: Evolution of URLLC and Sidelink.

Release	Main Objective	URLLC	V2X Sidelink (SL)
R15 (2018) TR21.915	The Foundation of 5G. Enhanced Mobile Broadband (eMBB) and NSA and V2X architecture phase 2.	**Basis for URLLC:** Introduction of flexible 5G NR radio frames (shorter slots, wider subcarrier spacing - SCS) that form the basis for low latency.	Focus remains on LTE-V2X (in Sidelink) for Basic Safety Messages, Sidelink in the ITS (Intelligent Transportation Systems) band, Broadcast only; uplink/downlink/sidelink
R16 (2020) TR21.916	5G Phase 2: Industrial 5G	Enhanced URLLC (eURLLC) - **Multi-Antenna and Multi-Point Design:** Increases reliability (spatial diversity). - **Transmit/Receive (Tx/Rx) Duplication:** Enables fast retransmission for extreme latency and reliability. - **Time Sensitive Networking (TSN)**: Integration with industrial networks for deterministic latency.	The Birth of the 5G NR V2X SL - **Introdução do NR V2X SL (PC5):** For the first time, Sidelink uses the 5G NR waveform. - **Support for New Applications:** Platoon, Advanced Driving, and Extended Sensors. - **Advanced Transmission Modes:** Supports Broadcast, Groupcast, and Unicast communication.
R17 (2022) TR21.917	5G Expansion and Refinement	URLLC for RedCap and NTN - **NR-Light 5G (Reduced Capability):** It optimizes 5G for low-cost industrial IoT devices that still require lightweight URLLC (e.g., surveillance cameras). - **Non-Terrestrial Networks (NTN)**: Extends URLLC for satellite communications.	NR V2X SL Expansion - **Improved Reliability and Coexistence:** Enhanced resource scheduling and re-selection mechanisms (Mode 2). - **Positioning**: Refinement of signals for precise location services (although SL Positioning is further improved in R18). - **UE relay:** UE acts as a relay - **Proximity Service (Prose)**: Enabling features such as direct discovery, group calls, and efficient off-network communication.
R18 (2024) TR21.918	5G-Advanced (AI e Convergência)	IA/ML for URLLC - **Application of Machine Learning (ML):** Using AI to optimize resource management, predict channel failures, and schedule to ensure reliability and latency. - **Duplex Improvements**: Optimization for latency-critical TDD applications.	Optimized and Convergent V2X Sidelink - **Integrated Communication and Sensing:** Sidelink is now used not only to communicate data, but also to sense and map the environment (e.g., vehicles "see" each other via radio signal). - **Sidelink in Unlicensed Bands (NR-U):** Extends V2X to unlicensed spectrum.

Mode 1 is similar to LTE V2X Mode 3. The base station, called gNB in 5G NR or eNB in 4G LTE, assigns and manages Sidelink radio resources for V2V communications in Mode 1 using the Uu interface. Therefore, UEs must be within the network coverage area to operate in Mode 1. Sidelink radio resources can be allocated by licensed carriers dedicated to Sidelink communications or by licensed carriers that share resources between Sidelink and Uu communications. Mode 1 uses dynamic scheduling, as in LTE V2X Mode 3, but replaces the semi-persistent scheduling in LTE V2X Mode 3 with a configured grant schedule [1] (presented in subsection 2.2). In Mode 2, UEs can autonomously select their Sidelink resources (one or more subchannels) from a resource pool. In this case, UEs can operate without network coverage. The resource pool can be preconfigured or configured by the gNB or eNB when the UE is within network coverage. Mode 2 can use a dynamic or semi-persistent scheduling scheme [12]. Based on modes 1 and 2, the following scenarios are possible:

(1) Two UEs are within the same gNB coverage area. In this case, the gNB manages the communication resources;
(2) Two UEs are in the coverage areas of different gNBs. Network coordination between the gNBs and the core network is usually necessary;

(3) Two UEs are communicating outside the coverage area. The UEs use preconfigured or self-managed resources;
(4) One UE is within the coverage area and the other is not. In this case, the UE inside the coverage area can act as a relay for the UE outside the coverage area.

Proximity-Based Services (ProSe) define the architectural services and functions that enable UEs to discover and communicate directly when they are near each other [5]. ProSe covers control plane aspects, including direct discovery – the procedures for a UE to find other nearby ProSe-enabled UEs; direct communication – the procedures for establishing and maintaining direct communication links; and security, authorization, and resource management – how the network controls access to the service and allocates resources. In the context of ProSe, mechanisms also exist to allow operators to charge for the Sidelink service.

Typical V2X applications involve message exchanges, such as those defined by the Intelligent Transport Systems (ITS) set of standards by the ETSI: Cooperative Awareness Messages (CAMs) and Decentralized Environmental Notification Messages (DENMs) [23]. CAMs provide continuous updates on a vehicle's position and movement, enhancing situational awareness for nearby vehicles. DENMs are triggered by potential hazards or accidents, alerting other drivers to changes in road conditions or emergencies. Together, these applications enable ongoing information exchange, helping vehicles maintain safety and navigate roads efficiently.

In general, applications align with the capabilities of standard 5G-NR communication, where requirements such as CAM transmission with delays typically under 100 ms and 90–99% reliability for packets of about 300 bytes at a rate of 10 Hz can be met. However, advanced applications present greater challenges, requiring latencies below 10 ms and reliability rates between 99.99% and 99.999%. These requirements are difficult to achieve with standard 5G-NR configurations because they are used in complex vehicular operations that demand enhanced control, which justifies the use of the URLLC service.

2.2 URLLC

URLLC is a continuously evolving 5G network service developed through various mechanisms and optimizations incorporated into the 3GPP specifications [1, 2, 3]. It does not involve activating a single interface, but rather configuring and integrating specialized features across the entire radio access network (RAN) and network core. For example, in the network core, network slicing allows multiple logical networks to operate on a shared physical infrastructure, creating logically separate "slices".

The main RAN mechanisms and configurations designed to meet URLLC requirements include numerology, mini-slots, and configured grants (grant-free or configured grants). According to Yan and Harri [23], most studies on URLLC consider only the Uu interface, where communication is mediated by the gNB. The authors also state that adjusting various 5G-NR RAN parameters, particularly in the MAC layer and physical layer resource block configurations, is crucial for meeting the requirements defined for the URLLC service. The following section describes the main URLLC features present in the RAN, with a focus on the sidelink.

Numerology. Similar to the 4G LTE standard, 5G NR also uses OFDM (Orthogonal Frequency Division Multiplexing) modulation at the physical layer. OFDM is a modulation technique in which the bandwidth is divided into frequency subcarriers that carry the modulated data [13]. However, unlike 4G, which uses a fixed subcarrier spacing (SCS) corresponding to a numerology value of 0, 5G allows the subcarrier spacing to be selected from seven numerologies (μ). Numerologies 0 to 2 are available for low and medium frequencies (below 6 GHz), known as Frequency Range 1 (FR1), while numerologies 2 to 6 are available for millimeter waves (24.25 to 52.6 GHz), known as Frequency Range 2 (FR2). The SCS can be calculated using the formula 15 kHz $\times 2^{\mu}$.

Regardless of numerology, 14 OFDM symbols are grouped into a time-domain slot with a duration of $1/2^{\mu}$ ms. As a result, higher numerologies are essential for low-latency communication in 5G. Table 2 summarizes the impact of numerology for μ values from 0 to 4, highlighting the Transmission Time Interval (TTI), symbol duration (DS), and the number of slots in a subframe (SS). By default, the slot duration is 1 ms. Higher numerologies significantly reduce transmission time and therefore directly impact latency.

Table 2: Numerology Table 5G NR.

μ	SCS	TTI	DS	SS	Use Case
0	15 kHz	1 ms	66,7μs	1	Compatibilidade com LTE
1	30 kHz	0.5 ms	33,3μs	2	Enhanced Mobile Broadband
2	60 kHz	0.25 ms	16,7μs	4	Low-latency services (FR1 e FR2)
3	120 kHz	0.125ms	8,3μs	8	URLLC em mmWave (FR2)
4	240 kHz	0.0625 ms	4,2μs	16	Synchronization in mmWave (FR2)

Mini-Slots. In 5G NR Sidelink, the minimum resource scheduling unit in the time domain is the full slot, typically consisting of 14 OFDM symbols with a normal cyclic prefix. A mini-slot is a transmission unit that uses fewer OFDM symbols, usually 2, 4, or 7. This allows the scheduler to transmit earlier, without waiting for the full slot duration. Numerology reduces latency by shortening the slot transmission time through increased subcarrier spacing, while the mini-slot allows the scheduler to initiate transmission before the full slot, further reducing latency. With numerology 0, the transmission time is 1 ms; with numerology 2, it drops to 0.25 ms. Without mini-slot scheduling, a 0.25 ms wait is required. With mini-slots, transmission can be accelerated based on the number of symbols used. For example, using only 2 symbols instead of 14 significantly reduces packet delivery time. Maghsoudnia et al. [16] argue that mini-slots suffer from higher network coordination complexity due to more granular scheduling, potentially limiting scalability. Due to increasing control signaling overhead, which grows with the number of UEs, mini-slots can reduce the system's overall efficiency.

Although NR version 15 allows the transmission of only a portion of a slot in both downlink and uplink (Uu interface), Sidelink does not support mini-slot operation for data transmission (PC5 interface) [12]. This applies in situations without gNB intermediation, such as Sidelink mode 2. Using full slots simplifies resource scheduling and sensing mechanisms, which is essential for Mode 2, where vehicles decide which resources to use.

Configured Grant. At the MAC layer, the 5G NR standard uses dynamic scheduling. With dynamic scheduling, the gNB informs UEs of the radio resources for each downlink packet before transmission. For uplink, the UE requests resources from the gNB, which responds with a grant and information about the allocated resources. This process introduces a non-negligible transmission delay, which is greater for uplink due to the larger number of messages exchanged between the UE and gNB. In contrast, Configured Grant pre-allocates radio resources periodically to UEs, eliminating the need to request scheduling and wait for a grant for each transmission. This removes the signaling overhead and delays associated with dynamic scheduling, enabling immediate transmissions [15, 12].

As described earlier, in mode 2, UEs can autonomously select their Sidelink resources from a resource pool. The resource pool can also be preconfigured by the gNB when the UE is within network coverage. When there is no base station to organize traffic, the devices manage this themselves through a process called Sensing and Selection [2].

A note about QoS. The mechanisms described above help achieve the QoS vision outlined in technical specifications 23.501 [6] and 23.287 [6] The first defines the QoS model for NR 5G networks using the Uu interface and introduces a series of 5G QoS Identifiers, called 5QIs. Among these, 5QIs 82 and 83 are specialized identifiers designed to support URLLC. Although originally defined for industrial automation, they are critical for advanced V2X (Vehicle-to-Everything) services such as platooning and cooperative driving. The second standard defines the QoS model for Sidelink based on the first, with the additional parameter of Range, and introduces a set of identifiers called PQIs (PC5 5QIs). The Range parameter allows the vehicle to decide, for example, that an emergency braking message is only "critical" for cars within a 200-meter radius, saving radio resources. Among PQI values, PQI 90 and 91 are for delay-critical applications.

The next subsection presents related work on Sidelink in combination with URLLC mechanisms such as numerology, mini-slot, and configured grant, aiming to reduce latency.

2.3 Related Work

The impact of numerology on URLLC and V2X is discussed in several works [10, 9, 19, 7, 13, 18]. Segura et al. [19] analyze the effect of numerology on delay at the radio link level in an industrial context using the LENA-5G simulator. Their results show that higher numerology does not always lead to lower delay; this depends on packet size and channel conditions. Campolo et al. [9] investigate the impact of numerology on autonomous access mode (LTE mode 4), where vehicles allocate transmission resources autonomously. Using the LTEV2Vsim[2] simulator, they confirm that higher numerologies increase the packet reception rate and reduce update delay. Zoraze et al. [7] examine the role of numerology in mode 2 (5G NR) using the LENA-5G simulator for both sensing-based and non-sensing-based resource selection. Similarly, Todisco et al. evaluate the impact of flexible subcarrier spacing and the configuration of modulation and coding schemes under different vehicle densities and data traffic patterns in Mode 2 using the open-source simulator WiLabV2Xsim.

Also in the V2X context, Valgas et al. [10] present a performance evaluation for different numerologies and device speeds, concluding that greater subcarrier spacing provides better protection against the Doppler effect. Khabaz et al. [13] state that selecting the appropriate numerology for each V2X scenario is particularly important and, using the Simu5G and Vienna 5G[3] simulators, demonstrate that the optimal numerology depends on propagation channel conditions and vehicle speed due to Intercarrier Interference (ICI) and Inter-Symbol Interference (ISI). Sayed et al. state that a mixed numerology system is essential for 6G networks. However, this approach faces challenges due to Inter-Numerology Interference (INI). To address this, the authors propose a user-based numerology and waveform approach that utilizes various OFDM-based waveforms and their parameters to mitigate intra-radioactive noise interference. By assigning a specific waveform and numerology to each user, the proposal reduces INI.

Weerackody et al. [22] investigate Sidelink Mode 2 in unlicensed bands, considering the 3GPP initiatives in Release 18. The authors provide the first analytical framework to quantify Sidelink performance in unlicensed bands under saturation conditions, and their results show that throughput loss is mainly due to packet collisions. Elleuch et al. [11] developed and implemented the repeater function as an extension of the Simu5G library in the OMNeT++ simulator for a mission-critical service. The implementation is not publicly available and supports only UEs within the gNB coverage area, i.e., mode 1.

Maghsoudnia et al. [16] investigate whether and how URLLC requirements have been met nearly a decade after their initial conception. Although not focused on Sidelink, this work raises important questions. The authors find it unclear in which architecture and network configuration such latencies can be achieved. They analyze the factors contributing to increased latency and categorize the origins of latency into three groups: protocol, processing, and radio. Through this analysis and a demonstration in a real experimental environment, the authors conclude that URLLC is theoretically possible, but only under very specific circumstances and with stringent hardware and software requirements. They also state that more research is needed before 5G systems can reliably enable URLLC.

Yan and Harri [24] evaluate the feasibility of meeting critical URLLC requirements for V2X Sidelink 5G NR communication in the 5.9 GHz band. Among their contributions, they assess the impact of numerology 3 on FR1 and introduce an admission control mechanism that limits the number of V2X UEs to ensure URLLC requirements are met. In another work [24], the same authors present a framework that integrates network slices tailored for platoon communication. Separate slices are allocated for V2X communication and vehicle platooning to minimize interference and ensure independent operation. The slice for vehicle platooning specifically supports URLLC for critical intra-platoon messages. Furthermore, ProSe is used to advertise platoon services and manage platoon entry and exit functions.

[2] https://github.com/alessandrobazzi/LTEV2Vsim

[3] https://www.tuwien.at/etit/tc/en/vienna-simulators/vienna-5g-simulators/

3 Evaluation methodology

A key issue in advancing research and development of 5G NR solutions is the use of simulators. Simulation platforms such as Vienna5G, WiLabV2Xsim, LENA 5G, and Simu5G are widely used to simulate features and innovation proposals in the context of 5G NR. While simulators provide an excellent starting point for simulating both Sidelink and URLLC, none currently support all the features and functionalities designed for both Sidelink and URLLC.

An important question is which features of Sidelink and URLLC are actually supported together by the main simulators. To begin addressing this question, this evaluation focuses on the Simu5G simulator. Simu5G implements the 5G RAN data plane according to Release 16 and enables simulation of various 4G and 5G scenarios. The protocol layers are 3GPP compliant, and physical transmission is modeled using realistic channel models. Resource scheduling in the uplink, downlink, and sidelink directions is supported, along with carrier aggregation and multiple numerologies, as specified by the 3GPP standard. Its D2D implementation is adapted for NR 5G from SimuLTE. Figure 2 shows the data flow, where datagrams from the IP layer are split into one of two branches at the Packet Data Convergence Protocol (PDCP) layer and processed accordingly at lower layers.

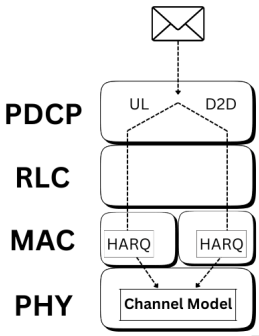

Figure 2: Data flow from the sender UE perspective [21].

The simulation evaluates the native features of both D2D and URLLC communication available in the latest stable version of the simulator. For this purpose, a D2D communication scenario provided in Simu5G will be used. Native support means the simulator offers these resources without requiring extensions. Figure 3 shows the simulation scenario. There is a transmitter (ueD2DTx) and a receiver (ueD2DRx) within the gNB coverage area. Although the gNB is connected to the network core, communication between the UEs occurs through the gNB and does not pass through the network core. In the MAC layer, a Hybrid ARQ (H-ARQ) scheme allows a configurable number of retransmissions and provides reliability. For Sidelink, native support for both mode 1 and mode 2, as well as support for ProSe, will be verified. In the context of URLLC, the simulation will check whether the simulator supports numerology, mini-slots, and configured concessions in D2D communication.

To determine whether the simulator supports the aforementioned features, both the code and the documentation available for Simu5G on its website and discussion forum will be reviewed. Published, peer-reviewed articles on the subject will also be considered.

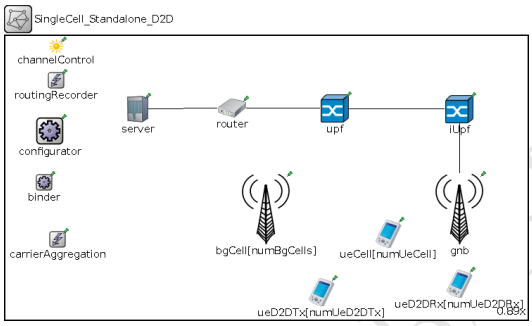

Figure 3: D2D simulation scenario.

To analyze the effectiveness of the URLLC features present in the simulator, end-to-end latency will be evaluated. First, latency will be measured in a standard communication scenario. Then, with each addition of available URLLC features, new latency measurements will be taken. This article does not analyze the reliability features of URLLC, leaving that for future work.

4 Results

The results were obtained using version 1.4.1 of Simu5G [4]. According to the documentation, Simu5G supports network-controlled D2D communication (mode 1) as a legacy of the D2D implementation in SimuLTE [17, 21]. Key information about Simu5G's support for D2D communication in 5G NR is available in Nardini et al [17]. In the network-controlled D2D model, data is sent via Sidelink, while the base station controls resource allocation and manages conflicts and interference.

Mode 2 support is not available in the simulator version, and there is no mention of Mode 2 in its documentation. In Mode 1, the simulator supports both point-to-point (P2P) and point-to-multipoint (P2MP) Sidelink communications. In P2P communication, the UE sends a message to a single receiver, which confirms receipt. In P2MP communication, the UE sends messages to neighboring UEs in a multicast group. The implementation scenario available in Simu5G supports only P2P communication.

Messages sent via Sidelink traverse all layers of the 5G NR protocol stack, similar to uplink and downlink messages. Communication involves UE IPs, and there is no discovery service. As Figure 2 shows, the PDCP layer assigns a Logical Connection Identifier (LCID) to the incoming data flow according to a 5-tuple defined by source/destination IP address/port, and flow direction. This creates different LCIDs for flows having different transmission directions, allowing lower layers to distinguish UL and D2D flows. In the simulation, the communication flow is statically defined. For resource allocation, Simu5G models dynamic scheduling of D2D transmissions. Whenever a UE has data to send using D2D, it sends a request to the gNB, which schedules Sidelink resources and issues a Sidelink grant to the UE.

Simu5G provides several D2D simulation scenarios that allow users configure an increasing number of devices and choose either

[4] https://github.com/Unipisa/Simu5G

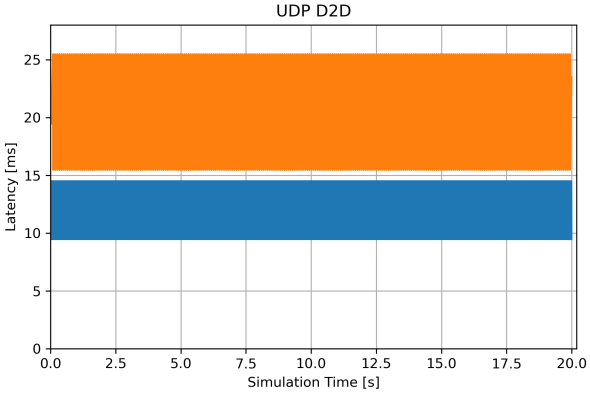

Figure 4: Latency for UDP Infra and UDP D2D modes.

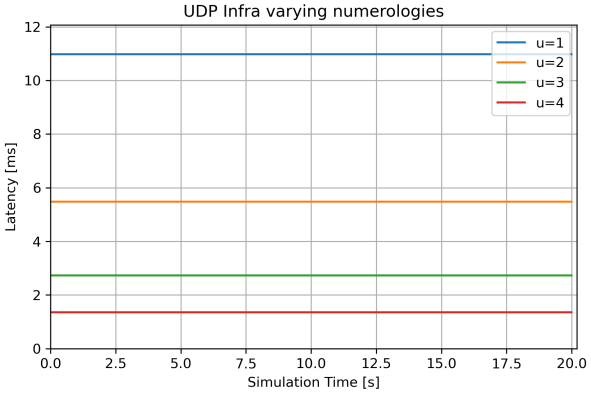

Figure 5: D2D UDP with different numerologies.

UDP or TCP as the protocol. These scenarios simulate communication between two UEs within the gNB coverage area. There are two communication modes: Infrastructure Mode and D2D Mode. In Infrastructure Mode, the two UEs communicate using the traditional infrastructure approach – a two-hop path through the gNodeB – so all communication between the UEs is managed by the gNB. In D2D Mode, communication between the two UEs occurs via Sidelink, with the gNB providing only control information to the UEs. The simulator enables switching between Infrastructure Mode and D2D Mode based on a policy that selects the mode with the best channel quality.

Figure 4 shows the communication latency between two UEs in Infra and D2D modes using the UDP protocol for a 1000-byte message. D2D mode achieves communication in 11.97 ms, compared to 21.60 ms using the communication infrastructure, a reduction of approximately 50%. This pattern remains consistent as the number of UEs increases.

Next, simulations were conducted to evaluate the simulator's ability to support different numerologies in UDP Infra and D2D modes. As described in [17], Simu5G implements numerologies using the *carrierAggregation (CA)* module, which stores all information related to the *Carrier Component (CC)*. In Simu5G, different numerologies (μ) can be associated with each CC. Each *componentCarrier* module has its own μ parameter, which can be configured via NED/INI. Figure 5 shows the latency results for $\mu = 1, 2, 3, 4$ in the UDP Infra scenario. Initially, the UDP Infra mode showed a latency of 21.6 ms, but the latency is halved with each increase in μ, following the pattern $1/2^{\mu}$. Although the simulation ran successfully for Infra Mode, when simulating D2D mode, the simulator launches an error, as described in the issue opened on the simulator's GitHub page: https://github.com/Unipisa/Simu5G/issues/287.

Based on the results obtained and information from the simulator's website and discussion forums, it appears that Simu5G has only partial support for Sidelink, as it does not natively implement Sidelink mode 2, for example. Regarding numerology, the simulator supports it as stated in its documentation and demonstrated in the simulations. In the Infra Mode experiments, applying different numerologies was effective, but it was not possible to observe the effect of numerology on Sidelink communication. Mini-slots and configured grants, although essential for implementing the URLLC service, have not yet been incorporated into the latest version of the simulator. Despite supporting numerologies, one of the mechanisms for URLLC, Simu5G's resource scheduling is dynamic and there is still no support for configured leases. As described earlier, only one full slot is transmitted via Sidelink, however Simu5G's support for Mini-Slots on the Uu interface is supported as indicated at https://github.com/Unipisa/Simu5G/issues/88. There is also no support for the Proximity Service.

5 Conclusion

This article investigates and evaluates Sidelink communication in 5G networks, focusing on the URLLC service. It describes the evolution of Sidelink and URLLC since the 3GPP releases. The Simu5G simulator, a leading tool for 5G network simulation, was used for the evaluation. In Sidelink, mode 2 enables communication with UEs outside gNB coverage. For URLLC, features such as numerologies, mini-slots, and configured grant can affect latency, although the mini-slot is not intended for direct communication between two UEs.

The evolution of Sidelink and URLLC in releases 16, 17, and 18 is not reflected in major simulators such as Simu5G. Simu5G supports only Sidelink mode 1 and different numerologies. Future work includes exploring QoS, once the PC5 interface has QoS support [12], investigating mechanisms to guarantee reliability in URLLC service, and expanding the evaluation by comparing additional simulators.

6 Acknowledgments

This work was partially funded by Fundação de Apoio da UFMG (Fundep), through Linha VI – Conectividade Veicular, a prioritary program from Mover (Mobilidade Verde e Inovação), project Auto5G (29271.02.01/2022.01-00).

References

[1] 3GPP. 2019. Release 15 description. 3GPP. 3GPP TR 21.915 V15.0.0. (Sept. 2019).
[2] 3GPP. 2022. Release 16 description. 3GPP. 3GPP TR 21.916 V16.2.0. (June 2022).
[3] 3GPP. 2023. Release 17 description. 3GPP. 3GPP TR 21.917 V17.0.1. (Jan. 2023).
[4] 3GPP. 2018. Study on enhancement of 3gpp support for 5g v2x services. 3GPP. 3GPP TR 22.886 V16.2.0. (Dec. 2018).
[5] 3GPP. 2021. Study on system enhancement for proximity based services (prose) in the 5g system (5gs). 3GPP. 3GPP TR 23.752 V17.0.0 (2021-03). (Mar. 2021).

[6] 3GPP. 2025. System architecture for the 5g system (5gs). 3GPP. 3GPP TS 23.501 V20.0.0. (Dec. 2025).

[7] Zoraze Ali, Sandra Lagén, and Lorenza Giupponi. 2021. On the impact of numerology in nr v2x mode 2 with sensing and no-sensing resource selection. (2021). https://arxiv.org/abs/2106.15303 arXiv: 2106.15303 [eess.SP].

[8] Nestor Bonjorn, Fotis Foukalas, and Paul Pop. 2018. Enhanced 5g v2x services using sidelink device-to-device communications. In *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 1–7. doi:10.23919/MedHocNet.2018.8407085.

[9] Claudia Campolo, Antonella Molinaro, Francesco Romeo, Alessandro Bazzi, and Antoine O Berthet. 2019. 5g nr v2x: on the impact of a flexible numerology on the autonomous sidelink mode. In *2019 IEEE 2nd 5G World Forum (5GWF)*. IEEE, 102–107.

[10] J Flores De Valgas, David Martín-Sacristán, and Jose F Monserrat. 2018. 5g new radio numerologies and their impact on v2x communications. *Waves, Univesitat Politecnica de Valencia*, 15–22.

[11] Wajdi Elleuch, Patrick Sondi, Ahmed Meddahi, and Sylvain Lecomte. 2025. Evaluation of 5g relay-empowered and device-to-device communications for rescue mission. *IEEE Access*, 13, 104614–104629. doi:10.1109/ACCESS.2025.3579872.

[12] Mario H. Castañeda Garcia, Alejandro Molina-Galan, Mate Boban, Javier Gozalvez, Baldomero Coll-Perales, Taylan Şahin, and Apostolos Kousaridas. 2021. A tutorial on 5g nr v2x communications. *IEEE Communications Surveys & Tutorials*, 23, 3, 1972–2026. doi:10.1109/COMST.2021.3057017.

[13] Sehla Khabaz, Kaouthar Ouali Boulila, Thi Mai Trang Nguyen, Guy Pujolle, Moustapha El Aoun, and Pedro B Velloso. 2022. A comprehensive study of the impact of 5g numerologies on v2x communications. In *2022 13th International Conference on Network of the Future (NoF)*. IEEE, 1–9.

[14] Katerina Koutlia, Biljana Bojovic, Zoraze Ali, and Sandra Lagén. 2022. Calibration of the 5g-lena system level simulator in 3gpp reference scenarios. *Simulation Modelling Practice and Theory*, 119, 102580. doi:https://doi.org/10.1016/j.simpat.2022.102580.

[15] Ana Larrañaga, M. Carmen Lucas-Estañ, Sandra Lagén, Zoraze Ali, Imanol Martinez, and Javier Gozalvez. 2023. An open-source implementation and validation of 5g nr configured grant for urllc in ns-3 5g lena: a scheduling case study in industry 4.0 scenarios. *Journal of Network and Computer Applications*, 215, 103638. doi:https://doi.org/10.1016/j.jnca.2023.103638.

[16] Arman Maghsoudnia, Eduard Vlad, Aoyu Gong, Dan Mihai Dumitriu, and Haitham Hassanieh. 2024. Ultra-reliable low-latency in 5g: a close reality or a distant goal? In *Proceedings of the 23rd ACM Workshop on Hot Topics in Networks* (HotNets '24). Association for Computing Machinery, Irvine, CA, USA, 111–120. ISBN: 9798400712722. doi:10.1145/3696348.3696862.

[17] Giovanni Nardini, Dario Sabella, Giovanni Stea, Purvi Thakkar, and Antonio Virdis. 2020. Simu5g–an omnet++ library for end-to-end performance evaluation of 5g networks. *IEEE Access*, 8, 181176–181191. doi:10.1109/ACCESS.2020.3028550.

[18] Mohamed S. Sayed, Hatem M. Zakaria, and Abdelhady M. Abdelhady. 2025. Enhancing flexibility and system performance in 6g and beyond: a user-based numerology and waveform approach. *Digital Communications and Networks*, 11, 4, 975–991. doi:https://doi.org/10.1016/j.dcan.2024.10.020.

[19] David Segura, Emil J. Khatib, Jorge Munilla, and Raquel Barco. 2021. 5g numerologies assessment for urllc in industrial communications. *Sensors*, 21, 7. doi:10.3390/s21072489.

[20] Theodoros Tsourdinis, Nikos Makris, Thanasis Korakis, and Serge Fdida. 2024. Demystifying urllc in real-world 5g networks: an end-to-end experimental evaluation. In *GLOBECOM 2024 - 2024 IEEE Global Communications Conference*, 2954–2959. doi:10.1109/GLOBECOM52923.2024.10901776.

[21] Antonio Virdis, Giovanni Nardini, and Giovanni Stea. 2016. Modeling unicast device-to-device communications with simulte. In *2016 1st International Workshop on Link- and System Level Simulations (IWSLS)*, 1–6. doi:10.1109/IWSLS.2016.7801579.

[22] Vijitha Weerackody, Hao Yin, and Sumit Roy. 2025. Nr sidelink mode 2 in unlicensed bands: throughput model and validation. *IEEE Transactions on Communications*, 73, 1, 216–229. doi:10.1109/TCOMM.2024.3435554.

[23] Jin Yan. 2024. *Towards Dependable 5G-NR Sidelink Communication*. Ph.D. Dissertation. Sorbonne Université.

[24] Jin Yan and Jérôme Härri. 2024. Towards 5g nr prose-based platoon services supporting urll vehicular communication. In *ICC 2024 - IEEE International Conference on Communications*, 1017–1022. doi:10.1109/ICC51166.2024.10622239.

Designing Cyberphysical Systems Software Components for Middleware Interoperability

Matheus Wagner
wagner@lisha.ufsc.br
Software/Hardware Integration Lab. Federal University of Santa Catarina
Florianópolis, Santa Catarina, Brazil

Antônio Augusto Fröhlich
guto@lisha.ufsc.br
Software/Hardware Integration Lab. Federal University of Santa Catarina
Florianópolis, Santa Catarina, Brazil

Abstract

Modern software frameworks for cyber–physical systems rely on component-oriented architectures, yet existing frameworks often couple these principles to specific operating systems and middleware technologies, limiting design flexibility and interoperability. This paper presents Abstraction-Based And Component-Oriented Systems (ABACOS), an abstraction-based framework that decouples component execution from communication mechanisms through a small set of orthogonal software abstractions. Components interact only through *Input* and *Output Ports*, allowing different middleware backends to be bound at deployment time without modifying functional behavior. A C++ implementation of ABACOS was evaluated using Data Distribution Service (DDS) and Scalable Service-Oriented Middleware over IP (SOME/IP) transports, with end-to-end latency measurements collected across multiple payload sizes. The results show that the overhead introduced by ABACOS remains small, with median values of only a few microseconds and is largely independent of payload size, indicating that interoperability can be achieved without materially altering the timing characteristics of the underlying middleware. These findings support the use of ABACOS as a foundation for low-overhead, middleware-agnostic CPS software architectures.

Keywords

Middleware Technologies, Component-Based Software Design, Low-Overhead Communication

1 Introduction

The introduction of autonomous Cyber-Physical Systems (CPSs) — notably Autonomous Vehicles (AVs), Unmanned Aerial Vehicles (UAVs), and Autonomous Robots (ARs) — has fundamentally altered the software-engineering landscape for CPSs. These systems require tightly integrated, heterogeneous architectures in which subsystems must cooperate under strict timing, safety, and computational-performance constraints. As a result, modern CPSs software design exhibits unprecedented complexity [14, 20], motivating the development of software frameworks to support the software-engineering process for such systems, particularly those capable of enabling interoperability across diverse middleware platforms while preserving low execution overhead.

Several software frameworks for CPSs have gained prominence in recent years. Frameworks such as *ROS2* [16], *OROCOS* [4], *OpenRTM* [2], and *XBot2* [13] have been developed by the robotics community, each addressing distinct challenges in robotics software development, including modularity, real-time constraints, and hardware abstraction. In the automotive domain, *AUTOSAR Adaptive* combined with SOME/IP has been introduced to support computationally demanding tasks that exceed the capabilities of classical AUTOSAR implementations [17]. In all of these frameworks, interoperability is typically achieved by adhering to a specific middleware or communication technology, which in turn couples component behavior to the underlying stack and may introduce execution overhead that is difficult to control or assess.

Despite their diversity, these frameworks share several fundamental architectural principles. First, they adopt a component-oriented design, in which the system is decomposed into a set of components with orthogonal responsibilities that interact to realize system-level functionality. Second, they employ loosely coupled communication, whereby components can asynchronously produce and consume data without knowledge of the identity or number of other producers and consumers. This approach makes such frameworks particularly well suited for distributed, modular, and real-time systems [18]. However, when these abstractions are bound to a particular middleware implementation, interoperability across heterogeneous technologies becomes difficult, and the performance impact of the selected stack, particularly its communication overhead, often becomes inseparable from the application design itself.

The effectiveness of this approach to CPSs software design is evidenced by its adoption in large-scale projects, such as *Autoware* [9] and *Apollo Auto* [21]. Nonetheless, the inherent opacity of many frameworks that employ these design strategies limits designers' ability to tune systems for specific application requirements and impedes interoperability among components developed using different frameworks or communication technologies, especially when attempting to reconcile performance constraints with middleware heterogeneity.

A key requirement for a software framework is that it minimizes the constraints imposed on application design, allowing the designer, within the context of the application, to enforce constraints specific to their requirements rather than being restricted by the framework itself [1]. Equally important is the timing of design decisions: whenever possible, decisions should be postponed and ideally deferred until deployment, thereby maximizing flexibility for choices that affect nonfunctional properties such as performance, reliability and scalability [3].

From a practical standpoint, a major limitation of current frameworks for CPSs software development is their tight coupling with specific communication stacks and Operating Systems (OSs). Although the functional behavior of a component is inherently independent of the particular communication protocol used for message exchange and the OS managing system resources, the choice of a

framework imposes implicit constraints on these technologies. This, in turn, restricts the designer's ability to tune the system to meet nonfunctional requirements such as timing predictability, latency, or resource consumption. For instance, *ROS2* and *AUTOSAR Adaptive* with SOME/IP are closely tied to IP-stack protocols and POSIX-like OSs, whereas frameworks such as *OROCOS* and *OpenRTM* rely heavily on CORBA for communication, thereby constraining the design space available for communication and resource management. The negative consequences of these design choices are well documented, as evidenced by numerous studies that seek to mitigate the impact of communication stacks and OSs on real-time performance [7, 17].

With the objective of decoupling the proven model of component-oriented design with loosely coupled communication, as employed in modern frameworks for CPSs software development, from the choice of underlying technologies that may constrain the system's ability to satisfy nonfunctional requirements, this work introduces the concept of an ABACOS framework, a framework for software development for CPSs. This framework achieves its objective by ensuring that the component model depends solely on abstract specification of the service interfaces required from the OS and communication stack, rather than on their concrete implementations. Interoperability is thus realized through abstract communication and execution primitives, such as input and output ports, behavior bindings, and execution pipelines, that can be mapped to multiple middleware technologies while preserving predictable timing behavior and minimizing overhead in cross-middleware integration. Consequently, the concrete implementation of these abstract services must be explicitly specified and chosen by the designer as part of the system configuration.

The remainder of this work is organized as follows. Section 2 reviews the related literature; Section 3 presents the ABACOS architecture in detail, emphasizing interoperability abstractions, while Section 4 provides the associated implementation details. Section 5 discusses the procedure employed to evaluate the overhead introduced by ABACOS and Section 6 presents the results. Finally, Section 7 presents the concluding remarks.

2 Related Works

Component-based models built on abstract communication interfaces, as well as flexible frameworks that provide designers with a rich configuration space and architectural choices, are not novel concepts [1, 11]. Numerous efforts in the literature have sought to apply these principles to the design of modern frameworks for CPSs software, often with the goal of improving portability and extensibility across platforms and execution environments. However, most existing approaches do not explicitly address interoperability across heterogeneous middleware technologies while preserving low execution overhead as a first-class design objective.

The XBot2 framework [13] was developed with the explicit goal of being adaptable to any host OS or Real-Time Operating System (RTOS) through the use of an OS abstraction layer; however, it is still tightly bound to the ROS ecosystem along with its communication stack. The use of OS abstractions is also present in other frameworks, such as OROCOS [4] and OpenRTM [2]. However, their communication abstractions are provided exclusively through CORBA, which constrains these frameworks entirely to that middleware. As a result, the design space for inter-component communication is severely restricted by the absence of a more general abstraction layer. Similarly, AUTOSAR Adaptive relies on a POSIX-based OS abstraction and is therefore limited to POSIX-compliant OSs [17]. In addition, it adopts SOME/IP as its communication abstraction, which further restricts its applicability to IP-based communication technologies and makes interoperability with alternative middleware stacks dependent on bridging mechanisms that may introduce additional latency and overhead.

The most widely adopted framework for CPS software development, ROS2, is tightly coupled to a POSIX-based OS abstraction and relies heavily on DDS as its communication middleware. Although DDS is defined as a middleware specification, widely used open-source implementations typically target POSIX-compliant OSs. Nonetheless, efforts have been made to develop portable implementations of DDS with the goal of broadening the range of platforms supported by ROS2 [8], complemented by several initiatives aimed at enhancing the portability of ROS and enabling its execution on real-time platforms [5]. These initiatives improve portability but do not fully decouple component behavior from the underlying communication stack, meaning that interoperability and performance properties remain largely tied to the chosen middleware implementation.

The limitations of these frameworks have been the subject of a significant body of research. Several works critique the lack of strong real-time guarantees in ROS2, identifying issues such as latency spikes and jitter under load [12, 22]. Others focus on bridging between frameworks, such as interoperability between ROS2 and AUTOSAR [6], or developing portable DDS-based implementations to expand deployment contexts [10]. Survey papers have also attempted to classify frameworks for CPSs by their architectural choices, supported domains, and communication models, consistently highlighting portability, extensibility, and middleware interoperability as open challenges [19].

Despite these efforts, there remains a gap in the literature for a framework that systematically decouples the component model from the underlying operating system and communication technologies, while simultaneously enabling interoperability across heterogeneous middleware stacks with minimal execution overhead. Existing frameworks either strongly couple to a given OS and middleware (as in ROS2, OROCOS, OpenRTM, and AUTOSAR Adaptive), or they provide portability in a narrow and domain-specific manner. In contrast, ABACOS explicitly addresses this gap by defining a component-oriented model that depends only on abstract service specifications—including abstract communication ports and execution bindings—leaving the binding to concrete OS and communication implementations (e.g., DDS or SOME/IP) as a configuration step. This architecture enables interoperability without embedding middleware-specific assumptions into the component model, thereby allowing designers to reason explicitly about the performance and overhead introduced by each underlying technology.

3 Abstraction-Based and Component-Oriented System Architecture

The ABACOS architecture is built upon a minimal yet expressive set of abstractions centered on the notion of a *Component*, which represents a functional unit of the overall system with a single, well-defined responsibility. As illustrated in Figure 1, an ABACOS *Component* interacts with three additional abstractions: the *Node*, responsible for managing the lifecycle of a set of components; the *Behavior*, which encapsulates the response of a component to a particular input; and the *Input and Output Ports*, which enable communication with other components in a distributed setting.

The *Component* is composed of a map that relates ports and behaviors, expressing that the arrival of data on a given port triggers the execution of its associated behaviors. Each component also maintains an event queue that registers the events observed on its ports and is used to select the next behavior to be executed according to a given scheduling strategy. Finally, a single *Thread* executes behaviors sequentially, following the event-queue scheduling policy, which induces run-to-completion semantics for the *Component* execution model.

Behaviors are a convenient abstraction of units of work. In practice, they abstract callable entities much like a functor, with the fundamental difference that the input data to the callable is stored in a buffer owned by the behavior rather than being passed as an argument at the moment of invocation. This design has two major implications. First, it decouples the scheduling of behavior execution from data arrival, allowing behaviors to be dispatched according to any policy selected for the event queue. Second, it segregates execution and data-ingestion semantics; consequently, the interfaces exposed to the *Component* ports and to the event-scheduling mechanism are orthogonal, reflecting the fact that these concerns are conceptually independent.

From the perspective of interoperability across different communication mechanisms, the *Port* interface plays a central role, as it abstracts the communication boundaries of a *Component*. In ABACOS, *Ports* are conceived purely as interfaces: from the viewpoint of a concrete *Component* implementation, its responsibilities are limited to specifying how it reacts to incoming data and how new data may be produced as a consequence of such reactions, without any reference to how data is transported to or from the component.

For this reason, the *Output Port* exposes a single method that enables the component to write data, delegating all transport concerns to the bound communication backend. Conversely, *Input Ports* must support the decoupling between execution and data arrival enforced by *Behaviors*. They therefore provide two methods: one for registering the function used by a behavior to ingest incoming data, and another for registering the function through which the component is notified of the occurrence of an input event. The latter notification is handled internally by the component and used to decide when the corresponding behavior should be scheduled for execution.

This enables different communication mechanisms to be selected at design and deployment time—such as switching between a simulation backend and a production communication stack—without altering the functional behavior of the component.

The implication of this design is straightforward: the concrete implementations of *Input and Output Ports* may be exchanged at the discretion of the designer without affecting the functional behavior of a given *Component*. This enables different communication mechanisms to be adopted at design and deployment time—such as switching between a simulation backend and a production communication stack—while preserving *Component* semantics. Moreover, distinct communication mechanisms may be assigned independently to different ports of the same component, allowing it to operate across heterogeneous communication domains without modification and thereby supporting interoperability by construction.

With the principal abstractions defined, the execution cycle of an ABACOS component is depicted in the sequence diagram of Figure 2, assuming a canonical publish–subscribe interaction model (while remaining valid for any peer-to-peer transport configuration). The *Publisher Component* produces data through one of its *Output Ports*, which delegates transmission to the configured communication backend and delivers the message to the corresponding *Input Ports* of other components. Upon reception, the *Input Port* forwards the data to the ingestion interface of the associated *Behavior* and, in parallel, issues an event notification to the *Component* identifying the port on which the event occurred. The *Component* records this notification in its event queue and, according to its scheduling policy, selects the next event to be processed, resolves the behavior bound to the notified *Input Port*, and invokes it. The bound function defined by the *Subscriber Component* is thus executed under run-to-completion semantics as part of the component's execution loop. This execution model makes explicit the separation between communication, event handling, and behavior execution, thereby supporting systematic analysis and predictable timing behavior.

An important architectural consequence of this model is the explicit separation between the communication domain and the execution domain. Communication occurs exclusively through *Ports*, whereas execution is driven solely by events delivered to the component's event queue. As a result, changes to the communication backend—such as replacing one middleware or transport mechanism with another—affect only the concrete *Port* implementations and have no impact on scheduling semantics, behavior logic, or dataflow dependencies inside the component. This property is essential for supporting deployment across heterogeneous platforms and middleware ecosystems while preserving functional and temporal consistency.

The execution model adopted by ABACOS further promotes analyzability by enforcing single-threaded, run-to-completion execution at the level of each component, while still allowing concurrency to emerge at the system level through multiple independently executing components. This approach reduces the need for intra-component synchronization, eliminates race conditions on the component's state, and enables timing analysis to be performed at the granularity of individual components. Alternative scheduling policies or execution models may be incorporated as extensions to the event-queue mechanism, but the underlying abstraction boundaries remain unchanged, ensuring that evolutions in scheduling strategy do not interfere with communication or interface semantics.

In summary, the abstractions introduced in this section define a component execution model that preserves functional modularity while enabling flexible binding to heterogeneous communication

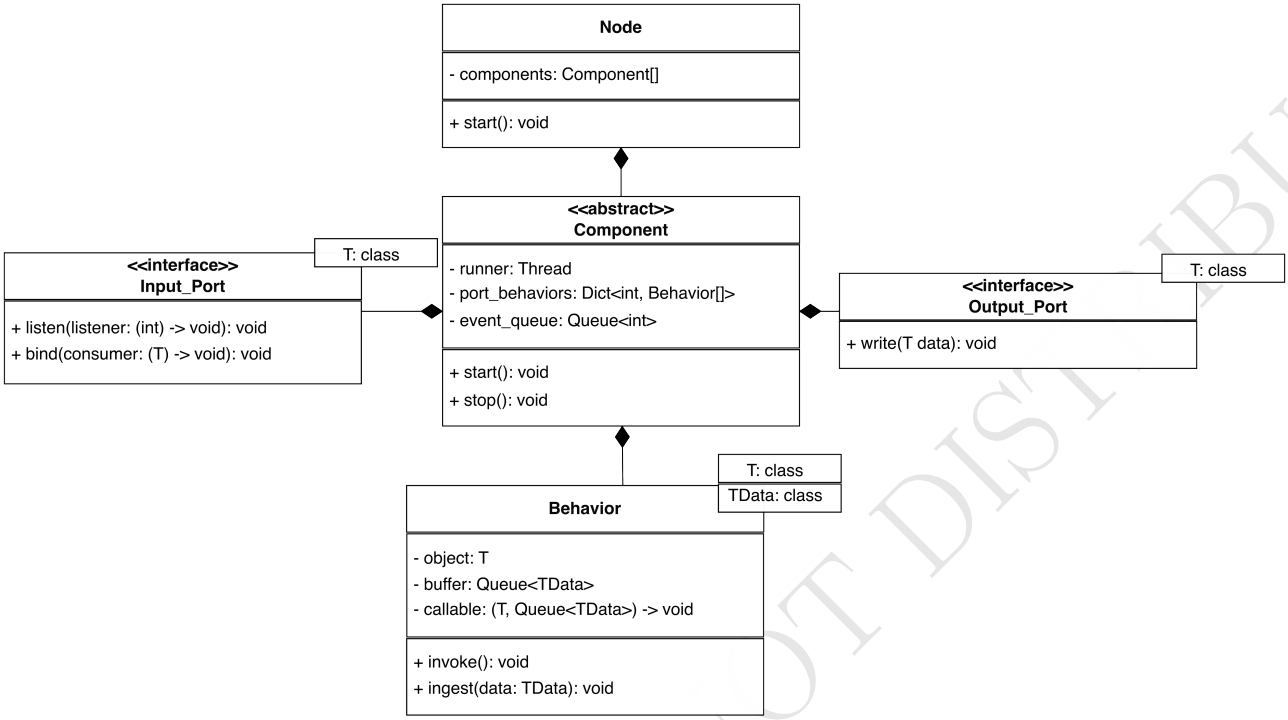

Figure 1: Class diagram of core ABACOS abstractions

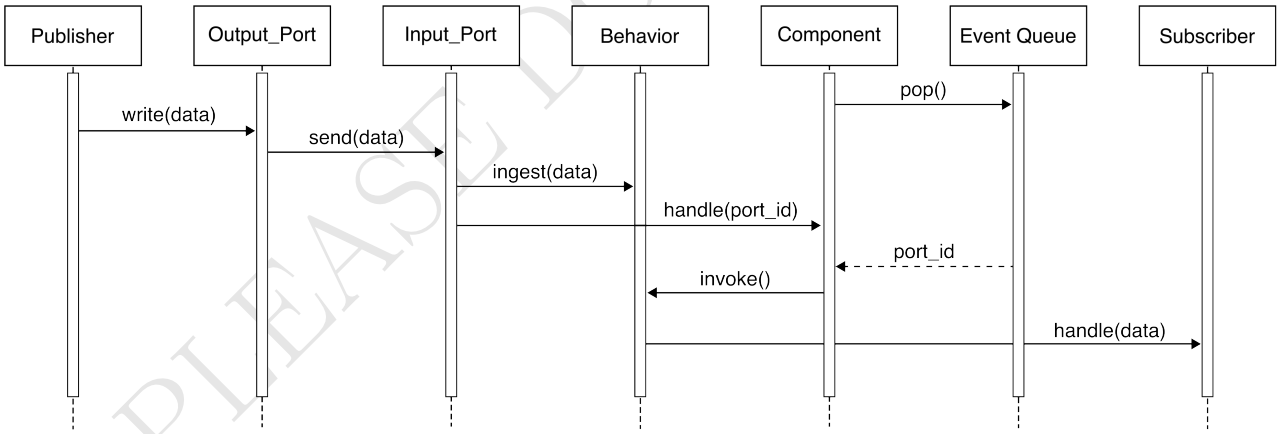

Figure 2: Sequence diagram of an ABACOS component execution cycle

mechanisms. By separating data transport, event handling, and behavior execution into orthogonal concerns, ABACOS provides a principled foundation upon which interoperability can be achieved without compromising timing predictability or implementation efficiency. This architectural structure further ensures that communication backends may be exchanged, configured, or specialized at design or deployment time, while the functional semantics of components and their behaviors remain invariant. As later sections will demonstrate, this decoupling is essential to supporting low-overhead integration across multiple middleware technologies in real-world cyber–physical systems.

4 ABACOS implementation details

ABACOS was implemented in C++ using a carefully selected set of language features aimed at achieving a principled balance between performance, expressiveness, and implementation ergonomics. Listing 1 illustrates the implementation of a simple ABACOS *Component*. The interfaces for *Input and Output Ports* are defined as C++ concepts, which constrain the admissible types used to instantiate these templates while avoiding the overhead and semantic rigidity associated with virtual-function dispatch. The combination of concepts and templates ensures that the communication mechanisms bound to a *Component* can be exchanged without modifying its functional implementation, allowing the concrete transport backends to be selected at deployment time. Moreover, the use of templates on input and output data types enables a single component to operate over semantically compatible, yet not necessarily identical, message definitions—such as messages generated from distinct Interface Definition Languages (IDLs)—naturally supporting the construction of lightweight gateways across heterogeneous communication domains.

The *Behavior* abstraction is parameterized by the type that owns the methods it encapsulates, as illustrated in Listing 1. Its execution and ingestion interfaces are exposed through a lightweight delegate mechanism, implemented as a pair consisting of a pointer to the target object and a pointer to the associated member function. In this design, an *Input Port* maintains references only to delegates, and therefore remains decoupled from specific component types and does not rely on inheritance or virtual interfaces to invoke the associated callbacks. This approach introduces only a single level of indirection while preserving static type safety and avoiding dynamic dispatch. The same delegate mechanism is employed for both the ingestion of input data and the execution of the behavior itself, reinforcing the separation between communication, event notification, and functional execution while maintaining minimal runtime overhead.

The concurrency model adopted in the implementation mirrors the architectural execution semantics. Each component executes within a single dedicated thread, ensuring that behaviors run atomically with respect to one another and eliminating the need for internal synchronization mechanisms. Concurrency is expressed at the system level by composing multiple components, each with its own execution thread, which allows independent timing domains to coexist while maintaining predictable behavior within each component boundary.

Memory management and data movement were also carefully considered to minimize unnecessary copies and avoid hidden performance penalties. Data is ingested through the *Input Port* into buffers owned by the corresponding *Behavior*, ensuring that message lifetime and ownership remain explicit. When supported by the underlying middleware, zero-copy or move-enabled data transfer can be exploited without altering component logic, whereas backends that require copying remain compatible through the same abstraction. This design allows performance and isolation trade-offs to be configured at deployment time, rather than being baked into the component implementation.

Listing 1: Implementation of a simple ABACOS component

```
template <typename TOutput_Port, typename TInput_Port, typename
      TInput_Data, typename TOutput_Data>
requires(Output_Port_Concept<TOutput_Port, TOutput_Data> &&
            Input_Port_Concept<TInput_Port, TInput_Data>)
class Simple_Component : public Component
{
    public:
        template <typename... TInput_Port_Args, typename...
      TOutput_Port_Args>
        Simple_Component(std::tuple<TInput_Port_Args...>
      input_port_args, std::tuple<TOutput_Port_Args...>
      output_port_args)
              : input_port_(std::make_from_tuple<TInput_Port>(
                  std::move(input_port_args))),
              output_port_(std::make_from_tuple<TOutput_Port>(
                  std::move(output_port_args))),
              behavior_(
                Behavior<
                    Simple_Component,
                    TInput_Data,
                    TInput_Port>::template
      create_behavior<&Simple_Component::handle>(this, &input_port_))
        {
            bind_behavior_to_input_port(
                &input_port_,
                behavior_.create_behavior_delegate());
        }

    private:
        void handle(const TInput_Data &data)
        {
            TOutput_Data output;

            // behavior implementation.

            output_port_.write(output);
        }

        Behavior<Simple_Component, TInput_Data, TInput_Port>
      behavior_;
        TOutput_Port output_port_;
        TInput_Port input_port_;
};
```

Finally, the separation between abstract interfaces and concrete transport bindings enables multiple deployment profiles to be realized from the same code base. The same component may be executed with different middleware backends for simulation, prototyping, or embedded deployment simply by selecting different *Port* implementations, without modifying component logic or behavior definitions. In this way, the implementation remains faithful to the architectural objective of supporting interoperability while retaining low overhead and predictable execution semantics.

The objective of the experimental evaluation is to assess the extent to which the abstractions introduced by ABACOS enable interoperability across multiple communication mechanisms while preserving low execution overhead and predictable timing behavior. To this end,a series of experiments were conducted comparing the performance of ABACOS components operating over different middleware backends, focusing in particular on the latency characteristics of data exchange and on the relative overhead introduced by the ABACOS execution model.

5 Experimental Evaluation

The experimental evaluation focused on assessing the behavior of ABACOS when binding the same application-level components to two representative communication stacks widely employed in CPSs, DDS and SOME/IP. For each middleware, equivalent *Input* and *Output Port* implementations were developed and bound to the same ABACOS components without modifying their functional behavior, reflecting the intended use of the framework in which interoperability is achieved through the exchange of transport bindings rather than through changes to component logic. This evaluation is intentionally scoped to isolate the incremental cost of the abstraction layer under controlled conditions. The objective is not to exhaustively benchmark the middleware stacks themselves, but to quantify the additional latency introduced by the ABACOS execution model when binding equivalent application logic to distinct communication backends.

Experiments were executed on a computer equipped with an Apple M2 Max processor and 32 GB of memory. The execution environment was configured to reduce external timing variability as far as permitted by the platform. Background activity was minimized, and publisher and subscriber processes were pinned to distinct CPU cores in order to limit scheduling interference and cross-process contention effects. In each experiment, a publisher component periodically transmitted messages of configurable payload size to a subscriber component using a fixed publication period, and identical application logic was employed for both middleware backends to ensure that timing effects were attributable only to the communication stack and to the ABACOS abstractions encapsulating it. No artificial workload was introduced beyond the periodic message exchange, so that the measured latency reflects the intrinsic behavior of the communication stack and the framework abstractions under low-contention conditions. While this configuration does not emulate heavy system load, it enables a controlled comparison in which the incremental overhead of ABACOS can be isolated with minimal confounding factors.

Latency was measured on an end-to-end basis, from the moment a message was released by the publishing component to the moment it was processed by the behavior associated with the corresponding *Input Port* in the subscribing component. To distinguish the contribution of the framework from that of the transport layer, timestamps were additionally collected at the port boundaries, allowing the relative share of the latency attributable to ABACOS, referred to as its overhead, to be evaluated with respect to the total communication latency. The experiments were repeated for multiple payload sizes under a publication period of 100 ms, and results were aggregated over extended execution intervals in order to capture both steady-state behavior and infrequent timing deviations. The selected publication period of 100, ms reflects a common operating regime in distributed perception and coordination tasks in cyber–physical systems. Although higher-frequency control loops (e.g., 1, kHz) may impose tighter timing budgets, the absolute overhead measured in this study can be directly interpreted relative to such periods, as discussed in Section 6.

For the DDS-based experiments, communication was configured using a *Best Effort* reliability policy with a *Keep Last (1)* history QoS profile, reflecting a lightweight configuration commonly adopted in resource-constrained or latency-sensitive deployments. In the SOME/IP-based experiments, message serialization and deserialization were performed within the transport binding, and their execution time was accounted for as part of the middleware latency so that the resulting measurements remained comparable across the two communication technologies. This evaluation setup, therefore, reflects realistic deployment conditions while enabling a fair comparison between middleware stacks bound through the same ABACOS component model. It should be noted that alternative QoS configurations (e.g., reliable delivery in DDS) or additional transport-layer features could increase baseline latency. However, since the objective is to evaluate the incremental framework cost rather than absolute middleware performance, lightweight and commonly adopted configurations were selected to provide a representative yet controlled comparison point.

6 Results

The experimental results from Section 5 are summarized in Figures 3–6, which report both the end-to-end latency of the evaluated middleware stacks and the incremental cost introduced by ABACOS. The boxplots in Figures 5 and 6 compare baseline DDS and SOME/IP end-to-end latencies with their respective end-to-end latencies when integrated in the system using ABACOS across a range of payload sizes. As expected, end-to-end latency increases with payload size for both middleware technologies, reflecting the dominant influence of serialization, transport, and buffer-management costs in the underlying communication stacks.

The contribution of ABACOS to the end-to-end latency is isolated in Figures 3 and 4. In absolute terms, the framework overhead remains tightly bounded across all payload sizes and both communication backends, with median values below approximately 4 μs and upper whiskers within the order of single-digit microseconds. The limited dispersion observed in the upper whiskers suggests that the dominant contribution to latency variability originates from the underlying middleware and operating system scheduler rather than from the ABACOS execution path itself. Since the framework introduces a fixed sequence of operations—event registration, delegate invocation, and run-to-completion execution—its contribution remains structurally bounded and largely insensitive to payload size. Importantly, the absence of any upward trend with increasing payload size indicates that the ABACOS execution path does not introduce payload-dependent copying or buffering effects. This behavior is consistent with the architectural design, in which data ingestion, scheduling, and behavior execution are decoupled without introducing additional transport-level data movement.

From a relative perspective, Figure 4 shows that the proportion of end-to-end latency attributable to ABACOS is highest for small payloads—where communication costs are intrinsically low—and decreases as payload size increases. Median values for small messages remain in the range of 10–17%, while larger payloads reduce the relative contribution to below 10% and, in some cases, to only a few percent. This scaling trend reinforces the interpretation that, for large payloads, the dominant timing contribution originates from the middleware transport layer, whereas the ABACOS scheduling and port abstractions contribute a small, payload-independent constant cost. To contextualize these values, an absolute overhead

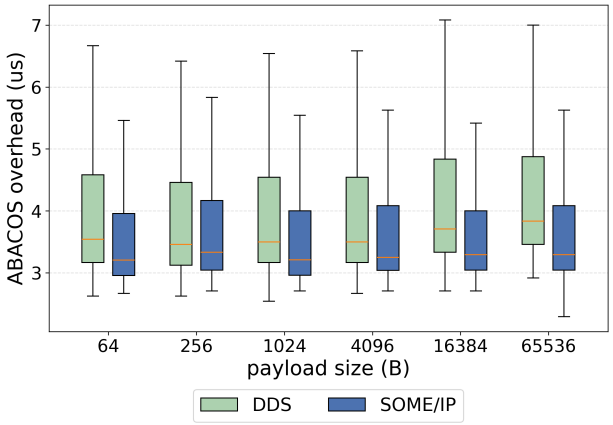

Figure 3: ABACOS overhead as a function of payload size

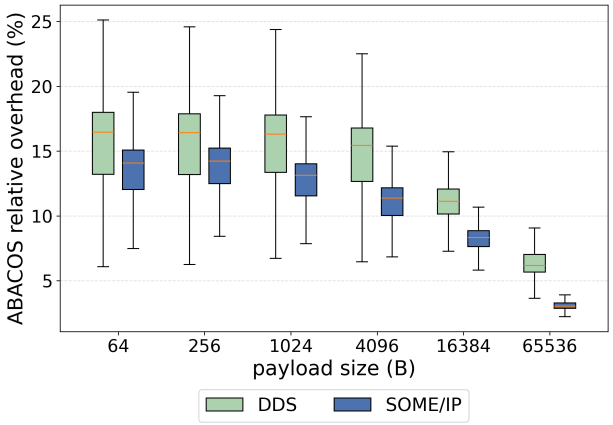

Figure 4: ABACOS relative overhead as a function of payload size

below 10μs corresponds to approximately 1% of a 1ms period for a high frequency task and less than 0.01% of a 100ms period of a typical task. For many distributed perception, planning, or supervisory-control workloads typical of modern CPSs [15], such a contribution is negligible relative to network and serialization delays.

The present evaluation focuses on single-hop communication between two components under controlled execution conditions. It does not address large-scale deployments with many interacting components, high-contention workloads, or mixed-criticality scheduling environments. Furthermore, experiments were conducted on a general-purpose operating system rather than a real-time kernel, and therefore do not constitute a worst-case execution time analysis. These aspects remain important directions for future empirical investigation.

Taken together, these results indicate that interoperability through ABACOS can be achieved without introducing payload-dependent or unbounded additional latency beyond that inherent to the underlying middleware. The overhead introduced by the framework is both bounded and stable, and remains significantly smaller than the intrinsic variability of the communication stacks themselves. This provides experimental evidence that the abstraction boundaries introduced by the ABACOS component, port, and behavior model do not impose additional communication-path penalties beyond those already present in middleware implementations.

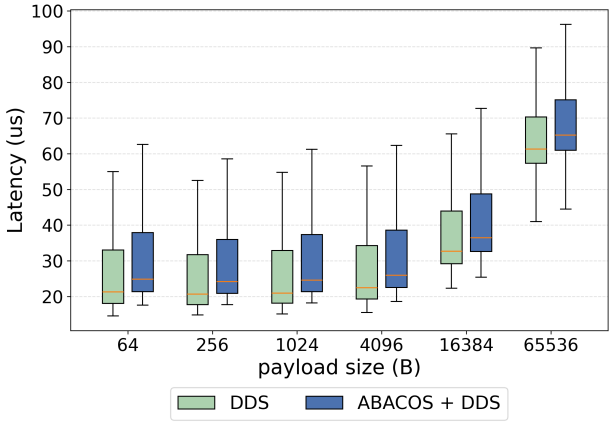

Figure 5: DDS and ABACOS with DDS end-to-end latency

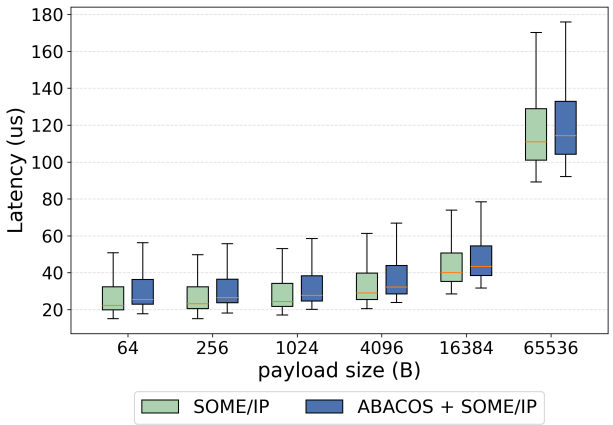

Figure 6: SOME/IP and ABACOS with SOME/IP end-to-end latency

7 Conclusion

This work presented ABACOS, a framework for the design and implementation of software components for cyber–physical systems based on a minimal set of abstractions that separate communication, scheduling, and functional execution concerns. By modeling components in terms of *Behaviors*, *Input and Output Ports*, and a run-to-completion execution model, ABACOS enables the binding of heterogeneous communication mechanisms without modifying component logic, thereby supporting interoperability as a configurable system property rather than a design constraint.

A C++ implementation of the framework was developed and evaluated experimentally using two widely deployed middleware

technologies, DDS and SOME/IP. The results demonstrate that the framework introduces a small and largely payload-independent constant overhead, while preserving the timing characteristics of the underlying communication stack. These findings provide empirical evidence that the abstractions proposed by ABACOS do not impose additional communication-path penalties beyond those inherent to the transport mechanisms themselves, and therefore constitute a viable foundation for middleware-agnostic component execution in CPSs.

While the current evaluation demonstrates bounded and stable overhead under controlled conditions, further studies are required to characterize scalability, high-load behavior, and integration with real-time operating systems. Nonetheless, the experimental evidence indicates that the abstraction boundaries introduced by ABACOS do not fundamentally compromise timing efficiency.

Future work will extend the framework toward richer execution semantics, including multi-threaded components and alternative event-scheduling strategies, as well as broader interoperability scenarios involving mixed-criticality platforms and embedded real-time operating systems. Additional studies will also examine large-scale deployment and compositional timing analysis, with the objective of further strengthening the role of abstraction-driven software architectures in the engineering of dependable, cross-middleware cyber–physical systems.

References

[1] Andrei Alexandrescu. 2001. *Modern C++ design: generic programming and design patterns applied.* Addison-Wesley.

[2] Noriaki Ando, Takashi Suehiro, and Tetsuo Kotoku. 2008. A software platform for component based rt-system development: Openrtm-aist. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots*. Springer, 87–98.

[3] Danilo Beuche, Antônio Augusto Fröhlich, Reinhard Meyer, Holger Papajewski, Friedrich Schön, Wolfgang Schröder-Preikschat, Olaf Spinczyk, and Ute Spinczyk. 2000. On architecture transparency in operating systems. In *Proceedings of the 9th Workshop on ACM SIGOPS European Workshop: Beyond the PC: New Challenges for the Operating system*. 147–152.

[4] Herman Bruyninckx. 2001. Open robot control software: the OROCOS project. In *Proceedings 2001 ICRA. IEEE international conference on robotics and automation (Cat. No. 01CH37164)*, Vol. 3. IEEE, 2523–2528.

[5] Saeid Dehnavi, Martijn Koedam, Andrew Nelson, Dip Goswami, and Kees Goossens. 2021. CompROS: A composable ROS2 based architecture for real-time embedded robotic development. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 6449–6455.

[6] Dongwon Hong and Changjoo Moon. 2024. Autonomous driving system architecture with integrated ROS2 and adaptive AUTOSAR. *Electronics* 13, 7 (2024), 1303.

[7] Xu Jiang, Dong Ji, Nan Guan, Ruoxiang Li, Yue Tang, and Yi Wang. 2022. Realtime scheduling and analysis of processing chains on multi-threaded executor in ros 2. In 2022 IEEE Real-Time Systems Symposium (RTSS). *IEEE, 27539* (2022).

[8] Alexandru Kampmann, Andreas Wüstenberg, Bassam Alrifaee, and Stefan Kowalewski. 2019. A portable implementation of the real-time publish-subscribe protocol for microcontrollers in distributed robotic applications. In *2019 IEEE intelligent transportation systems conference (ITSC)*. IEEE, 443–448.

[9] Shinpei Kato, Shota Tokunaga, Yuya Maruyama, Seiya Maeda, Manato Hirabayashi, Yuki Kitsukawa, Abraham Monrroy, Tomohito Ando, Yusuke Fujii, and Takuya Azumi. 2018. Autoware on board: Enabling autonomous vehicles with embedded systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 287–296.

[10] Suhong Kim, Hyeongju Choi, Suhaeng Lee, Minseo Kim, Hyunseo Shin, and Changjoo Moon. 2025. A Dynamic Bridge Architecture for Efficient Interoperability Between AUTOSAR Adaptive and ROS2. *Electronics* 14, 18 (2025), 3635.

[11] Hermann Kopetz. 1997. *Real-time systems: design principles for distributed embedded applications*. Springer.

[12] Tobias Kronauer, Joshwa Pohlmann, Maximilian Matthe, Till Smejkal, and Gerhard Fettweis. 2021. Latency overhead of ros2 for modular time-critical systems. *arXiv preprint arXiv:2101.02074* (2021).

[13] Arturo Laurenzi, Davide Antonucci, Nikos G Tsagarakis, and Luca Muratore. 2023. The xbot2 real-time middleware for robotics. *Robotics and Autonomous Systems* 163 (2023), 104379.

[14] Edward A. Lee. 2008. Cyber Physical Systems: Design Challenges. In *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. 363–369. doi:10.1109/ISORC.2008.25

[15] Hyoeun Lee, Youngjoon Choi, Taeho Han, and Kanghee Kim. 2022. Probabilistically guaranteeing end-to-end latencies in autonomous vehicle computing systems. *IEEE Trans. Comput.* 71, 12 (2022), 3361–3374.

[16] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. 2022. Robot Operating System 2: Design, architecture, and uses in the wild. *Science Robotics* 7, 66 (2022), eabm6074. doi:10.1126/scirobotics.abm6074

[17] Christian Menard, Andrés Goens, Marten Lohstroh, and Jeronimo Castrillon. 2019. Achieving determinism in adaptive AUTOSAR. *arXiv preprint arXiv:1912.01367* (2019).

[18] Sangyoon Oh, Jai-Hoon Kim, and Geoffrey Fox. 2010. Real-time performance analysis for publish/subscribe systems. *Future Generation Computer Systems* 26, 3 (2010), 318–323.

[19] Navrattan Parmar, Virender Ranga, and B Simhachalam Naidu. 2020. Syntactic interoperability in real-time systems, ros 2, and adaptive autosar using data distribution services: An approach. In *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2019*. Springer, 257–274.

[20] Ragunathan Rajkumar, Insup Lee, Lui Sha, and John Stankovic. 2010. Cyber-physical systems: The next computing revolution. In *Design Automation Conference*. 731–736. doi:10.1145/1837274.1837461

[21] Baidu Apollo team. 2017. Apollo: Open Source Autonomous Driving. https://github.com/ApolloAuto/apollo. Accessed: 2019-02-11.

[22] Yanlei Ye, Zhenguo Nie, Xinjun Liu, Fugui Xie, Zihao Li, and Peng Li. 2023. Ros2 real-time performance optimization and evaluation. *Chinese Journal of Mechanical Engineering* 36, 1 (2023), 144.

Observability: The Missing Piece of Management in NFV-based Network Environments

Guilherme Werneck de Oliveira
guilherme.oliveira@ifpr.edu.br
Federal Institute of Paraná
Pinhais, Paraná, Brazil

Elias P. Duarte Jr., Vinicius Fulber-Garcia
{elias,vinicius}@inf.ufpr.br
Federal University of Paraná
Curitiba, Paraná, Brazil

Abstract

Networks are dynamic entities, as the demands on their resources, functions, and services vary over time, leading to continuous changes in their state. In recent years, the Network Function Virtualization (NFV) paradigm has emerged as a practical approach to address this dynamic nature, offering high flexibility, elasticity, and mobility for managing network functions and services. To fully leverage this flexibility, it is essential to observe the network environment by monitoring metrics that reflect its state, enabling the inference of complex indicators and trends. Despite its importance, observability in NFV-based networks remains underexplored in the literature. Existing works often lack discussions that explicitly integrate observability concepts into the NFV reference architecture's working domains and operational elements, as well as analyses of the practical implications of enforcing them. This paper presents an investigation of the application of observability in the context of NFV, linking the fundamentals of observability with the latest NFV architectures and technologies. It also demonstrates, through a case study on detecting network state changes caused by malicious activity, the impact of adopting different observability measurement models. Our results show that distinct measurement models present varying overheads on network traffic and network function instances, underscoring the importance of a well-planned integration of observability in NFV-based networks.

CCS Concepts

• **Networks** → **Network measurement**; **Network management**; **Network monitoring**;

Keywords

NFV, Observability, Metrics, Measurements, Monitoring, Management

1 Introduction

Computer networks have become increasingly complex, requiring the implementation and deployment of innovative and sophisticated functions and services to support them. In this context, adaptive networks, which can be reconfigured in response to changes in network state, have emerged as a critical technology [15, 53]. To address the challenges of flexibility, elasticity, and mobility in adaptive networks, softwarization paradigms such as Software-Defined Networking (SDN) and Network Function Virtualization (NFV) have been extensively explored by both academia and industry [20]. In particular, the NFV paradigm has received special attention due to its ability to deploy Virtualized Network Functions (VNF) and orchestrate them through Service Function Chains (SFC) in a highly adaptable manner [13].

Although NFV was developed to provide high flexibility for network environments, leveraging all this potential is not trivial. First of all, it requires effective management of virtual functions and services [16, 18]. Network management must effectively provide a deep understanding of the network state. This challenge becomes even more complex when the network is regarded as a dynamic entity whose state changes through the interactions of its connected nodes [12, 34]. Therefore, we argue that NFV can significantly benefit from observability, which has been successfully applied to support strategic management and decision-making across multiple scenarios [33, 46].

Observability can be defined as the extent to which a system's state can be traced and understood by an observer entity [35]. To simplify observability in NFV, it can be seen as the monitoring of metrics, through measurements that characterize network functions, services, and the overall network state. However, although a vast literature exists on solutions to manage NFV-based networks, keeping them operational and optimized [14, 21, 41, 42], there are few works on metrics and measurement models to provide such solutions with appropriate data aligned to their objectives. Furthermore, there are few insights into where these metrics should be defined and how they can be made available for measurement by observer entities within the NFV reference architecture. At last, every operation in the network incurs a cost, including observing virtualized functions, services, and the network itself. Thus, discussing the costs and consequences of adopting different observability measurement models represents another gap in the literature.

Accordingly, this article discusses observability as a key requirement for managing NFV-based network environments. However, implementing observability requires support from multiple operational elements defined in the NFV reference architecture, which may act as observed entities, observer entities, or both depending on the case. Moreover, adopting different observability measurement models can generate heterogeneous impacts on defining the network state and shape decision-making in management. To discuss these impacts, we conducted an empirical evaluation simulating a scenario in which the network faces malicious traffic, with observability enabling the detection of an ongoing attack. The results revealed distinct operational characteristics and collateral effects for each tested measurement model, indicating that choosing one model over another may lead to overhead on the network traffic and network function instances.

The remainder of this work is organized as follows. Section 2 presents the fundamentals of observability. Section 3 identifies observability enablers within the NFV reference architecture and describes them as observed or observer entities. Section 4 discusses metrics and measurement models, highlighting implementation

opportunities based on state-of-the-art NFV architectures and technologies. Section 5 demonstrates the impact of adopting different observability measurement models in NFV-based networks through a case study. Finally, Section 6 concludes the paper.

2 Observability in a Nutshell

The concept of observability was introduced in 1960 within the framework of control theory to formalize a mathematical approach for inferring a system's internal state from its external outputs [30]. Initially, observability was applied to the study of formal methods and linear algebra, particularly in the analysis of dynamic systems in mechanical and automation engineering.

In networking, observability enables an agent (**observer entity**) to compute metrics and generate indicators from network functions, services, and links (**observed entities**). These metrics and indicators can then be analyzed by management and provisioning elements to assess the evolution of the network state and to support decision-making processes, with the goal of, for example, meeting performance requirements, fulfilling service level agreements, and improving both Quality of Service (QoS) and Quality of Experience (QoE) [10, 38].

Technically, observability can be analyzed from two primary perspectives: **metrics** and **measurements**. Metrics are standardized definitions of values computed in an experiment designed for specific performance-related purposes [25]. Examples include the state of an operational element in terms of CPU and RAM usage, as well as the state of links in terms of latency, throughput, etc. By analyzing these metrics, it is possible to assess the functionality of each operational element and its connections, thereby deriving the network's abstract state. Measurements, on the other hand, refer to the process of executing a series of operations to get the value of a metric within a specific scenario and time frame.

It is important to note that observability extends beyond mere monitoring by providing a holistic view of the network. Observability defines which metrics should be measured and how they should be collected, integrating multiple data sources to provide complex indicators and actionable insights, such as proactive issue detection and resolution. In contrast, monitoring consists of evaluating the state of a network and its functions and services by comparing them against an expected state. Observability does not rely on predefined expected states; instead, it treats the network as a dynamic entity and aims to deeply analyze how its states evolve and what valuable information and actions can be derived from these dynamics. It includes assessing the potential interference introduced by the observation process itself and adapting it to obtain the most appropriate information from the network, balancing freshness, accuracy, and costs.

3 Observability in NFV: A Map

The NFV reference architecture, proposed by the European Telecommunications Standards Institute (ETSI) [8], comprises three working domains: Virtualized Infrastructure (VI), Management and Orchestration (MANO), and Virtualized Network Functions (VNF). The VI domain is responsible for managing the underlying infrastructure that provides the computational resources required to support the deployment and execution of virtualized functions and services.

The MANO domain provides management and orchestration of the NFV environment through three main operational elements: the Virtualized Infrastructure Manager (VIM), which communicates with the VI domain to allocate, release, and monitor computational resources; the VNF Manager (VNFM), which interacts with elements in the VNF domain to manage the lifecycle of individual virtualized function instances; and the NFV Orchestrator (NFVO), which is responsible for managing the lifecycle of complete network services, typically composed of multiple interconnected functions.

Finally, the VNF domain consists of two operational elements: the Virtualized Network Functions (VNFs) themselves, which represent the primary traffic-processing elements by executing network functions over the traffic; and the Element Management System (EMS), a management element that interacts directly with VNF instances, abstracting their complexity and heterogeneity from the MANO.

Some works in the literature address specific aspects of defining which working domains or operational elements of the NFV reference architecture assume particular observability roles in a network. In [4], the authors propose a mechanism for trust assessment in NFV-based networks. Their solution integrates a trust monitor into the MANO working domain, in accordance with ETSI guidelines on NFV trust and security. The architectural discussion underpinning this decision reinforces MANO's role as the central actor in observability, monitoring, and decision-making within the NFV reference architecture, as it manages and orchestrates both the VNF and VI domains (observed entities). Accordingly, MANO and its operational elements can be considered top-level observer entities within the NFV reference architecture.

In [29], the concept of a target entity for NFV observability is introduced, which refers to the elements that must be considered to compute a given metric. For example, when measuring the CPU load, a specific VNF instance serves as both the target and the observed entity. However, this model highlights an important point: computing a single metric may involve multiple target entities. For instance, when measuring the Round-Trip Time (RTT) from a VNF X to a VNF Y, the observed entity is the VNF X (since RTT is not necessarily symmetric). In contrast, both VNF instances (X and Y) are required to compute the metric and are therefore considered target entities. Thus, based on this work, VNF instances are assumed to be the primary observed entities in an NFV environment.

Enabling observer entities within MANO to request measurements and access information from observed entities requires the establishment of appropriate communication interfaces. The ETSI NFV reference architecture defines several interfaces that can be used for this purpose, including the Nf-Vi interface between MANO (VIM) and the VI domain; the Ve-Vnfm-vnf interface between MANO (VNFM) and VNF instances; and the Ve-Vnfm-em interface between MANO and EMS instances. These interfaces, along with their associated communication protocols, must be implemented within each working domain and its operational elements. However, in practice, there is limited standardization regarding the set of supported operations and the technologies used to implement them.

The literature, however, provides architectures and frameworks to enable holistic communication among different implementations

of interfaces provided by observer and observed entities. For instance, the framework presented in [22] enables different orchestrators to communicate to deploy and manage the network functions of a single service, including an abstract implementation of the Nf-Vi interface. Furthermore, the architecture for implementing VNF platforms described in [17] introduces an internal module, referred to as a management agent, which is specifically designed to establish the Ve-Vnfm-vnf interface.

In this sense, the work in [11] presents an architecture for the EMS module. According to the proposed internal organization, the EMS acts as an intermediary via the Ve-Vnfm-em interface, abstracting the heterogeneity involved in accessing VNF instances from the VNFM. In this architecture, the EMS is responsible for directly interacting with VNF instances, including observing them, and thus acts as an observer entity. At the same time, the EMS acts as an observed entity with respect to the VNFM (MANO), as it receives and responds to measurement requests for VNF instances. To support this, the architecture defines three main modules: the VNF subsystem, which establishes communication between the EMS and the VNF instances; the monitoring subsystem, which enables operators to define proactive monitoring routines for VNF instances; and the access subsystem, which implements the Ve-Vnfm-em interface.

By leveraging the observability enablers provided by the NFV paradigm, it is possible to define multiple metrics and heterogeneous measurement models that trigger processes to compute and communicate them. Naturally, an NFV-based network may present a diversity of requirements, making specific metrics and measurement models more suitable for certain scenarios.

4 Metrics and Measurements Tailored to the NFV Paradigm

Observability has the potential to play a critical role in managing NFV-based networks, providing the visibility needed to act on virtualized elements in real time. In modern networks, where virtualized functions and services are deployed on demand to meet varying service requirements, observability enables operators to continuously assess their health, performance, and resource utilization. Furthermore, NFV technology has been also used to deploy arbitrary functions and services in the network [51], multiple services have been built according with this paradigm [9, 44, 49, 50]. Real-time observability enables management and orchestration platforms [23, 43], for example, to automatically scale VNF instances up or down, in or out, based on current demand or to migrate them across different virtualization infrastructures.

However, NFV-based networks are highly dependent on the underlying virtual infrastructure, and their performance can be affected by multiple factors, including resource contention, the state of network links, and the way metrics are measured. Consequently, different metrics and measurement models must be defined and analyzed from the perspective of the NFV paradigm in order to identify the most suitable observability approach for each specific network context.

4.1 Observing NFV-based Networks: Metrics

While a myriad of metrics are commonly used in daily network operations, their definitions are far from straightforward. The design of metrics must ensure they are meaningful, measurable, and interoperable across diverse network environments, enabling consistent measurement and comparison. RFC 6390 [25] outlines a process for designing metrics, which includes the following steps: *i)* problem statement, clearly defining the problem that the new metric aims to address; *ii)* metric definition, creating a precise and unambiguous definition of the metric, including what is being measured and the units of measurement; *iii)* method of measurement, specifying how the metric will be measured, including any relevant algorithms or methodologies; and *iv)* reporting and interpretation, providing guidelines for how the results should be reported and interpreted, ensuring clarity and consistency.

In the context of virtualized network services, the ETSI defines a comprehensive set of metrics related to virtualized network functions and the management of virtualized infrastructures [6, 7]. These metrics address relevant aspects of virtualization-enabled networks, including efficiency (*e.g.*, packet delay, jitter, throughput of delivered packets, interruption, provisioning, and configuration latency), efficacy (*e.g.*, packet loss rate, clock error, placement, and compliance policy), and reliability (*e.g.*, provisioning reliability, broken connections, premature releases, and failed release rates).

The Internet Engineering Task Force (IETF), in turn, has expanded the scope of the Benchmarking Methodology Working Group (BMWG) to include methods for evaluating virtualized network functions and their supporting technologies, such as SDN controllers and virtual switches. The BMWG has developed a specific methodology for benchmarking virtual network performance [26], which includes key metrics such as throughput, packet loss rate, latency, and CPU and RAM consumption.

Other metrics related to NFV performance are outlined in RFC 8172 [27], including the time required to deploy and migrate virtual network functions. This RFC also offers insights into the impact of measurement policies on VNF performance, highlighting that different measurement policies can have heterogeneous effects on VNF performance when processing network traffic.

The academic literature also presents works that highlight metrics related to NFV. In [31], for instance, the authors present performance comparison metrics for SDN and NFV controllers. NFV metrics include vCPU and vMemory (compute-associated), latency and throughput (communication-associated), I/O rate, and VNF recovery time (storage-associated). The work of [19] illustrates the applicability of metrics specific to NFV fault tolerance, such as packet loss, average packet throughput, congestion interconnection, among others. A deeper study of the impacts of environmental metrics on Machine Learning (ML) models used for management is presented in [32] and [5]. The work [32] presents a supervised learning study to predict VNF deployment decisions under changing network conditions. In [5], the authors focus on VNF placement to mitigate DDoS attacks in industrial IoT systems. The observed metrics included environment deployment time, CPU and RAM consumption, latency, throughput, time to attack detection and mitigation, response time, and ML model accuracy.

Table 1 presents a classification of NFV metrics based on the works presented in this section. Three categories of metrics are considered: *i)* metrics for diagnosing the state of virtualized functions, which depend on the implementation of the network function; *ii)* metrics related to the state of virtualized nodes (v-nodes); and *iii)* metrics concerning the overall network state.

Table 1: Categories of metrics for NFV environments.

Metric		
VNF	**V-Node**	**Network**
Setup and response time	CPU, RAM and disk/swap consumption	Latency
Number of rules executed/not-executed	Disk and network I/O	Throughput
Policy enforcement time	Deployment, migration and recovery time	Jitter
Machine learning models metrics	Scalability rate	Delivered/discarded packets
Packet processing time		
Packet and data receiving/transmitting rate		

Finally, it is important to note that, in addition to conventional metrics, other resources can be instrumental for observing the state of applications and services in NFV provider environments. These resources include [35]: *i)* logs, which provide a detailed, timestamped, and immutable record of events; and *ii)* traces, which capture end-to-end temporal events related to specific actions in distributed systems, such as the management of virtualized network services. Notably, tracing is also crucial for addressing the challenge of explainability (understanding how algorithms build their results and make decisions) in dynamic networks.

4.2 Observing NFV-based Networks: Measurements

After designing and implementing metrics in an NFV-based network environment, the next step toward achieving observability is to measure them across network elements and links. Thereby, these measurements must be conducted in a way that does not overload the network or its elements, ensuring that regular traffic processing remains unaffected. For a considerable time, network operators have relied on protocols and technologies such as the Simple Network Management Protocol (SNMP), Command-Line Interface (CLI), or Syslog for network monitoring. However, dynamic network operations require data to be measured from multiple sources with varying granularity and frequencies, a need that legacy protocols and technologies often struggle to meet. Even with advances such as NETCONF [24], gRPC collectors [39], perfSONAR [40], and In-band Network Telemetry (INT) [37], there remain limitations in collecting metrics for v-nodes and VNF instances.

Upon analyzing the literature on observability, it is possible to identify the primary characteristics of measurement models suitable for network environments. Authors in [52] propose an abstract framework for implementing general entities as observable sources of knowledge and services, introducing two models regarding the **dissemination** of measured metrics: an observer entity can get information from an observed entity using a reactive or proactive approach. When an observer entity sends a state request, and the observed entity processes it as soon as it arrives, the dissemination model is called reactive. However, the observed entity may or may not respond immediately. For example, if a response has the same value as the previous one, the observed entity may decide to spare a retransmission. This technique is called ”conditioned responses´´, where replies are sent only when specific rules are met in the observed entity or, generally, when a relevant event occurs.

Unlike the reactive model, the proactive dissemination involves the observed entity notifying the observer entity, which can be implemented using a publish/subscribe scheme [3]. In this case, the observer entity sends a request to register its interest in receiving notifications whenever the value of a specific metric or state from the observed entity changes. This subscription can be carried out directly with the observed party or through an intermediary, such as a message broker. Moreover, to manage the number of requests and responses generated in the network environment, the previously mentioned technique of conditioned responses can also be integrated into this model.

Furthermore, the **encapsulation** of measured metrics for transmission from the observed entity to the observer also defines two models: active and passive. According to [47] and [1], the passive model does not introduce additional traffic into the network. In contrast, the active model involves the transmission of dedicated packets to provide information about network entities, such as links and devices; as a result, it is often intrusive and may disrupt ongoing services due to the additional traffic it generates.

It is important to note that passive and active measurement models can be applied both to reactive dissemination, through reactive responses [52] or query-based polling [28], and proactive dissemination, through proactive reporting [52] or subscription-based pushing [28]. The classification ultimately depends on how the communication of the measurement process is structured. For instance, using dedicated packets to transfer measurement results characterizes an active model, whereas techniques such as piggybacking measurement data onto existing traffic correspond to a passive model.

Another characteristic to consider regarding measurements is the **acquisition** mode, which determines whether measurements are triggered continuously or on demand. Continuous acquisition involves constant monitoring of environmental metrics over time [45], enabling real-time visibility and trend analysis. The continuous model enables early identification of patterns, anomalies, and performance degradation, making it valuable for proactive network management and long-term optimization. In contrast, on-demand acquisitions are triggered by an entity when specific information about the network’s current state is required, providing flexibility and targeted insights without constantly collecting data.

Moreover, the **interval** between measurements is also relevant, since it may affect not only the network but also the observed entities, such as VNF instances [25, 27]. Short sampling intervals offer valuable insights into an entity's performance. However, they can result in an overwhelming number of measurements and an excessive processing load concurrent with the primary function of the observed entity. On the other hand, longer sampling intervals reduce the volume of data collected, which may be insufficient to accurately assess the observed entity performance or the network state, as the metrics being measured might fluctuate significantly over time. In some cases, the percentile-based statistical technique may be used to disregard potential outliers. Table 2 summarizes the described measurement characteristics and models presented in this section.

Table 2: Observability measurement: characteristics and models.

Characteristic	Models	Description
Dissemination	Reactive Proactive	Refers to the strategy by which measurements are requested and delivered from observed to observer entities, either reactively upon request or proactively based on conditions or events.
Encapsulation	Active Passive	Refers to the strategy used to transfer measurements from observed to observer entities, either actively through dedicated packets or passively by, for example, piggybacking on existing traffic.
Acquisition	Continuous On-Demand	Refers to the strategy by which measurements are triggered to define or update metric values, either continuously through constant and regular monitoring or on demand in response to occasional requests.
Interval	Short Long	Refers to the sampling interval strategy that defines how frequently metrics are measured, either short to improve information freshness or long to reduce observability overhead.

From a technical implementation perspective in NFV, there is a strong synergy between observability measurement models and state-of-the-art architectures designed for the operational elements of the paradigm. For example, the VNF platform architecture presented in [17] is prepared to enable reactive dissemination in an active measurement mode through its management agent modules, which create the interface to communicate with EMS and VNFM (the observer entities), and extended agents, which provide specific metrics related to the network function executed within the VNF platform. Notably, there is no restriction on the measurement process for either the acquisition or the interval models. Consequently, the observer entity is fully responsible for determining both the measurement frequency and the strategy for executing requests. Partially or entirely, the VNF platforms ClickOS [36], Click-on-OSv [2], and COVEN [17] follow this architectural approach, supporting the measurement models as described.

The EMS architecture presented in [11], in turn, can leverage most measurement models, as it operates as both an observed and an observer entity. Through its internal modules, the EMS supports reactive measurements via the ETSI Ve-Vnfm-em standard interface implemented in the access subsystem, as well as proactive measurements through a subscription mechanism associated with the monitors of the monitoring subsystem. In this sense, the access subsystem enables top-level observer entities to submit on-demand measurement requests. In contrast, the monitoring subsystem allows the EMS to measure metrics from VNF instances either continuously or on demand, at both short and long intervals. From an architectural perspective, there is no explicit support for the EMS operating in a passive mode with respect to packet encapsulation, with the active model being the most straightforward alternative, as implemented in the HoLMES EMS [11]. Nevertheless, a passive model could also be realized in EMS-based solutions, for example, by piggybacking measurement results onto packets used to confirm the execution of lifecycle operations in VNF instances requested by the VNFM or by operations and business support systems.

Furthermore, top-level observer entities, such as the VNFM, must be capable of triggering measurements and receiving and processing their results, regardless of the measurement models adopted. The ETSI specification provides only a high-level definition of the VNFM and does not define an internal architecture for implementing this operational element within the MANO domain. Nevertheless, the literature offers insights into possible implementations, including the definition of internal modules that enable refined observability operations through the VNFM. For example, the authors in [48] propose a framework to address technical gaps in the VNFM that are not covered by its specification documents. They introduce a Monitoring Module composed of two agents: one responsible for monitoring VNF instances, either directly via the Ve-Vnfm-vnf interface or indirectly through the EMS via the Ve-Vnfm-em interface, and another responsible for monitoring the underlying virtualization infrastructure through the Nf-Vi interface. These agents can execute different models for acquisition and interval, and play a central role in lifecycle management decision-making in an NFV environment.

Another important factor is that observability can have significant implications for both infrastructure security and user privacy. On the one hand, observability enhances threat detection, incident response, and forensic analysis by enabling fine-grained monitoring of anomalous behavior across distributed systems, such as lateral movement within virtualized clusters or suspicious API calls in workloads. Thus automatic and autonomous systems can leverage real-time observations to detect misconfigurations, privilege escalation attempts, or data exfiltration patterns, thereby strengthening resilience and compliance. On the other hand, the same data collection mechanisms that improve visibility may also expand the attack surface and introduce privacy risks. Extensive logging of user interactions, session identifiers, IP addresses, or payload data, particularly without adequate access controls, encryption, or data minimization practices, can inadvertently expose sensitive

personal information. In addition, centralized observability platforms may become high-value targets, as aggregated observations can reveal system architecture details and user behavior patterns. Consequently, while observability is essential for securing modern virtualized infrastructures, it requires rigorous governance frameworks, strict access policies, and privacy-by-design principles to prevent the transformation of monitoring mechanisms into vectors of surveillance or compromise.

Regardless of whether measurement is proactive or reactive, passive or active, continuous or on-demand, and whether it is performed over short or long intervals, measurement techniques may consume significant network resources and generate redundant or sensitive data. Therefore, the enforcement of observability must carefully consider the trade-offs associated with different metrics and measurement models, taking into account network requirements and capacity constraints. This approach enables the tracing of functions, services, and overall network state with an appropriate level of detail, aligned with the objectives of users and operators.

5 The Impacts of Observability: A Case Study

The level of insight derived from metric measurements is an important factor in planning observability in NFV-based networks, since VNF instances are highly dependent on the underlying virtual infrastructure and their performance can be influenced by several factors, such as resource contention, the state of network links, and even the way in which measurements are executed. To demonstrate the impacts of adopting different observability measurement models, the case study illustrated in Figure 1 is presented. The experimental setup consists of a simulated NFV network implemented with Docker containers, running applications written in C. We consider three main containers in the experiment:

- **VNF**: A container that runs an application whose primary functionality is to receive packets from a client and analyze their payloads, searching for malicious signatures in the content, thus simulating a virtual Deep Packet Inspection (vDPI) function. For each detected signature, an attack counter is incremented; this counter is reset upon communication of the measured value resulting from a requested or executed measurement. The VNF provides two types of interfaces. The first is a server-side network socket that waits for requests and responds with the counter value (**reactive dissemination model**). The second is a client-side network socket that proactively forwards signature counter values to the observer application without explicit requests (**proactive dissemination model**);
- **Client**: A container running an application that sends a large volume of predefined packets, which may or may not contain attack signatures identified by the vDPI VNF;
- **Observer**: A container with an application whose purpose is to request metric measurements from the vDPI VNF (reactive dissemination) or to receive metric values automatically (proactive dissemination) from the VNF instance.

In this experiment, all metric measurement requests are responded to using dedicated packets; thus, observability measurements are implemented using an **active encapsulation model**. Moreover, all measurements adopt the same **acquisition model (continuous)** and **interval model (adaptable)**, according to the following policy: after each measurement, the VNF instance (in the proactive case) or the observer (in the reactive case) increases the measurement interval by two (2) seconds when no malicious signatures are identified, or decreases it by two (2) seconds when one or more malicious signatures are detected. The interval is bounded by a minimum of zero (0) seconds and a maximum of eleven (11) seconds. Furthermore, in the proactive dissemination model, the measurement agent running within the VNF platform may not initiate communication with the observer at the end of a measurement interval if no attack activity has been detected during that period (*i.e.*, when the signature counter equals zero).

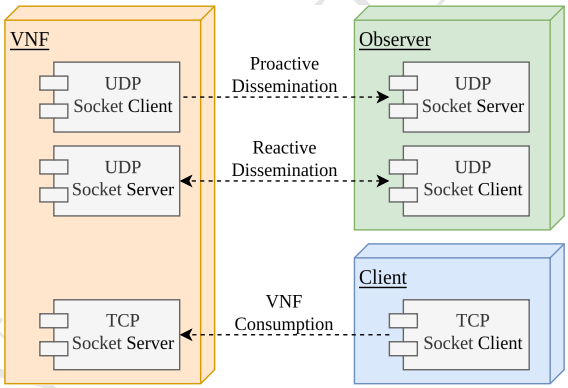

Figure 1: Experimental setup for the case study.

Furthermore, additional metric values are transmitted along with the signature counter with the measurement results, including CPU and RAM utilization percentages, as well as the number of packets and the volume of data received and transmitted per second (rxpck/s, txpck/s, rxkB/s, and txkB/s). These metrics are obtained using the `sysstat` tool[1].

After establishing the experimental setup, we conducted tests using three sets of messages, each containing 10%, 40%, and 90% of the packets with attack signatures (Cases 1, 2, and 3), for a total of 500 million packets per case. To simulate a burst attack, the packets containing the respective signatures were distributed across three equal-sized traffic windows. Each case was executed 30 times to assess the impact of adopting proactive or reactive measurement models on network load and the detection of malicious activity. The developed applications, scripts, and raw data and results are available in a public GitHub repository[2].

The number of packets exchanged between the vDPI VNF instance and the observer was determined for the three testing scenarios. As shown in Figure 2, the reactive measurement model generated approximately 2.25 times more observations than the proactive measurement model, with this difference decreasing as the number of identified malicious packets increased. This behavior occurs because the reactive model always requires a request from the observer to the VNF to trigger a measurement and, subsequently, to update the measurement interval according to the defined policy.

[1]https://github.com/sysstat/sysstat
[2]https://github.com/werneckg/vnf-observability

Since there are no synchronized clocks or fault-detection mechanisms in the system, each request issued by the observer must receive a response, even when the signature counter is zero. In contrast, under the proactive model, the VNF does not communicate with the observer when the attack counter is zero; it only adapts its measurement interval. This distinction is also evident in Figure 3, which presents the cumulative number of requests and responses generated by each approach based on their respective observations.

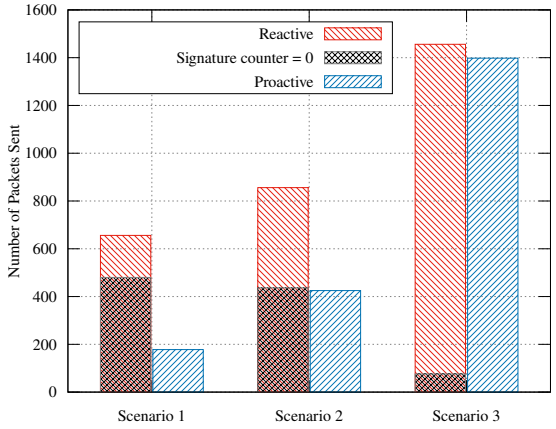

Figure 2: Comparison between reactive and proactive dissemination models according to the number of metric measurements sent from the VNF to the observer.

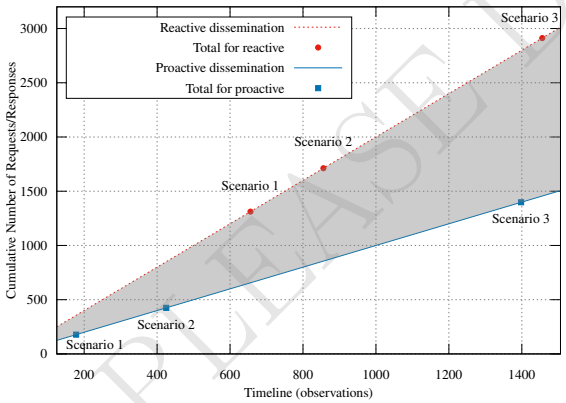

Figure 3: Cumulative number of measurement results sent from the VNF instance to the observer along the three scenarios.

Regarding the performance metrics measured and sent by the VNF, along with the signature counter, a clear difference is observed in the average number of packets per second received and transmitted under the reactive and proactive dissemination models, as shown in Figure 4. The 90th percentile confirms this difference, also shown in Figure 4. Specifically, the proactive model consistently results in fewer received and transmitted packets. This behavior stems from two features of the proactive dissemination model implemented in this case study: it does not require network requests to trigger the measurement process, and it does not transmit measurement packets when the signature counter is zero. In contrast, the reactive model always involves request–response exchanges, resulting in a slightly higher network load.

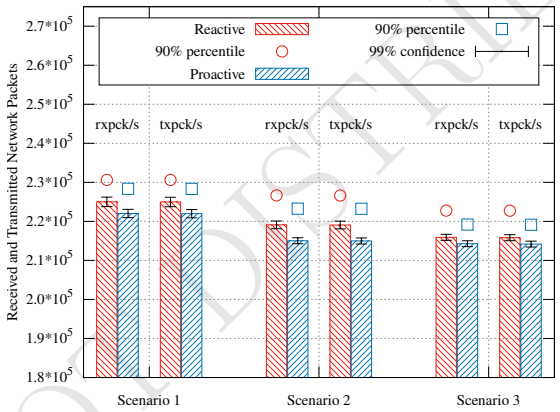

Figure 4: Received and transmitted packet rate from the VNF perspective.

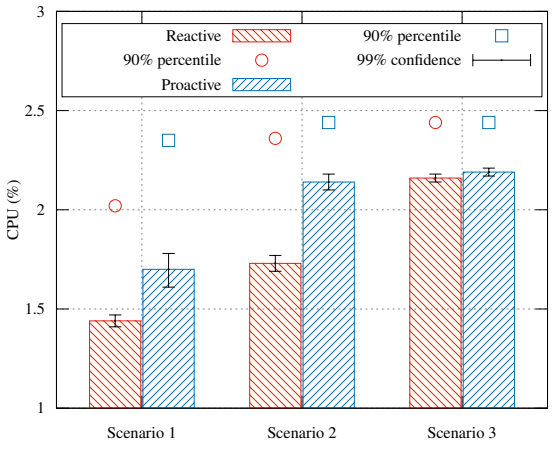

Figure 5: CPU load during the execution of testing scenarios in the VNF instance.

However, to reduce network overhead caused by the reactive dissemination model, proactive dissemination requires implementing a measuring agent within the VNF platform, which consequently consumes computational resources to configure, trigger, execute, and evaluate measurements. As a result, a slightly higher CPU consumption is noted when this model is enforced, as shown in Figure 5. This processing overhead increases as the number of measurement

agents and the complexity of their operations grow. Nevertheless, in the specific case study presented, the increase in CPU consumption, although prominent in percentage terms when compared to the reactive model, is not significant in absolute terms (no more than 0.5% of additional CPU load). Moreover, the difference between the dissemination models decreases as the measurement interval is reduced, a phenomenon that becomes explicit when analyzing the 90th percentile across the scenarios presented in Figure 5.

As demonstrated in this case study, the adopted measurement model can directly affect the network and the functions and services deployed on it. It is worth noting that the experiment was conducted in a controlled infrastructure and does not fully reflect the heterogeneity of demands and VNFs found in real-world environments, where multiple factors can naturally modify the nature and magnitude of these impacts. For instance, depending on a network function's attributes, a passive encapsulation model may be a suitable alternative for reducing the network load associated with reactive dissemination, potentially making it comparable to proactive dissemination. Therefore, there is no golden rule for implementing observability in NFV-based networks. Instead, the specific characteristics of the environment and the metrics of interest must be carefully considered to select the most appropriate model for each measurement characteristic, enabling robust and effective observability.

6 Conclusion

The flexibility introduced by the NFV paradigm has brought several advantages to modern dynamic network environments, making it an efficient approach for guaranteeing quality of service across different contexts. However, the unpredictability of dynamic network environments demands efficient solutions to manage them effectively. Observability, can be seen as a strategy for holistically determining the network state in a precise way. It is the cornerstone for enabling operators to make the best management decisions for the network. However, defining metrics to determine these states and the best measurement models to assess them is far from trivial.

In this paper, we broadly discussed observability in the context of the NFV paradigm. First, we identified the observability enablers of the paradigm, considering the NFV reference architecture proposed by ETSI. We then examined how observability metrics and measurement models can be applied to NFV, from both conceptual and technical perspectives. The operational elements across NFV working domains were characterized as observed or observer entities, and state-of-the-art architectures and frameworks were analyzed with respect to their support for practical observability implementations. Finally, we presented a case study demonstrating that adopting different observability measurement models can lead to distinct impacts in NFV-based network environments. In particular, the chosen dissemination models may introduce additional network overhead or increased CPU consumption at VNF instances.

Implementing observability in conjunction with NFV is inherently complex, as multiple factors must be considered, including the definition of metrics aligned with the objectives of network managers and operators, as well as measurement models that assess metric values in a balanced manner, providing timely information with minimal overhead. Nevertheless, despite this complexity, observability offers significant opportunities to manage and optimize networks by considering both the current network state and predicting future states based on historical observations. Examples of operations that can benefit from refined observability include mapping and placing virtualized services on the underlying infrastructure, preventing and mitigating network bottlenecks, and strategically migrating virtualized functions to improve quality of service. Furthermore, through observability, cost models can be designed and tailored to each type of infrastructure environment based on its behavioral characteristics, such as network traffic patterns, deployed VNFs and services, and end-user demands, enabling detailed analysis and planning for network management.

Acknowledgments

This work was developed with the support of the Federal University of Parana (COFPI/PRPI 19/2025 - 23075.058047/2025-51), the Program of Academic Excellence (PROEX) - Coordination for the Improvement of Higher Education Personnel (CAPES - AUXPE 88881.189840/2025-01) and the National Council for Scientific and Technological Development (CNPq - Project 305108/2025/5).

References

[1] Thomas M. Chen. 2001. Increasing the observability of Internet behavior. *Communications of the ACM* 44, 1 (2001), 93–98.

[2] Leonardo da Cruz Marcuzzo, Vinicius F Garcia, Vitor Cunha, Daniel Corujo, Joao P Barraca, Rui L Aguiar, Alberto E Schaeffer-Filho, Lisandro Z Granville, and Carlos RP dos Santos. 2017. Click-on-osv: A platform for running click-based middleboxes. In *IFIP/IEEE Symposium on Integrated Network and Service Management*. IFIP/IEEE, Lisbon, Portugal, 885–886.

[3] João Paulo De Araujo, Luciana Arantes, Elias P Duarte, Luiz A Rodrigues, and Pierre Sens. 2017. A publish/subscribe system using causal broadcast over dynamically built spanning trees. In *2017 29th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. IEEE, 161–168.

[4] Marco De Benedictis and Antonio Lioy. 2019. A proposal for trust monitoring in a network functions virtualisation infrastructure. In *IEEE Conference on Network Softwarization*. IEEE, Paris, France, 1–9.

[5] Guilherme Werneck De Oliveira, Michele Nogueira, Aldri Luiz dos Santos, and Daniel Macêdo Batista. 2023. Intelligent VNF Placement to Mitigate DDoS Attacks on Industrial IoT. *IEEE Transactions on Network and Service Management* 20, 2 (2023), 1319–1331.

[6] ETSI Industry Specification Group (ISG) NFV. 2014. Network Functions Virtualisation (NFV); NFV Performance & Portability Best Practices.

[7] ETSI Industry Specification Group (ISG) NFV. 2014. Network Functions Virtualisation (NFV); Service Quality Metrics.

[8] ETSI Industry Specification Group (ISG) NFV. 2024. *Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Architectural Framework Specification.* Group Specification GS NFV 006 v4.5.1. European Telecommunications Standards Institute (ETSI).

[9] Bruno E Farias, José Flauzino, and Elias P Duarte Jr. 2025. VNF-Cache: An In-Network Key-Value Store Cache Based on Network Function Virtualization. *arXiv preprint arXiv:2512.19964* (2025).

[10] Ummay Faseeha, Hassan J Syed, Fahad Samad, Sehar Zehra, and Hamza Ahmed. 2025. Observability in Microservices: An In-Depth Exploration of Frameworks, Challenges, and Deployment Paradigms. *IEEE Access* 13 (2025), 72011–72039.

[11] Vinicius Fulber-Garcia, José Flauzino, Carlos RP Dos Santos, and Elias P Duarte. 2023. An ETSI-compliant Architecture for the Element Management System: The Key for Holistic NFV Management. In *IEEE International Conference on Network and Service Management*. IEEE, Niagara Falls, Canada, 1–9.

[12] Vinicius Fulber-Garcia, José Flauzino, Giovanni Venâncio, Alexandre Huff, and Elias P Duarte Junior. 2024. Breaking the limits: Bio-inspired sfc deployment across multiple domains, clouds and orchestrators. In *2024 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. IEEE, 1–6.

[13] Vinicius Fulber-Garcia, Alexandre Huff, Carlos R P dos Santos, and Elias P Duarte Jr. 2020. Network service topology: Formalization, taxonomy and the CUSTOM specification model. *Elsevier Computer Networks* 178 (2020), 107337.

[14] Vinicius Fulber-Garcia, Alexandre Huff, Leonardo da C Marcuzzo, Marcelo C Luizelli, Alberto E Schaeffer-Filho, Lisandro Z Granville, Carlos RP dos Santos,

and Elias P Duarte Junior. 2021. Customizable Deployment of NFV Services. *Journal of Network and Systems Management* 29, 3 (2021), 1–27.
[15] Vinicius Fulber-Garcia, Marcelo C Luizelli, Carlos R Paula dos Santos, Eduardo J Spinosa, and Elias P Duarte Jr. 2023. Customizable mapping of virtualized network services in multi-datacenter environments based on genetic metaheuristics. *Journal of Network and Systems Management* 31, 4 (2023), 71.
[16] Vinícius F Garcia, Leonardo C Marcuzzo, Giovanni V Souza, Lucas Bondan, Jéferson C Nobre, Alberto E Schaeffer-Filho, Carlos RP dos Santos, Lisandro Z Granville, and Elias P Duarte Jr. 2019. An nsh-enabled architecture for virtualized network function platforms. In *International Conference on Advanced Information Networking and Applications*. Springer, 376–387.
[17] Vinicius Fulber Garcia, Leonardo da C Marcuzzo, Alexandre Huff, Lucas Bondan, Jeferson C Nobre, Alberto Schaeffer-Filho, Carlos RP dos Santos, Lisandro Z Granville, and Elias P Duarte. 2019. On the design of a flexible architecture for virtualized network function platforms. In *IEEE Global Communications Conference*. IEEE, Big Island, Hawaii, USA, 1–6.
[18] Vinícius Fülber Garcia, Giovanni Venâncio De Souza, Elias Procopio Duarte Jr, Thales Nicolai Tavares, Leonardo Da Cruz Marcuzzo, Carlos RP Dos Santos, Muriel Figueredo Franco, Lucas Bondan, Lisandro Zambenedetti Granville, Alberto Egon Schaeffer-Filho, et al. 2020. On the design and development of emulation platforms for NFV-based infrastructures. *International Journal of Grid and Utility Computing* 11, 2 (2020), 230–242.
[19] Lav Gupta, Tara Salman, Maede Zolanvari, Aiman Erbad, and Raj Jain. 2019. Fault and performance management in multi-cloud virtual network services using AI: A tutorial and a case study. *Computer Networks* 165, C (2019), 22 pages.
[20] Mu He, Alberto Martínez Alba, Arsany Basta, Andreas Blenk, and Wolfgang Kellerer. 2019. Flexibility in softwarized networks: Classifications and research challenges. *IEEE Communications Surveys & Tutorials* 21, 3 (2019), 2600–2636.
[21] Thanh Tung Hoang, Manh Linh Pham, and Hoai Son Nguyen. 2024. Scaling and Dynamic Resource Reallocation in NFV: Challenges and Research Perspectives. *International Journal of Electrical and Computer Engineering Systems* 15, 10 (2024), 851–863.
[22] Alexandre Huff, Giovanni Venâncio, Vinícius Fulber Garcia, and Elias P Duarte. 2020. Building multi-domain service function chains based on multiple nfv orchestrators. In *IEEE Conference on Network Function Virtualization and Software Defined Networks*. IEEE, Virtual, 19–24.
[23] Alexandre Huff, Giovanni Venancio, Leonardo da C Marcuzzo, Vinicius F Garcia, Carlos RP dos Santos, and Elias P Duarte. 2018. A holistic approach to define service chains using click-on-osv on different nfv platforms. In *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 1–6.
[24] IETF. 2011. RFC 6241: Network Configuration Protocol (NETCONF). https://datatracker.ietf.org/doc/html/rfc6241
[25] IETF. 2011. RFC 6390: Guidelines for Considering New Performance Metric Development. https://datatracker.ietf.org/doc/html/rfc6390
[26] IETF. 2017. Benchmarking Methodology for Virtualization Network Performance. https://datatracker.ietf.org/doc/html/draft-huang-bmwg-virtual-network-performance-03
[27] IETF. 2017. RFC 8172: Considerations for Benchmarking Virtual Network Functions and Their Infrastructure. https://datatracker.ietf.org/doc/html/rfc8172
[28] IETF. 2022. RFC 9232: Network Telemetry Framework. https://datatracker.ietf.org/doc/html/rfc9232
[29] Wolfgang John, Farnaz Moradi, Bertrand Pechenot, and Pontus Sköldström. 2017. Meeting the observability challenges for VNFs in 5G systems. In *IFIP/IEEE Symposium on Integrated Network and Service Management*. IFIP/IEEE, Lisbon, Portugal, 1127–1130.
[30] R.E. Kalman. 1960. On the general theory of control systems. *International IFAC Congress on Automatic and Remote Control* 1, 1 (1960), 491–502.
[31] Taekhee Kim, Taehwan Koo, and Eunkyoung Paik. 2015. SDN and NFV benchmarking for performance and reliability. In *Asia-Pacific Network Operations and Management Symposium*. IEEE, Busan, Korea, 600–603.
[32] Stanislav Lange, Hee-Gon Kim, Se-Yeon Jeong, Heeyoul Choi, Jae-Hyung Yoo, and James Won-Ki Hong. 2019. Predicting vnf deployment decisions under dynamically changing network conditions. In *IFIP/IEEE/ACM International Conference on Network and Service Management*. IFIP/IEEE/ACM, Halifax, Canada, 1–9.
[33] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. 2013. Observability of complex systems. *Proceedings of the National Academy of Sciences* 110, 7 (2013), 2460–2465.
[34] Marcelo Caggiani Luizelli et al. 2017. The actual cost of software switching for NFV chaining. In *IFIP/IEEE Symposium on Integrated Network and Service Management*. IFIP/IEEE, Lisbon, Portugal, 335–343.
[35] Charity Majors, Liz Fong-Jones, and George Miranda. 2022. *Observability Engineering*. O'Reilly, Springfield, Missouri.
[36] Joao Martins, Mohamed Ahmed, Costin Raiciu, Vladimir Olteanu, Michio Honda, Roberto Bifulco, and Felipe Huici. 2014. {ClickOS} and the Art of Network Function Virtualization. In *USENIX Symposium on Networked Systems Design and Implementation*. USENIX, Seattle, USA, 459–473.
[37] Chris Misa, Ramakrishnan Durairajan, Reza Rejaie, and Walter Willinger. 2021. Revisiting Network Telemetry in COIN: A Case for Runtime Programmability. *IEEE Network* 35, 5 (2021), 14–20.
[38] Arthur N Montanari and Luis A Aguirre. 2020. Observability of network systems: A critical review of recent results. *Journal of Control, Automation and Electrical Systems* 31, 6 (2020), 1348–1374.
[39] OpenConfig Project. 2025. OpenConfig: Vendor-neutral, model-driven network management designed by users. https://www.openconfig.net/
[40] perfSONAR Project. 2025. perfSONAR: performance Service-Oriented Network monitoring ARchitecture. https://www.perfsonar.net/
[41] Guto Leoni Santos, Diego de Freitas Bezerra, Elisson da Silva Rocha, Leylane Ferreira, André Luis Cavalcanti Moreira, Glauco Estácio Gonçalves, Maria Valéria Marquezini, Ákos Recse, Amardeep Mehta, Judith Kelner, et al. 2022. Service function chain placement in distributed scenarios: a systematic review. *Springer Journal of Network and Systems Management* 30, 1 (2022), 4.
[42] Jie Sun, Yi Zhang, Feng Liu, Huandong Wang, Xiaojian Xu, and Yong Li. 2022. A survey on the placement of virtual network functions. *Elsevier Journal of Network and Computer Applications* 202 (2022), 103361.
[43] Thales Nicolai Tavares, Leonardo da Cruz Marcuzzo, Vinícius Fulber Garcia, Giovanni Venâncio de Souza, Muriel Figueredo Franco, Lucas Bondan, Filip De Turck, Lisandro Zambenedetti Granville, Elias Procópio Duarte Junior, Carlos Raniery Paula dos Santos, et al. 2018. Niep: Nfv infrastructure emulation platform. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, 173–180.
[44] Rogerio C Turchetti and Elias Procopio Duarte. 2015. Implementation of failure detector based on network function virtualization. In *2015 IEEE International Conference on Dependable Systems and Networks Workshops*. IEEE, 19–25.
[45] Rogério C Turchetti, Elias P Duarte Jr, Luciana Arantes, and Pierre Sens. 2016. A QoS-configurable failure detection service for internet applications. *Journal of Internet Services and Applications* 7, 1 (2016), 9.
[46] Muhammad Usman, Simone Ferlin, Anna Brunstrom, and Javid Taheri. 2022. A survey on observability of distributed edge & container-based microservices. *IEEE Access* 10 (2022), 86904–86919.
[47] Niels L. M. van Adrichem, Christian Doerr, and Fernando A. Kuipers. 2014. OpenNetMon: Network monitoring in OpenFlow Software-Defined Networks. In *IEEE Network Operations and Management Symposium*. IEEE, Krakow, Poland, 1–8.
[48] Giovanni Venâncio, Vinícius Fulber Garcia, Leonardo da Cruz Marcuzzo, Thales Nicolai Tavares, Muriel Figueredo Franco, Lucas Bondan, Alberto Egon Schaeffer-Filho, Carlos Raniery Paula dos Santos, Lisandro Zambenedetti Granville, and Elias P. Duarte Jr. 2021. Beyond vnfm: Filling the gaps of the etsi vnf manager to fully support vnf life cycle operations. *International Journal of Network Management* 31, 5 (2021), e2068.
[49] Giovanni Venâncio, Rogério C Turchetti, Edson T Camargo, and Elias P Duarte Jr. 2021. VNF-Consensus: A virtual network function for maintaining a consistent distributed software-defined network control plane. *International Journal of Network Management* 31, 3 (2021), e2124.
[50] Giovanni Venâncio, Rogério C Turchetti, and Elias P Duarte. 2019. Nfv-rbcast: Enabling the network to offer reliable and ordered broadcast services. In *2019 9th Latin-American Symposium on Dependable Computing (LADC)*. IEEE, 1–10.
[51] Giovanni Venâncio, Rogério C Turchetti, and Elias Procópio Duarte Jr. 2022. Nfv-coin: Unleashing the power of in-network computing with virtualization technologies. *Journal of Internet Services and Applications* 13, 1 (2022), 46–53.
[52] Mirko Viroli and Andrea Omicini. 2002. Specifying agent observable behaviour. In *International Joint Conference on Autonomous Agents and Multiagent Systems*. ACM, New York, USA, 712–720.
[53] Qixia Zhang, Fangming Liu, and Chaobing Zeng. 2021. Online Adaptive Interference-Aware VNF Deployment and Migration for 5G Network Slice. *IEEE/ACM Transactions on Networking* 29, 5 (2021), 2115–2128.

Towards Automatic Discovery of Correlations between Unstructured and Structured Data in Automotive Data Lakes

Rodrigo Gonçalves
goncalves@lisha.ufsc.br
Software/Hardware Integration Lab,
Federal University of Santa Catarina
Florianópolis, Santa Catarina, Brasil

José Luis Conradi Hoffmann
hoffmann@lisha.ufsc.br
Software/Hardware Integration Lab,
Federal University of Santa Catarina
Florianópolis, Santa Catarina, Brasil

Antônio Augusto Fröhlich
guto@lisha.ufsc.br
Software/Hardware Integration Lab,
Federal University of Santa Catarina
Florianópolis, Santa Catarina, Brasil

João R. Campos
jrcampos@dei.uc.pt
CISUC/LASI, DEI,
University of Coimbra
Coimbra, Coimbra, Portugal

Abstract

Data Lakes have emerged as an architectural approach for integrating data from multiple heterogeneous sources into large-scale data processing pipelines. A key challenge lies in correlating structured, semi-structured, and unstructured data through metadata in order to transform raw data into actionable information and support automated decision-making at scale. This research paper builds upon SmartData, a self-contained data construct originally designed for structured data, to propose a Data Lake infrastructure that enables efficient integration across diverse data modalities. We introduce *SmartDataContext*, an intermediary semi-structured representation that encapsulates *SmartData Models* and supports time-series tagging. Additionally, we propose a *SmartTagging* framework that extracts semantic information from unstructured data and applies text relationship analysis to automatically discover correlations. The proposed approach is evaluated through an automotive case study, demonstrating the identification of relationships between sections of standards documents in PDF format and corresponding *SmartDataContexts*. From a data management viewpoint, *SmartTagging* acts as an "active metadata" mechanism for automotive data lakes, allowing for automatic and continuous discovery of tags from contextual artifacts and external standards (here, ETSI C-ITS documents), thus, avoiding manual catalogs or schema-on-read and the creation of data-swamps.

Keywords

SmartData, Automotive Systems, Cloud, Internet of Things.

1 Introduction

From Industry 4.0 to transportation systems, data lie at the core of the design of modern safety-critical systems. Data are sensed, processed, secured, stored, and transmitted to support the decision-making processes required for higher levels of autonomy, including performance optimization, fault detection, and autonomous control [11, 22, 25, 26]. A major step in Industry 4.0 is the integration of data into big data processing pipelines deployed on cloud-based IoT platforms, where metadata plays a key role in guiding data analysis. In this context, Fröhlich proposed the concept of SmartData [6], defined as a data item enriched with sufficient metadata to render it self-contained with respect to semantics, spatial location, timing, and trustworthiness.

Beyond the metadata provided by SmartData, IoT data analysis often requires additional contextual information. Examples include the types of devices used, their firmware, software versions, as well as characteristics of the environment in which the data were collected (e.g., weather conditions). Furthermore, identifying which standards and reference literature are related or applicable to a given data set can further improve the quality and reliability of subsequent reasoning and analysis.

To enable such additional information to be stored, retrieved, and organized together with IoT data (SmartData), we introduce in this work the concept of *SmartDataContext*. A SmartDataContext is an entity that can contain structured, semi-structured, and unstructured data associated with a given time-series. It also provides a multi-tag-based classification of SmartData, either manually (user-specified) or automatically, based on the SmartDataContext contents and its relationship with available unstructured data. This automated tagging system is named *SmartTagging*. Other than automatically identifying correlated unstructured data entries, the *SmartTagging* concept enables the comparison and retrieval of related time-series based on their associated tags. We leverage on this concept to accelerate large-scale data-series analysis with hints of highly correlated time-series if they share tags. In this way, the main contributions of this paper are:

- A Data Lake architecture for Automotive data that efficiently integrates structured, semi-structured, and unstructured data.
- The *SmartDataContext* a Data Lake construct that allows for semi-structured definition of *SmartData Models*.
- The *SmartTagging* framework, a Data Lake tool that allows for automatic tagging extraction to identify relationships

ADVANCE 2026, Florianopolis, SC-Brazil

ACM ISBN XXXXXXXXXX
https://doi.org/10.1145/XXXXXXX.XXXXXXXX

between time-series, *SmartData Models*, and unstructured data.

The remainder of this paper is organized as follows: Section 2 presents related works. Section 3 presents our automotive data lake architecture supported by SmartData and SmartDataContext. In Section 4 we introduce the SmartTagging solution. Section 5 presents our experimental results considering an automotive case study where we investigate *SmartDataContext* of simulations and their relationship to ETSI Cooperative ITS standards. Finally, Sections 6 and 7 discuss our findings and outline directions for future work.

2 Related Work

This section surveys two main strands of research that underpin Automotive Data Lake and the baseline for the proposed SmartDataContext and SmartTagging. First, we review methods from natural language processing (NLP) for identifying relationships among unstructured textual data, with a focus on representation and similarity modeling techniques that enable implicit links between information fragments to be made explicit. Second, we discuss architectural approaches to data lakes in the automotive domain, emphasizing how contemporary designs address scalability, governance, and heterogeneity in large-scale telemetry and IoT-driven environments.

Compared to knowledge graphs and active data catalogs [24], SmartData and SmartDataContext provide a more practical approach to managing IoT data flowing through the Data Lake as they do not require previously established schemas or extensive AI processing.

2.1 Natural Language Processing for Unstructured Data Relationships

To identify related data in SmartDataContext, one approach is to analyze textual representations of such data and assess their similarity or relationship. Natural language processing (NLP) for unstructured data relationship identification relies critically on how texts are represented and compared so that implicit links between fragments of information can be made explicit. Vector space models based on term frequency–inverse document frequency (TF–IDF) encode documents and queries as high-dimensional sparse vectors, where each dimension corresponds to a vocabulary term weighted by its local frequency and global rarity [13, 20]. In this setting, relationships between a given text extract and a corpus of unstructured documents are calculated as similarity scores (e.g., cosine similarity) between their TF–IDF vectors, enabling the discovery of which documents are most lexically related to the extract [19].

Relationship identification in unstructured text may require going beyond surface-level lexical overlap, especially when relevant documents express related concepts using different terminologies, paraphrases, or domain-specific jargon. Transformer-based language models such as BERT learn contextualized token representations that encode syntactic and semantic information by jointly conditioning on the surrounding context [3]. Sentence-level architectures like Sentence-BERT (SBERT) project sentences, paragraphs, and document segments into a dense embedding space in which cosine similarity reflects semantic relatedness rather than mere term co-occurrence [18]. When a text extract and each candidate document (or document segment) are embedded with such models, nearest-neighbor search in the embedding space can reveal deeper semantic relationships, allowing the system to identify which documents are most conceptually connected to the extract, even in the absence of vocabulary overlap. Recent benchmark studies show that transformer-based embedding models may outperform traditional TF–IDF baselines on semantic matching tasks across a variety of domains, including technical and biomedical corpora where nuanced terminology and long-range dependencies are prevalent [12, 23]. At the same time, dense embeddings are more computationally expensive to compute and index.

2.2 Automotive Data Lake Architectures

In the automotive domain, data lakes are emerging as a central architectural pattern to cope with the massive, heterogeneous, and fast-evolving telemetry produced by modern vehicles and infrastructure. Conceptually, they extend classical data warehousing by relaxing up-front schema constraints while still aiming to provide governed, semantically rich storage across structured, semi-structured, and unstructured data [8–10]. General surveys emphasize that data lakes must balance low-cost, scalable storage with active metadata extraction, data quality management, and provenance tracking in order to avoid devolving into "*data swamps*", where data are neither findable nor reusable [9]. For automotive applications, where safety, regulatory compliance, and long-term traceability are crucial, this tension between flexibility and governance is even more pronounced, since the same platform must support both exploratory analytics and safety-critical, auditable workloads over extended time spans.

Architecturally, state-of-the-art data lake solutions converge towards multi-zone or multi-pond layouts, in which raw, cleansed, and curated data are managed in separate logical areas, combined with rich metadata catalogs and governance processes [8, 10, 17]. Metadata models such as GEMMS and subsequent data-vault–based approaches aim to capture schema, lineage, quality, and usage information at different granularities, enabling schema evolution and cross-dataset integration without sacrificing flexibility [7, 17]. Experience reports on implementing heterogeneous data lakes show that schema-on-read alone is insufficient in practice; instead, successful deployments introduce lightweight but explicit modeling layers and employ hybrid architectures that combine batch storage with streaming and lakehouse mechanisms [14, 21]. This evolution towards lakehouse designs—where ACID properties, indexing, and query optimization are brought closer to the data lake—responds directly to the need for reliable, repeatable analytics and machine-learning pipelines on top of highly heterogeneous data.

When these concepts are specialized to IoT and automotive settings, additional requirements emerge around distributed processing, low-latency ingestion, and edge intelligence. IoT-oriented surveys highlight the importance of layered architectures that distribute computation between devices, edge nodes, and the cloud, and they catalog typical processing styles (stream, rule-based, complex event, semantic, and learning-based) along with "13 V's" challenges specific to IoT big data [1, 2]. Concrete designs for intelligent transportation systems advocate edge-based data lake architectures in

which roadside or gateway nodes perform initial cleaning, cataloging, and transformation of vehicular data, while centralized cloud layers provide large-scale storage and batch analytics [5]. In production automotive deployments, large manufacturers have reported multi-layer big data architectures that combine MQTT ingestion, Kafka buffering, object storage, and NoSQL backends to serve both internal and external applications at scale, with strong emphasis on privacy, GDPR compliance, and fine-grained access control [15]. Complementary work on Kappa-style streaming architectures and smart-farming data lakes further illustrates how message queues, stream processors, and specialized stores (e.g., HBase, Druid) can be orchestrated to support high-throughput, low-latency analytics over continuous sensor data [16].

3 Proposed Data Lake Architecture

The proposed automotive data lake architecture is organized around the interactions among connected vehicles, human stakeholders, and a cloud-based analytics back end. The baseline use-case diagram (Figure 1) captures this ecosystem by showing how Drivers, Vehicles, Developers, Fleet Managers, Urban Managers, and potential Misusers interact with the system through different interfaces, such as smartphone applications, electronic control units (ECUs), and web APIs.

These actors request services including navigation support, vehicle inspection records, operational optimization, and planning, while the architecture also explicitly models malicious behaviors (e.g., cyberattacks) that must be detected and mitigated. At a high level, the system is divided into two main subsystems: a vehicle subsystem, which executes sensing, control, and local decision-making; and a data lake subsystem, which performs large-scale storage, processing, and analysis of vehicular and contextual data following the SmartData/SmartDataContext paradigm. Within this structure, the vehicle subsystem acts both as a producer and consumer of data. Through a smartphone interface or in-vehicle HMI, the driver can request navigation, participate in gamification scenarios, or trigger inspection and diagnostic workflows.

The vehicle subsystem communicates with the data lake via V2X links, continuously sending sensing streams, motion vectors, waypoints, driver profiles, and embedded predictions. In return, it receives optimized trajectories, parameter configurations, and high-level guidance. External services—such as policy repositories, digital maps, and weather providers—feed additional information into the data lake, which is fused with the vehicle-generated data to support route planning, traffic-aware navigation, and safety analysis. Developers, fleet managers, and urban managers access this processed information through secure web APIs, using it to plan and optimize operations, while the architecture ensures that all interfaces are hardened against misuser attacks to preserve reliability, availability, and privacy.

The internal structure of the data lake subsystem is detailed in Figure 2, which decomposes the platform into three logical layers: API, data processing, and data storage. The API layer terminates all external connections and enforces cross-cutting concerns such as mutual TLS authentication, certificate-based access control, and anonymization when applicable (see Section 3.3). It also integrates domain-specific *SmartData Models* and certificate management, thereby binding incoming requests to the appropriate logical domains and privacy policies. Human–machine interface (HMI) modules provide visualization, configuration, and management capabilities for operators and data consumers. Beneath the API layer, the data processing layer hosts modular components that implement the core data-flow of the system. Collectors and extractors ingest heterogeneous vehicular and contextual data; qualifiers assess quality and consistency; and explorers support interactive or semi-automatic enrichment, including tagging and exploratory analysis.

These modules feed into a transformation chain comprising aggregators, runners, mappers, optimizers, and causality analyzers, which together normalize, map, and analyze data according to domain-specific workflows. The data storage layer then persists the results in dedicated repositories for time series (SmartData), contextual models (SmartDataContext), and unstructured artifacts, while an external data agent continuously enriches these stores with auxiliary sources such as maps, weather, and policy databases. The outputs—processed data, learned models, mappings, reports, and analysis products—are exposed back to vehicles and planning tools, closing the loop between on-board systems and cloud analytics.

3.1 SmartData

A SmartData is a piece of data enriched with enough metadata to make it self-contained regarding semantics, spatial location, timing, and trustfulness [6]. Each piece of data is tagged with a 32-bit type identifier called *Unit*, designating either an *SI Physical Quantity* or plain digital data. Several properties from the data can be directly derived from the Unit attribute alone, like domain limits (e.g., the temperature in Kelvin must always yield values in $\mathbb{R}_+$). Moreover, a SmartData record carries the origin of a sample, given by 3-D coordinates of its generation (relative to the Earth's mass center) and a high-resolution timestamp t (`Origin(x,y,z,t)`). The SmartData can be finally described, as defined in [6] as:

$$SmartData(unit, value, Origin(x, y, z, t), r, des) \tag{1}$$

where *des* is a disambiguation identifier for multiple sensors of the same Unit and space-time coordinates.

SmartData objects can be stored in collections as time series based on their spatial-temporal coordinates, defined as a sphere with central point (x, y, z) and radius r, with the respective SmartData unit, as defined in [6] as:

$$SmartData\ series(unit, x, y, z, r, t0, tf) \tag{2}$$

In this way, a definition of series as in (2) can be used for geographic queries. Data points from the resulting series can be retrieved considering specific time intervals.

Finally, a *SmartData series* can encompass a text field *semantics* dedicated to the data semantics, which is expected to be filled out when defining a data model for a system. For instance, consider a SmartData series originating from an accelerometer inside an inertial measurement unit, thus, the semantic field could comprise the following: "Measured Acceleration in the x-axis using an accelerometer in the IMU set at the vehicle geometrical center (limits of IMU within [-16g,16g])". Whenever dealing with periodic time-series, an

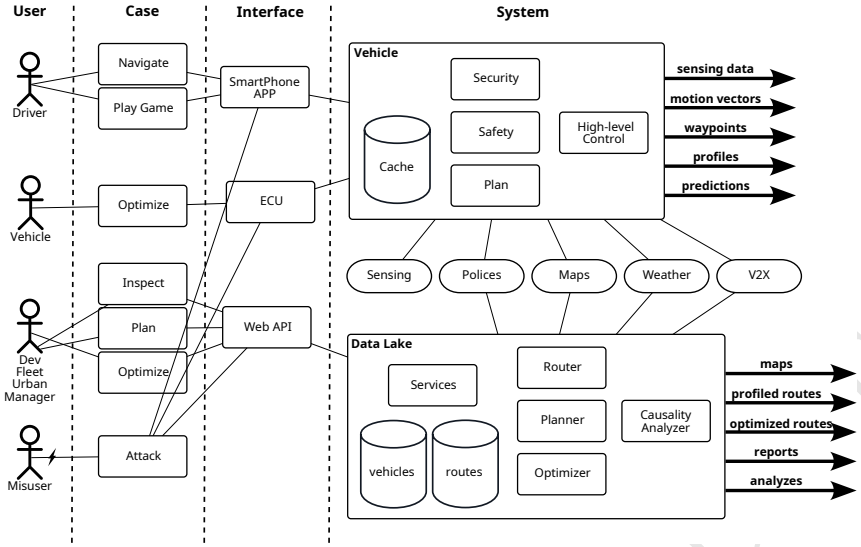

Figure 1: Main Data Lake Use Cases

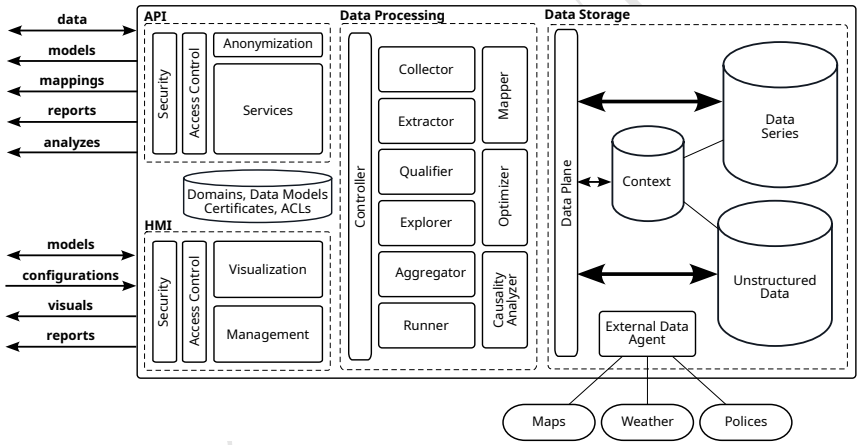

Figure 2: Main Data Lake Components

optional field *period* can be included to describe the expected periodicity of data generation. Finally, for sporadic data sampling that is associated with an event, we can also use an optional field *event* comprising a description of the event of interest. These features will be of high-value when exploring automatic tagging alongside contextual information.

3.2 Contextual Information

A *SmartDataContext* denotes any contextual information associated with a given *SmartData series*. Each SmartDataContext instance comprises a unique identifier (*id*), a *content* field, a reference to the associated *SmartData series*, and a set of *tags* qualifying the SmartData series, based on the SmartData and SmartDataContext data. The *content* can be provided in semi-structured format (e.g., JSON) and/or in unstructured format (e.g., audio, video, documents). When semi-structured content is used, it should ideally conform to a domain-specific *conceptual model* that prescribes recommended fields and structures. Such a conceptual model promotes standardization and improves the comprehensibility, interoperability, and manipulability of SmartDataContext instances.

When storing unstructured data, SmartDataContext supports the inclusion of semi-structured content alongside the raw artifacts. This semi-structured layer may be derived from the unstructured data—for example, metadata extracted from a video or audio recording—thereby facilitating indexing, querying, and downstream processing.

A SmartDataContext must define at least one of the *tags* or *content* properties. When tags are present, the SmartDataContext acts as a qualifier for its associated SmartData series, enabling efficient retrieval and filtering of SmartData based on contextual criteria.

To persist SmartDataContext instances, we adopt a multi-paradigm storage architecture that combines relational, NoSQL, and object storage systems, as illustrated in Figure 3. Each storage component fulfills a distinct role in supporting the flexible and scalable representation of contextual data. The relational database manages structured metadata and maintains referential integrity between entities such as SmartDataContext, SmartData series (series_id in Table smartdatacontext_series_v_1_1 and v_1_2), and associated tags. The NoSQL database stores semi-structured content and feature attributes in simple, domain-based collections without enforcing a fixed schema, thereby allowing arbitrary contextual information to be indexed and queried efficiently. Object storage is used to persist unstructured artifacts (e.g., media files), also organized by domain but kept schema-less to support scalable management of large binary assets.

This hybrid approach provides both the flexibility required to accommodate heterogeneous contextual data and the structural guarantees necessary for efficient querying, indexing, and future machine-learning–based processing.

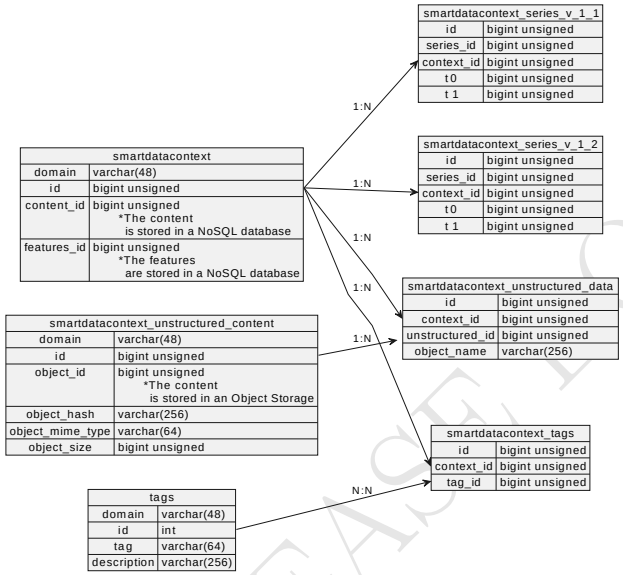

Figure 3: Conceptual SmartDataContext storage architecture.

3.3 Security and Privacy

The Data Lake security model is based on mutual TLS (mTLS) and a Public Key Infrastructure (PKI) using X.509 certificates. A trusted Certificate Authority (CA) issues and signs certificates for both clients and the Data Lake API. During connection establishment, the server and client mutually authenticate by validating each other's certificates against the CA. Only if both identities are verified is an encrypted TLS session established, ensuring confidentiality, integrity, and protection against man-in-the-middle attacks.

Authorization is enforced after authentication by mapping certificate attributes (such as roles, domains, or policy identifiers) to access control rules. The Access Control component extracts these attributes from the client certificate and determines whether the requested operation (e.g., data insertion or retrieval) is permitted. Requests with insufficient permissions are rejected before reaching core Data Lake services, making the security layer a strict gateway for all interactions.

The Data Lake privacy-preserving strategy is designed to prevent vehicle re-identification and tracking while still enabling meaningful data analysis. The primary privacy risk is the correlation of space, time, and persistent identifiers (signatures), which could allow an attacker to trace individual vehicles. To address this, the system strictly separates identity management from data storage: real vehicle identities and certificates remain under the control of a trusted Certificate Authority, while the Data Lake only handles pseudonymous or anonymized references. Direct identifiers are never exposed to end users through queries.

Vehicle signatures used for data attestation are handled in a privacy-aware manner. A signature derived from the vehicle certificate is never stored in plaintext; instead, it is hashed (e.g., SHA-256) before being stored in the SQL metadata layer and is omitted entirely from query results. During queries, results are aggregated over a domain and a specified space–time region, and vehicle identifiers are replaced with shuffled, query-local indices (e.g., 0, 1, 2). This guarantees that time series from different vehicles are not mixed, while preventing users from linking data across queries or reconstructing long-term vehicle trajectories. While spatial anonymization is applied to remove precise location traces (e.g., either truncation of spatial location or by moving the center of the coordinate system to a new (0,0,0)), time anonymization was intentionally discarded to preserve the usability of the data for analytics and machine learning. Overall, the strategy ensures that the Data Lake supports large-scale analysis while minimizing the risk of identity leakage, location tracking, and cross-query correlation, aligning privacy protection with practical data utility.

4 SmartTagging

SmartTagging is conceived as an automatic and semi-automatic tagging process for SmartData series, driven by the contextual information associated with those series as well as the series own data.Rather than relying on users to manually describe and classify data—an activity that is costly, error-prone, and often neglected—SmartTagging treats tags as a first-class semantic layer that can be derived from both the series themselves (e.g., their units and semantic fields) and from SmartDataContext artifacts such as models, reports, and configuration files.

The initial exploration of the SmartTagging concept established four main goals: (i) to identify low-preparation methods for automatic tagging based on external knowledge sources; (ii) to define a consistent and reusable semantics for tags across the data lake; (iii) to investigate how to analyze relatedness between SmartData series through their tags; and (iv) to propose an architectural framework that operationalizes SmartTagging on top of existing data lake services. This framework and its application are described in subsequent sections of this paper.

Starting from unstructured textual artifacts, in this case a selected subset of ETSI Cooperative ITS (C-ITS) standards, we describe how

documents are ingested and pre-processed, how contextual information is extracted and represented within SmartDataContext, and how tags are generated. We then present the SmartTagging pipeline as a plugin-based infrastructure that operates over heterogeneous contextual materials, and show how tag similarity and clustering techniques leverage a domain-specific taxonomy to compute relatedness among SmartData series. By integrating unstructured document processing, contextual modeling, and taxonomy-driven tag comparison, the data insertion flow bridges the gap between raw data and higher-level semantic correlation. This allows the data lake to support use cases such as standard-aware navigation, recommendation, and cross-series analysis without relying solely on numerical similarity of time series.

To implement an unstructured data processing pipeline capable of discovering semantically related SmartData series in large-scale data lakes, we introduce the SmartTagging framework. This framework addresses the challenge of relatedness by shifting the focus from raw numeric similarity to contextual semantics, using SmartDataContext and associated artifacts as the primary source of semantic information.

Architecturally, the framework (Figure 4) is designed as a plugin-based infrastructure that operates over SmartDataContext material. Each plugin is responsible for analyzing a specific type or aspect of contextual data, identifying patterns or features of interest, and associating one or more tags with the corresponding SmartDataContext. This design allows heterogeneous analyzers to coexist and evolve independently while sharing a common tagging vocabulary. Examples of possible plugins include audio processors (to detect abnormal sounds), video processors (to identify visible damage on components), document analyzers (to locate relevant keywords in reports), image processors (to detect visual anomalies), spreadsheet inspectors (to mine tabular metadata), and JSON analyzers (to interpret structured context models).

Together, these plugins transform diverse contextual artifacts into a unified tag space that can be exploited across domains. The framework also specifies a detailed execution and coordination model for these plugins. When a SmartDataContext is to be analyzed, the framework may invoke all registered plugins for the domain, but each plugin is responsible for deciding whether the given context is applicable and should be processed.

Operationally, SmartTagging is designed as a background, cyclic, and periodic process that runs in parallel with other data-lake activities. SmartDataContext objects that have been analyzed before can be reconsidered in subsequent cycles, enabling continuous refinement of tags as new plugins, improved heuristics, or additional knowledge sources become available.

Finally, plugin registration is domain-aware: domains can have zero or more associated plugins, each marked as “common” (generic, applicable to any series in the domain) or “specialized” (more expensive, bound to specific series). This provides the framework with a flexible mechanism to mix broad coverage with deep, targeted analyses where needed.

4.1 Tag Similarity and Clustering

SmartTagging tag similarity and clustering address the problem of how to use tags, once they have been automatically assigned to SmartData series and SmartDataContext, to identify related, though not necessarily numerically similar, time series in the data lake. Instead of comparing raw values or units directly, the approach focuses on comparing sets of tags pertaining to each series, aiming to infer semantic relatedness at a higher level of abstraction. This leads to the central idea of organizing tags in a structured, hierarchical taxonomy and using distances in this hierarchy to assess how related two tag sets are.

Several key guidelines were considered when designing the taxonomy: (i) it should be extensible, not requiring structural revisions whenever new tags are introduced; (ii) it should be capable of handling a mixture of different kinds of tags, with various origins (e.g., standards-based tags, tags derived from patterns extracted from data, and so forth); and (iii) a tag, by itself, should clearly define its meaning - that is, given a tag, a user should be able to understand its semantics and where to look for additional information.

Based on these guidelines, we developed a procedure to automatically build a taxonomy. We will now present a run-example considering a subset (six) of ETSI Cooperative ITS (C-ITS) standards as unstructured data that can be automatically associated with SmartData series through SmartDataContext and SmartTagging:

The ETSI standards define a coherent framework for safety-related vehicular messaging in the European ITS architecture. At the facilities layer, ETSI EN 302 637-2 V1.4.1 specifies the Cooperative Awareness (CA) service and the generation of periodic Cooperative Awareness Messages (CAMs) between vehicles and roadside units to describe their dynamic state. ETSI EN 302 637-3 V1.3.1 defines the Decentralized Environmental Notification (DEN) service and the event-driven DEN Messages (DENMs), describing detected hazards, abnormal traffic situations, and other environment-related events. Together, they describe how awareness and hazard notifications interact with networking, security, and channel access mechanisms.

Release 2 specifications update these services through ETSI TS 103 900 V2.2.1 (Cooperative Awareness Service) and ETSI TS 103 831 V2.1.1 (Decentralized Environmental Notification Service), refining service architectures, interfaces, and message lifecycles to address scalability and deployment experience. It also refines CAM generation rules, timing constraints and exception handling, reflecting deployment experience and the need for scalable, interoperable C-ITS services across heterogeneous ITS stations. Both services rely on the common ITS data dictionary defined in ETSI TS 102 894-2, with versions V1.3.1 (Release 1) and V2.1.1 (Release 2) documenting the evolution of shared data elements and semantics, providing a comprehensive catalogue of data elements and data frames, specifying their semantics, units and ASN.1 representations. Collectively, these six standards form a coherent, layered ecosystem for interoperable cooperative vehicular applications and serve as a realistic test case for automated correlation with Data Lake time-series.

The taxonomy is presented in Figure 5. Due to space constraints, the figure shows only an excerpt of the full taxonomy. At the top level, a node `Standard` aggregates all standard-related tags; this node can later be complemented by others such as `Proprietary` for vendor-specific tags. Other categories of tags can be added by extending the taxonomy at this level (e.g., `ImagePatterns`, `AudioPatterns`, `VehicleDefects`). Below `Standard`, the taxonomy introduces a `TC-ITS` node to reference the standardization technical committee, and under it standard-specific nodes derived

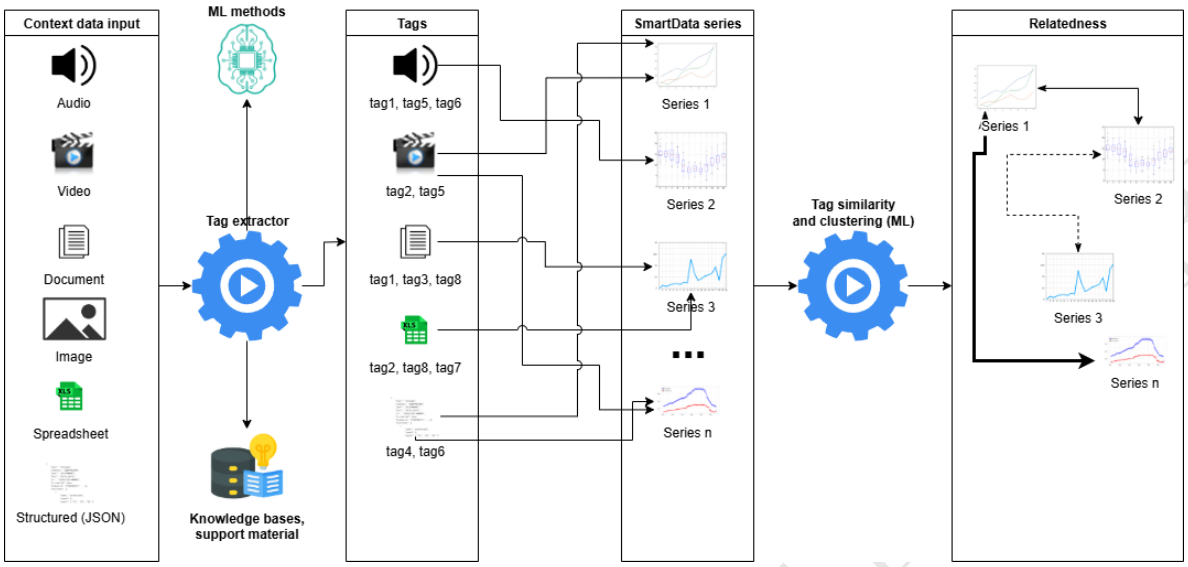

Figure 4: SmartTagging framework.

from the standard names of the PDF files used to build the taxonomy. Under each standard, a node is introduced for each section of the respective document.

This hierarchy reflects how domain knowledge is structured in practice, creating a semantic scaffold on which tag similarity can be defined. The leaves of this taxonomy are the actual tags assigned to SmartData series and contexts. Each leaf tag is written following a pattern derived from its ancestors in the taxonomy, so that the tag string itself encodes its path in the hierarchy.

Consequently, given only a tag, one can recover its position in the taxonomy and, therefore, its relationships to other tags. Such an encoding simplifies both storage and computation, as there is no need to maintain a separate mapping structure to locate tags in the hierarchy. This model also supports transparent interpretation by human experts, who can read a tag and immediately understand its associated standard, document section, and specific semantic meaning.

On top of this taxonomy, we developed a heuristic for comparing sets of tags associated with different SmartData series. The heuristic operates as follows: for each tag in one set, the algorithm identifies the nearest tag (shortest path in the taxonomy graph) in the other set and uses these distances to compute an average distance between the two sets. This average distance is taken as the relatedness score between the corresponding SmartData series.

This approach is designed to be general: once suitable taxonomies exist for different domains (e.g., additional ITS standards, proprietary schemas, or other knowledge bases), the same distance-based matching can be reused to support tag-driven navigation, recommendation, and correlation analysis across the data lake.

5 Case Study: Automotive Data Simulation and ETSI-CITS Documents

To investigate automatic tagging, two complementary experiments were conducted. We consider a concrete end-to-end deployment scenario for an automotive data lake. We instantiate the proposed models and data flows on top of a controlled, yet realistic, experimental environment, demonstrating that the infrastructure can ingest, store, and analyze heterogeneous vehicular data while preserving the semantics required by downstream analytics. The implementation covers both the SmartData series and their contextual SmartData models, as well as the unstructured-document processing pipeline used to relate time series and metadata to external standards.

In the first experiment, SmartData units were linked to elements of ETSI ICC Standards (introduced in Section 4.1, using the a SmartData Model comprising all elements defining a SmartData series description, including the semantics textual field, and the definitions from the standards (PDF documents segmented by sections) as inputs to an NLP-based matching process. The Standards documents and SmartData unit descriptions were tokenized and embedded into a vector space; term frequency–inverse document frequency (TF–IDF) was then used to compute similarity scores and capture the lexical and semantic relationships between units and the standards documentation.

In the second experiment, *SmartDataContext*, a JSON representation of the environment and configuration related to one or more SmartData series, were decomposed into smaller structural fragments, whose attribute names and values were converted into token sequences. This JSON was based on the conceptual model for driving observation introduced earlier. The text fragments were again matched against the standards documents, identifying for each fragment the most related document sections.

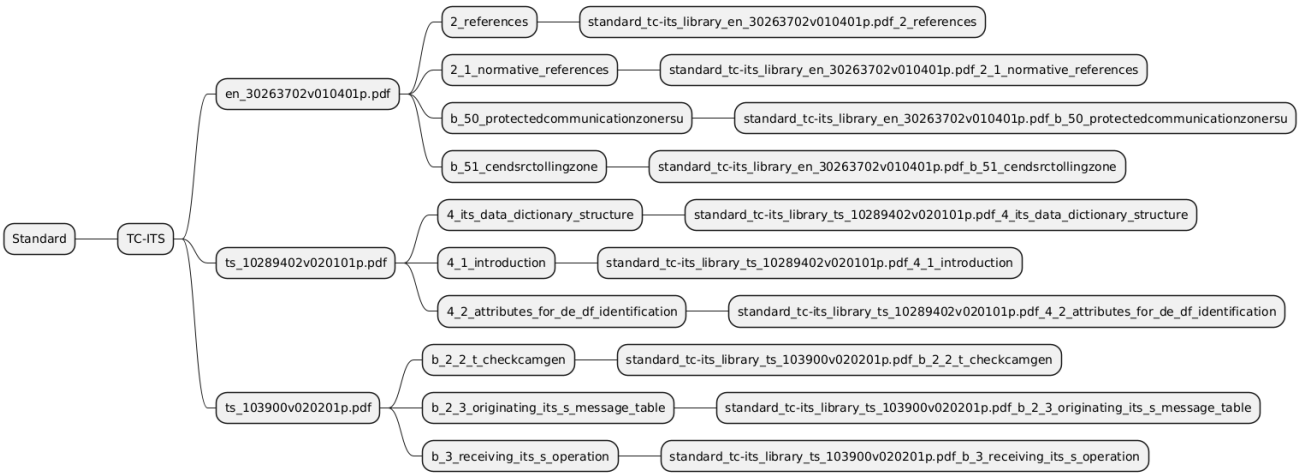

Figure 5: ASN Taxonomy.

Once tags are available, SmartTagging treats them as the primary vehicle for expressing and computing relationships between SmartData series. These relationships are then structured and interpreted according to the taxonomies introduced in Section 4.1, enabling systematic comparison, retrieval, and aggregation of SmartData series based on their tag profiles.

5.1 Structured Data Setup: Automotive Simulation on CARLA

To generate realistic *SmartData Models* with configurable and controllable semantics, we leverage on Cars Learning to Act (CARLA) simulator [4], an open-source simulator that promotes detailed scenarios for simulating vehicles in different traffic and weather conditions. CARLA exposes a rich set of automotive sensors—such as cameras, LiDAR, RADAR, GPS, and IMU—together with data acquisition and telemetry interfaces. This combination makes it suitable not only for classical perception and control tasks, but also for studies on eco-driving, V2X applications, and predictive maintenance, where the simulator can generate large volumes of data under controlled road and weather conditions, including degradations such as potholes and speed bumps that affect vehicle components.

The goal is to exercise the SmartData and SmartTagging framework under realistic driving conditions. Using CARLA as a high-fidelity urban-driving simulator, we instantiate a *SmartDataContext* on top of synthetic yet physically consistent telemetry. The resulting time-series are ingested into the automotive data lake together with rich contextual information about vehicles, roads, and environmental conditions, providing a representative workload for the proposed infrastructure. Examples of these contextual data embedded into a *SmartDataContext content* are presented in Figures 6 and 7 as JSON semi-structured data.

5.1.1 SmartData Model. The data model defines a structured set of vehicle signals and associated semantics tailored for self-contained metadata to allow for data analysis in a secure and privacy-aware automotive data lake. Table 1 presents a slice of the SmartData


```
"weatherElements": [
    {
        "kind": "SUNLIGHT",
        "comment": "Carla (-90 to +90)",
        "intensity": {
            "unit": 3300018468,
            "valueF32": 0.2617993877991494
        }
    },
    {
        "kind": "FOG",
        "comment": "",
        "intensity": {
            "unit": 4160749569,
            "valueF32": 0.0
        },
        "viewDistance": {
            "unit": 3298183460,
            "valueF32": 0.0
        },
        "fallOff": {
            "unit": 3297167652,
            "valueF32": 0.0
        }
    },
```


Figure 6: SmartDataContext JSON extract - Weather

model derived for vehicle dynamics in a CARLA simulation. We have omitted the column semantics due to space constraints, but it includes a brief textual description of the data, for instance, "Measured Engine Revolutions in Hz. Used to obtain torque in the torque curve." for "Engine Revolutions", and "Gear currently engaged in the vehicle. Used to obtain gear ratio. Integer values in the range [0,20]." for Gear.

For each quantity, the SmartData Model specifies the original source unit, the normalized SmartData unit, numerical range, sampling frequency, and semantics, ensuring that heterogeneous simulators and on-board sensors can be mapped into a coherent representation. This allows for a single structured data representation inside the data lake, where conversions are derived directly from the data model. Other data captured during simulation also includes

Table 1: SmartData model for vehicle dynamics in CARLA Simulation.

Name	Original Qty.	SmartData Unit	des	SmartData Qty.	Conversion	Min	Max	Sampling Rate
Engine Revolutions	rpm	0xC4923924 (F32)	0	Hz	value $\times \frac{1}{60}$	0.0	167.0	10 Hz
Gear	Gear enum	0x98000006 (I32)	0	Gear enum	value	0	20	10 Hz
Vehicle Speed	km/h	0xC4962924 (F32)	0	m/s	value $\times \frac{1}{3.6}$	−17	139	10 Hz
Velocity X	m/s	0xC4963924 (F32)	0	m/s	value	−17	139	10 Hz
Velocity Y	m/s	0xC4963924 (F32)	1	m/s	value	−17	139	10 Hz
Velocity Z	m/s	0xC4963924 (F32)	2	m/s	value	−17	139	10 Hz
Drag	N	0xC496A924 (F32)	0	N	value	0	1.28	10 Hz
IMU Longitudinal Acceleration	m/s^2	0xC4962924 (F32)	0	m/s^2	value	−156.9	+156.9	10 Hz
IMU Lateral Acceleration	m/s^2	0xC4962924 (F32)	1	m/s^2	value	−156.9	+156.9	10 Hz
IMU Vertical Acceleration	m/s^2	0xC4962924 (F32)	2	m/s^2	value	−156.9	+156.9	10 Hz
IMU Pitch Rate	Degree/s	0xC4B23924 (F32)	0	rad/s	value $\times \pi/180$	−34.90659	+34.90659	10 Hz
IMU Roll Rate	Degree/s	0xC4B23924 (F32)	1	rad/s	value $\times \pi/180$	−34.90659	+34.90659	10 Hz
IMU Yaw Rate	Degree/s	0xC4B23924 (F32)	2	rad/s	value $\times \pi/180$	−34.90659	+34.90659	10 Hz
Pitch	degree	0xC4B24924 (F32)	0	rad	value $\times \pi/180$	$-\pi$	π	10 Hz
Vehicle Mass	kg	0xC492C924 (F32)	0	kg	value	0	45000	10 Hz
Brake	%	0xF8000001 (D64)	0	%	value	0	1	10 Hz
Throttle	%	0xF8000001 (D64)	1	%	value	0	1	10 Hz
Altitude	m	0xC4964924 (F32)	9	m	value	0	1	10 Hz
Front overhang	m	0xC4964924 (F32)	0	m	value	0	4	10 Hz
Rear overhang	m	0xC4964924 (F32)	1	m	value	0	4	10 Hz
Wheel base	m	0xC4964924 (F32)	2	m	value	0	15	10 Hz
Stiffness of Front Suspensions	N/m	0xC492A924 (F32)	0	N/m	value	0	1,000,000	10 Hz
Damping of Front Suspensions	N s/m	0xC492B924 (F32)	0	N s/m	value	0	20,000	10 Hz
Stiffness of Rear Suspensions	N/m	0xC492A924 (F32)	1	N/m	value	0	1,000,000	10 Hz
Damping of Rear Suspensions	N s/m	0xC492B924 (F32)	1	N s/m	value	0	20,000	10 Hz


```
"sensors": [
    {
        "kind": "IMU",
        "description": "",
        "model": "carla.sensor.other.imu",
        "firmware": "",
        "interferences": [],
        "readings": [
            {
                "measurement": "longitudinal",
                "domain": "sdav",
                "signature": 42,
                "dev": 0,
                "unit": 3298175268,
                "x": 0,
                "y": 0,
                "z": 0,
                "t0": 1750721687184964,
                "tf": 1750721708184964,
                "version": "1.2"
            },
```


Figure 7: SmartDataContext JSON extract - Sensors

information regarding wheel telemetry, suspension parameters, and wheel base information.

5.2 SmartTagging: Structured and Non-Structured Relationship

SmartTagging for SmartData models targets the contextual layer that describes how, where, and under which conditions SmartData series are produced. A SmartData model captures the environment and key configuration parameters associated with one or more series. In a simulated scenario, such as an experiment in the CARLA simulator, the model may include the simulated city, weather conditions, vehicle models, suspension type, engine characteristics, and other relevant parameters. In real-world deployments, it can describe properties of the driver, vehicle, road, and weather, among others. These models are grounded in the conceptual modeling of the domain and are currently represented as JSON documents, which makes them both machine-processable and flexible enough to encode heterogeneous contextual information.

To enable automatic tagging of SmartData models, the JSON representation is processed by a custom parser that decomposes the document into smaller structural fragments. Each JSON element that acts as a structure, i.e., contains sub-elements—is extracted individually, producing a collection of context fragments that reflect different parts of the overall model. Figures 6 and 8 illustrate the original JSON structure and its corresponding textual representation. For each fragment, attribute names and their corresponding values are converted into a token-based string representation, while numerical values are discarded because they carry limited direct information for text-based semantic matching. The resulting strings can be viewed as “flattened” textual projections of the JSON structures, capturing the semantic cues present in field names and textual content while preserving enough structure to distinguish different contextual aspects.

```
[
  "weatherElements kind: \"SUNLIGHT\" comment: \"Carla (-90 to +90)\" intensity unit valueF32
  kind: \"FOG\" comment: \"\" intensity unit valueF32 viewDistance unit valueF32 fallOff unit
  valueF32 kind: \"WIND\" comment: \"\" intensity kind: \"RAIN\" comment: \"\" intensity kind: \"
  WETNESS\" comment: \"\" intensity",
  "environment trafficConditions type: \"TRAFFIC_LIGHTS\" present: false type: \"TRAFFIC_SIGNS\"
  present: false simulator name: \"Carla\" version: \"0.9.15-243-g9a59963ca-dirty\" url: \"\"
  location simulator name: \"Carla\" version: \"1.0.0\" url: \"\" description: \"Town04 with
  additional road features\" map: \"Carla/Maps/Town04_Lisha\" map_features ramp start latitude
  longitude altitude end latitude longitude altitude rx ry rz w class: \"asc\" start latitude
  longitude altitude end latitude longitude altitude rx ry rz w class: \"desc\" start latitude
  longitude altitude end latitude longitude altitude rx ry rz w start latitude longitude altitude
  end latitude longitude altitude rx ry rz w class: \"asc\" start latitude longitude altitude end
```


Figure 8: JSON extract converted to text

These fragment-level string representations are then matched against textual representations of ITS ASN.1 elements or standards-document (PDF) extracts using TF–IDF in a vector space. The outcome is a set of tags derived from the standards and attached to the SmartData models, and by extension to the associated SmartData series. The results of this experiment were qualitatively comparable to those obtained when tagging SmartData units directly, indicating that contextual models are also effective sources of semantic information. More broadly, this approach demonstrates that SmartData models can be automatically tagged using external knowledge sources with relatively low preparation cost, and it opens a path for incorporating additional, higher-quality knowledge bases and more advanced NLP methods in future iterations.

5.3 Tag extractors for ETSI standards association

We developed two tag extractors: one focused on associating SmartData Models with ETSI standards, and another that matches all information in *SmartDataContext* to the same standards. In both cases, we adopted an information-retrieval-based process, following the techniques discussed in Section 2.1. The process is organized into the following steps: (i) pre-process the standards documents and convert them into a vector-space representation; (ii) convert the series-associated data (SmartData unit descriptions or SmartData models) into a vector-space representation within the same space; and (iii) match the vectors obtained in step (ii) against the standards-document representations.

We selected the six ETSI standards (introduced in Section 4.1) for the experiment: ETSI TS 103 831 V2.1.1 (2022-11), ETSI TS 102 894-2 V1.3.1 (2018-08), ETSI TS 102 894-2 V2.1.1 (2022-11), ETSI EN 302 637-3 V1.3.1 (2019-04), ETSI TS 103 900 V2.2.1 (2025-02), and ETSI EN 302 637-2 V1.4.1 (2019-04). For each document, the textual content is extracted using standard PDF processing libraries and segmented according to the document structure (sections). These text excerpts were then converted into a common TF–IDF vector-space representation shared by all documents. Step (ii) consists of obtaining the *SmartData Model* and converting them into the same vector space, thus enabling direct comparison between *SmartData series* and ETSI standard excerpts. Finally, the SmartDataContext associated with SmartData series were converted into textual representations (using the process introduced in the previous section) and mapped into the same vector space.

With all elements embedded in a common vector space, a TF–IDF –based information-retrieval method was applied to determine which standards excerpts best matched the SmartData units and SmartData models. Table 2 presents selected examples of SmartData units and their corresponding ETSI standards sections. Due to space constraints, only one section is shown per *SmartData series*, although a single series may match multiple sections within the same ETSI standard. The “Max Score” column indicates the highest similarity score obtained among all sections of the corresponding standard PDF for the given SmartData unit description. Note that a lot of SmartData Series presented high "Max Score" for "annex_a_normative_data_type_specifications". This section consisted as the mother section to all subsection within it, and, therefore, if any of its subsections match, a match is also expected in the higher level in the document format hierarchy.

Similarly, we matched the textual representations of SmartDataContext to the ETSI standards. In this case, we adopted an alternative scoring strategy, counting how many fragments from the SmartDataContext representation selected a given excerpt from the ETSI standards as their best match. Table 3 provides an overview (again, limited by space) of the resulting matches.

We validated the solution by running a manual inspection of the "Max Score" result. First, we discarded any relationship with "Max Score" < 10. Next, the manual analysis considered three levels of relationship: 0 - Unrelated; 1 - Slightly related; 2 - Strongly Related. The final results accounted for 188 relationships with "Max Score" ≥ 10 from witch 128 were classified as Strongly related, 38 as Slightly Related, and 22 as Unrelated;

Overall, the results indicate that traditional, cost-effective information -retrieval techniques—such as vector-space representations combined with TF–IDF—can successfully identify unstructured documents that are semantically associated with SmartData series, based on their series semantics (*SmartData Model*) and/or SmartDataContext. This approach can be extended to any type of document that can be represented textually, including web pages, forum posts, and technical reports. We also performed preliminary experiments using large language models (LLMs) instead of TF–IDF to match SmartData units and SmartData models to standards excerpts. However, the qualitative results were similar, while the computational cost of the LLM-based approach was substantially higher. Consequently, we opted to adopt TF–IDF for the current implementation.

6 Discussions

The proposed architecture combines three elements often treated separately in automotive big-data systems: (i) Structured Data (*SmartData*) and Contextual Information (*SmartDataContext*) to uniformly represent sensor streams and their context; (ii) a tag-centric *SmartTagging* layer that relates heterogeneous *SmartData series* through semantics rather than raw values; and (iii) an automatic discovery methodology to identify relationships between Structured and Unstructured data, thus, exposing its semantics to data scientists and other users of the Data Lake with sufficient access permissions to this data. Together with case study of the automotive data lake on CARLA simulations and ETSI C-ITS standards, this shows that a lightweight metadata layer can improve data discoverability and cross-series correlation without changing existing ingestion or storage pipelines.

From a data management viewpoint, *SmartTagging* acts as an “active metadata” mechanism for automotive data lakes. Instead of relying only on manual catalogs or schema-on-read, the framework continuously derives tags from contextual artifacts and external

Table 2: SmartData series matching

series	PDF File	Max Score	Matched Section
IMU Lateral Acceleration	ts_102894v02010031p0.pdf	0.4249598393	annex_a_normative_data_type_specifications
IMU Longitudinal Acceleration	ts_102894v02010031p0.pdf	0.413202318	annex_a_normative_data_type_specifications
Steeering angle	ts_102894v02010031p0.pdf	0.4069368073	a_79_de_steeringwheelangleconfidence
IMU Vertical Acceleration	ts_102894v02010031p0.pdf	0.37674164	annex_a_normative_data_type_specifications

Table 3: SmartData model matching

PDF Filename	Section	Matches
ts_10289402v010301p.pdf	a_74_de_speedvalue	34
ts_10289402v010301p.pdf	a_42_de_lateralaccelerationvalue	25
ts_10289402v010301p.pdf	a_15_de_curvaturevalue	22
en_30263702v010401p.pdf	annex_e_informative_extended_cam_generation	8
en_30263702v010401p.pdf	6_cam_dissemination	6
en_30263702v010401p.pdf	6_1_cam_dissemination_concept	6
en_30263702v010401p.pdf	6_1_cam_dissemination_requirements	6
en_30263702v010401p.pdf	6_1_2_ca_basic_service_activation_and_termination	6
ts_103831v020101p.pdf	b_42_roadtype	4
ts_103831v020101p.pdf	b_43_roadworks	4
ts_103831v020101p.pdf	b_44_speedlimit	4
ts_103831v020101p.pdf	b_45_startingpointspeedlimit	4

standards (here, ETSI C-ITS documents). This addresses a common cause of "data swamps", where data lack usable metadata. By treating tags as first-class objects linked to SmartDataContext and SmartData and exposing them via a Discovery API, the architecture improves findability and reusability while remaining compatible with multi-zone and lakehouse-style deployments.

The experiments with ETSI standards show that classical, low-cost information-retrieval methods can connect unstructured documentation to structured telemetry in a meaningful way. Matching SmartData Models and SmartDataContext fragments to standards sections yields tags that encode both where each quantity (unit) is defined in the standards and which parts of the conceptual model it relates to. This enables workflows in which engineers search for time series associated with specific ETSI elements and analytics pipelines that reason about which signals may be affected by changes in a standard. Although the evaluation is qualitative, the examples in Tables 2 and 3 indicate that the approach can recover plausible links between physical quantities, contextual models, and normative documents.

In our future works, we would extend the analysis to cover *SmartTagging* robustness under more heterogeneous conditions, including proprietary schemas, technical reports, maintenance reports, and multilingual documentation. In addition, the assessment is based on illustrative matches rather than metrics such as precision, recall, or ranking quality over a labeled ground truth, which would require annotated datasets and user studies with domain experts.

The taxonomy and similarity model introduce further constraints. The taxonomy in Section 4.1 is largely derived from ETSI document structure and manual design rules. This yields an interpretable hierarchy but incurs maintenance cost as new standards and knowledge bases are added. The distance-based heuristic on the taxonomy graph is simple and efficient, but it does not account for tag importance, uncertainty, or conflicting evidence. Adding probabilistic or learning-based layers on top of the taxonomy could refine similarity estimates and improve ranking and recommendation.

Architecturally, the solution is intentionally conservative in its use of machine learning, relying mainly on TF–IDF-based retrieval. This simplifies deployment on-premises or in commercial clouds and keeps costs predictable, but limits the richness of semantic relationships that can be captured. Transformer-based embeddings and cross-encoders could improve matching quality, especially for long or noisy contextual documents, at the expense of higher computational and operational complexity.

Finally, the approach raises questions around safety, privacy, and governance. The same tags that improve discoverability and correlation may expose sensitive patterns about drivers, vehicles, or operating conditions. In safety-critical contexts, traceability to standards and explicit semantics support assurance and compliance, but must be balanced against data minimization and strict access control. Embedding SmartTagging in a broader governance framework—where tags themselves are governed, audited, and managed over their lifecycle is therefore a key step before adopting the approach in production-scale environments or mixed fleets.

7 Final Remarks

This paper introduced SmartDataContext and SmartTagging as extensions to the SmartData paradigm for automotive data lakes. We proposed a tag-centric framework in which SmartData series and their contextual artifacts are enriched with tags derived from

internal models and external standards. A plugin-based SmartTagging pipeline and a taxonomy-guided similarity model were implemented and evaluated using data incoming from multiple configuration of simulations in CARLA, and linked to ETSI C-ITS standards. The experiments, though exploratory, show that simple information-retrieval techniques can automatically associate standards documentation with units and contextual models, enabling standard-aware navigation and correlation.

For practitioners, the main contribution is an architectural pattern that can be added incrementally to existing ingestion and storage solutions. Using *SmartDataContext* as the anchor for contextual artifacts and exposing tags through a discovery layer allows engineers to quickly identify relevant time series and logs, relate them to domain standards, and reason about their semantics without inspecting raw signals or schemas.

Future work first involves scaling and evaluation: applying the framework to larger, more diverse datasets (including real fleet data and additional standards) and quantitatively assessing tag quality and similarity metrics through precision/recall, ablation studies comparing TF–IDF and transformer-based methods, and user studies. A second line of work is to explore hybrid matching strategies that combine TF–IDF with neural embeddings, using dense encoders as re-rankers and supporting few-shot or weakly supervised refinement of domain-specific taggers based on user feedback.

A third direction is to evolve the taxonomy and similarity model toward richer knowledge representations. Aligning the current hierarchy with ontologies and knowledge graphs for intelligent transportation systems, and adding dimensions such as safety integrity, privacy, or maintenance criticality, would make SmartTagging more directly useful for safety and compliance. A fourth avenue is tighter integration with downstream analytics, using tags for feature selection, scenario mining, anomaly detection, and automated model monitoring and dataset versioning.

Acknowledgements

This work was partially funded by Fundação de Apoio da UFMG (Fundep), through Linha VI – Conectividade Veicular, a prioritary program from Mover (Mobilidade Verde e Inovação), project AutoDL (29271.03.01/2023.04-00).

References

[1] A. R. Al-Ali, R. Gupta, I. Zualkernan, and S. K. Das. 2024. Role of IoT Technologies in Big Data Management Systems: A Review and Smart Grid Case Study. *Pervasive and Mobile Computing* 100 (2024).

[2] Mehak Bansal, Inderveer Chana, and Siobhán Clarke. 2021. A Survey on IoT Big Data: Current Status, 13 V's Challenges, and Future Directions. *Comput. Surveys* 53, 6 (2021), 1–59.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4171–4186. arXiv:1810.04805

[4] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. CARLA: An Open Urban Driving Simulator. In *Proceedings of the 1st Annual Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 78)*, Sergey Levine, Vincent Vanhoucke, and Ken Goldberg (Eds.). PMLR, 1–16. https://proceedings.mlr.press/v78/dosovitskiy17a.html

[5] Diego Fernandes, Lucas L. Moura, Gabriel Santos, Gabriel S. Ramos, Francisco Queiroz, and Ana L. L. Aquino. 2023. Towards Edge-Based Data Lake Architecture for Intelligent Transportation System. In *Proceedings of the ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (MSWiM'23)*. 1–8.

[6] Antônio Augusto Fröhlich. 2018. SmartData: an IoT-ready API for sensor networks. *International Journal of Sensor Networks* 28, 3 (2018), 202.

[7] Christian Giebler, Christoph Gröger, Erik Hoos, and Bernhard Mitschang. 2019. Modeling Data Lakes with Data Vault: Practical Experiences, Assessment, and Lessons Learned. In *Proceedings of the International Conference on Conceptual Modeling*. 63–77.

[8] Christian Giebler, Christoph Gröger, Erik Hoos, Holger Schwarz, and Bernhard Mitschang. 2019. Leveraging the Data Lake: Current State and Challenges. In *Lecture Notes in Computer Science*. Vol. 11708. 179–188.

[9] Rihan Hai, Christos Koutras, Christoph Quix, and Matthias Jarke. 2023. Data Lakes: A Survey of Functions and Systems. *IEEE Transactions on Knowledge and Data Engineering* 35, 12 (2023), 12571–12590.

[10] Bill Inmon. 2016. *Data Lake Architecture: Designing the Data Lake and Avoiding the Garbage Dump.* Technics Publications.

[11] Yuchen Jiang, Shen Yin, and Okyay Kaynak. 2018. Data-Driven Monitoring and Safety Control of Industrial Cyber-Physical Systems: Basics and Beyond. *IEEE Access* 6 (2018), 47374–47384. https://doi.org/10.1109/ACCESS.2018.2866403

[12] Aditya Lahiri, Sangeeta Shukla, Ben Stear, Taha Mohseni Ahooyi, Katherine Beigel, Elizabeth Margolskee, and Deanne Taylor. 2025. Benchmarking Transformer Embedding Models for Biomedical Terminology Standardization. *Machine Learning with Applications* 21 (2025), 100683. https://doi.org/10.1016/j.mlwa.2025.100683

[13] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press.

[14] Hammad Mehmood et al. 2019. Implementing Big Data Lake for Heterogeneous Data Sources. In *Proceedings of the IEEE International Conference on Data Engineering Workshops*. 37–44.

[15] Abdelkader Mostefaoui, Mohamed A. Merzoug, Amine Haroun, Amine Nassar, and François Dessables. 2022. Big Data Architecture for Connected Vehicles: Feedback and Application Examples from an Automotive Group. *Future Generation Computer Systems* 134 (2022), 374–387.

[16] Jean B. Nkamla Penka, Samira Mahmoudi, and Olivier Debauche. 2021. A New Kappa Architecture for IoT Data Management in Smart Farming. *Procedia Computer Science* 191 (2021), 17–24.

[17] Christoph Quix, Rihan Hai, and Ivan Vatov. 2016. Metadata Extraction and Management in Data Lakes with GEMMS. *Complex Systems Informatics and Modeling Quarterly* 9 (2016), 67–83.

[18] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 3982–3992. https://doi.org/10.18653/v1/D19-1410

[19] Stephen E. Robertson. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009), 333–389. https://doi.org/10.1561/1500000019

[20] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing and Management* 24, 5 (1988), 513–523. https://doi.org/10.1016/0306-4573(88)90021-0

[21] Johannes Schneider, Christoph Gröger, Andreas Lutsch, Holger Schwarz, and Bernhard Mitschang. 2024. The Lakehouse: State of the Art on Concepts and Technologies. *SN Computer Science* 5, 5 (2024), 449.

[22] S. A. Seshia, S. Hu, W. Li, and Q. Zhu. 2017. Design Automation of Cyber-Physical Systems: Challenges, Advances, and Opportunities. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36, 9 (2017), 1421–1434. https://doi.org/10.1109/TCAD.2016.2633961

[23] Mehrzad Shahinmoghadam and Ali Motamedi. 2025. Benchmarking pre-trained text embedding models in aligning built asset information. *Scientific Reports* 15, 23866 (2025). https://doi.org/10.1038/s41598-025-09052-5

[24] Jibin Wang, Yanmei Guo, Yong Jiang, Jing Shang, Zhuo Chen, Yu Liu, Qingyuan Hu, and Zheng Yin. 2024. Data Catalogs with Artificial Intelligence and Active Metadata: A Case Study of China Mobile. In *2024 Sixth International Conference on Next Generation Data-driven Networks (NGDN)*. 453–458. https://doi.org/10.1109/NGDN61651.2024.10744079

[25] Hao Ye, Le Liang, Geoffrey Ye Li, JoonBeom Kim, Lu Lu, and May Wu. 2018. Machine Learning for Vehicular Networks: Recent Advances and Application Examples. *IEEE Vehicular Technology Magazine* 13, 2 (2018), 94–101. https://doi.org/10.1109/MVT.2018.2811185

[26] Junping Zhang, Fei-Yue Wang, Kunfeng Wang, Wei-Hua Lin, Xin Xu, and Cheng Chen. 2011. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Transactions on Intelligent Transportation Systems* 12, 4 (2011). https://doi.org/10.1109/TITS.2011.2158001

The Impact of Process Competition on Energy Consumption: Analysis and Modeling

Eduardo Gomes Campos
Polytechnic School of University of São Paulo
São Paulo, SP, Brazil
eduardogc0303@usp.br

Rafaela Sousa de Alencar Lacerda
Polytechnic School of University of São Paulo
São Paulo, SP, Brazil
rafaelalacerda@usp.br

Adnei Willian Donatti
Polytechnic School of University of São Paulo
São Paulo, SP, Brazil
adnei.donatti@usp.br

Joberto S. B. Martins
Universidade Salvador (UNIFACS)
Salvador, BA, Brazil
joberto.martins@animaeducacao.com.br

Charles C. Miers
Santa Catarina State University
Joinville, SC, Brazil

Tereza C. M. B. Carvalho
Polytechnic School of University of São Paulo
São Paulo, SP, Brazil
terezacarvalho@usp.br

Abstract

With the development of distributed systems, the need to manage the sharing of machines among multiple simultaneous users arises. In the cloud computing context, the instantiation of virtual machines and containers by different users utilizing the same infrastructure leads to a dispute for physical computational resources. In this regard, this paper analyses a process's energy consumption as a function of the competition for computational resources it encounters. Investigating this behavior is fundamental for many applications, such as pricing in cloud computing services, and for task scheduling and load balancing, while increasing energy efficiency. To determine this behavior, experiments were conducted and resulted in a dependency on the number of processor cores of the physical machine hosting the process. As the number of cores increases, the process's energy consumption as a function of the competition it faces transitions from linear to a root function.

CCS Concepts

• **Hardware** → **Energy metering**.

Keywords

Energy Consumption, Resource Competition, Process, Kubernetes

ACM Reference Format:
Eduardo Gomes Campos, Rafaela Sousa de Alencar Lacerda, Adnei Willian Donatti, Joberto S. B. Martins, Charles C. Miers, and Tereza C. M. B. Carvalho. 2025. The Impact of Process Competition on Energy Consumption: Analysis and Modeling. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

With the development of distributed systems, handling physical machines shared by multiple users at once is necessary. For instance, in the context of cloud computing, the process of instantiating virtual machines causes a scenario in which computational resources are being competed for by multiple individuals. However, this utilization is usually measured according to the machine specifications and period of use, disregarding how each uses those resources. Nowadays, energy consumption and sustainability are at high stakes since the world is trying to become more aware of its environmental impact. Since energy consumption has become a relevant cost in cloud computing data centers [28], there is an urge to assess the environmental impact of computer usage and one way is to measure its energy consumption.

One topic worth mentioning is how the competition for Central Processing Units (CPUs) resources affects the power consumption of a process. In this context, this article proposes an approach to analyze the energy consumption at the process level in these scenarios, which unveils the possibility of modeling a process power. In this context, we perform experiments to measure how much energy a process would consume as it faces competition for CPU resources. With that data available, we correlate how high the competition is and how much power the main process consumes. The final objective is to obtain a function $\mathcal{W}(p)$ to represent the power consumed by the process. In this case, p represents the percentage of the CPU used by the competition and, therefore, $p \in [0, 100 - q]$, given that q represents the percentage of the CPU used by the analyzed process.

To establish a reliable $\mathcal{W}(p)$ function, we comprehend diversified variables that might cause an impact on the obtained metrics. Per [13], several factors might impact the power consumption on a computer besides CPU usage. Despite the processor being highlighted as the main energy consumer according to [7], elements such as fans, memory, and disk usage can also have their share of effects on this topic. Moreover, there have been studies that modeled those variables as energy consumption [20]. Thus, we replicate those tests in machines with different hardware configurations to get feasible results.

When it comes to real-world applications, this paper suggests a model that could be used in many scenarios. As it tracks energy consumption at the process level, the function can estimate how much energy will be consumed by a single application running on the computer. For instance, in the context of cloud pricing, it is common to see the hardware selected by the user as the main definer of how much will be charged [1]. However, with our model, the cloud provider could differentiate how each user requires resources from the machines (by tracking its processes). This could lead to different and more accurate prices for each individual. Besides, other interesting applications might be related to large energy forecast models. As an example, during the instantiation of Virtualized Network Functions (VNFs), a theme of interest has been finding ways to allocate them with energy efficiency criteria [32]. With our model, it is possible to estimate how much energy a VNF will consume given the competition for CPU resources on a certain machine. Hence, this could be used to optimize the placement of these functions to reduce total energy consumption.

The rest of the paper is organized as follows: in Section 2 we explain the problem to which our solution will be applied to, clarifying on which domains our works collaborates with the state-of-the-art. Section 3 discusses the state-of-the-art models for energy consumption regarding CPU usage and delves into theorical definitions for our applications. Section 4 we bring up papers that are related to our work on topics such as resource utilization modeling in virtualized contexts, the influence of hardware on energy consumption, etc. These will be presented as theory background to give the necessary presumptions to set up the experiments and to justify the obtained conclusions. Additionally, Section 5, describes the methodology utilized to develop the experiments utilized for the model construction. Section 6 describes both the experiment environment and its main tools. Moreover, Section 8 showcases the results achieved with those tests graphically and discusses the thought process with each experiment's results. Section 9 concludes the article by summarizing the conclusions obtained from the experiments' results.

2 Problem Definition and Motivation

With this contextualization, describing problems associated with processing resource sharing among processes is imperative. Among those, we highlight the instantiation of VNFs in a network-slicing context. With the development of 5G mobile networks, one of its core components is Network Slicing (NS), which proposes a dynamic provisioning system of networking resources (and functions) to achieve compliance with user needs [29] [5]. In this situation, one relevant theme of research has been the placement of VNFs to achieve high energy efficiency. In this matter, previous work [32] [27] [31] has used models to estimate energy consumption and use this information to apply statistical and algorithmic approaches to provide techniques to optimize the placement of such functions.

Nevertheless, most models do not specify the power usage of a single function but rather analyze the power dissipation of the system as a whole. That is, we cannot analyze the consumption of each VNF and, thus, it becomes harder to estimate the energy consumption of a single slice. Therefore, describing a process's energy consumption within the context of the entire system is essential since it would allow measuring the power dissipation of each of its components.

We focus on assessing how a process's power dissipation behaves when it faces competition for CPU resources. Thus, we describe it as a function $\mathcal{W}(p)$, given that p stands for the competition the process is facing at a certain instant. The behavior of a process's power dissipation is a key factor for VNF placement heuristics and algorithms generating a network topology that guarantees energy efficiency and balance among slices, for instance. Therefore, this showcases the importance of such a description.

In this sense, we proceed by formulating a sample VNF placement problem described as: let $\mathcal{M}$ be a set composed of different machines that would be used to create Network Slices. Then, $\mathcal{S}$ is a set of the created slices and $\mathcal{F}$ is a set of VNFs to be placed on these machines. Thus, each element from $\mathcal{F}$ belongs to a certain element from $\mathcal{S}$, since, in this problem, a VNF composes a fraction of a slice running in a certain machine.

To exemplify such problem, we describe it with Figure 1. In this situation, we have 2 network slices (S1 and S2) built with 3 different machines (M1, M2, M3). Each machine hosts a few network functions (F1, F2, ..., F6, F7), and each of them either belong to S1 or S2, as previously suggested with the problem definition.

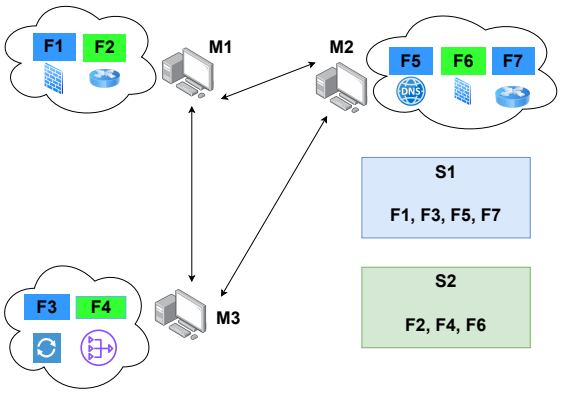

Figure 1: Diagram representing a topology for the NSs

$$\mathcal{M} = [m_1, m_2, ..., m_a] \tag{1}$$

$$\mathcal{S} = [s_1, s_2, ..., s_b] \tag{2}$$

$$\mathcal{F} = [f_1, f_2, ..., f_c] \tag{3}$$

Then, we define the power dissipated by a single VNF as $W(f_j)$, $j \in [1, c]$, the power dissipated by each slice as $P(s_i)$, $i \in [1, b]$, and the power dissipated by whole slice group as $P'(\mathcal{S})$:

$$P(s_i) = \sum_{j=1}^{c} q_{i,j}.W(f_j) \tag{4}$$

$$P'(\mathcal{S}) = \sum_{i=1}^{b} P(s_i) = \sum_{i=1}^{b} \sum_{j=1}^{c} q_{i,j}.W(f_j) \tag{5}$$

$$q_{i,j} = \begin{cases} 1 & \text{if } f_j \in s_i \\ 0 & \text{else} \end{cases} \quad (6)$$

As we wish to minimize the power consumption (4) of all the slices in $\mathcal{S}$, there is an urge to develop a description for $W(f_j)$. Since VNFs could be initially placed in any machine, we suggest the following structure for its power consumption $W(f_j)$, given that p_{mk} represents the competition for resources on a certain machine m_k:

$$W(f_j) = minimum[\mathcal{W}(p_{m1}), \mathcal{W}(p_{m2}), ..., \mathcal{W}(p_{ma})] \quad (7)$$

This means that one can choose which machine m_k to allocate a VNF based on how the competition for resources in each m_k affects the VNF's power consumption. With this construction, the description we found for $\mathcal{W}(p)$ is useful in optimization problems for the placement of VNFs considering the competition for processing resources. Moreover, it would allow a per-slice energy analysis even if s_i shares the same physical machine with other NS. Thus, displaying the potential of our research. With this motivation, this paper aims to answer the following research questions:

- Q1: How to describe Energy Consumption mathematically?
- Q2: How do CPU usage and energy consumption correlate?
- Q3: How to estimate the energy consumption of a process?
- Q4: How does competition for resources affect the energy consumption of a process?

By the end of our research, we expect to prove or refute the following hypothesis: the energy consumption of a constant process is constant, regardless of the competition for resources it faces.

3 Background

Energy is the basic resource needed for performing human activities in the current century, specially in the field of computing. However, its definition is easily misunderstood for that of *Power*, and it is important to differentiate those terms. Energy (E) represents the total work done by a system over a specified period of time (T), as shown in (8), whereas power (P) refers to the rate at which the system performs that work [9]. Therefore, power can be defined as the infinitesimal amount of energy consumed or produced in a very small time interval, as shown in (9). In computer systems, it is common to use power instead of energy when analysing the consumption profile. Since power is an instantaneous quantity, it offers more precise insights into how energy consumption varies dynamically over time.

$$E = \int_0^T P(t)\, dt \quad (8)$$

$$P(t) = \frac{dE(t)}{dt} \quad (9)$$

The energy consumption of a computer derives from several of its components, such as CPU, memory, storage, fans, and network interface card. However, as the CPU consumes the most energy compared to other components, it is common in power modeling to take the energy consumed by the processor as representative of the entire machine's consumption [9] [14] [15]. Work [7] also highlights that the energy consumed by memory and network devices is insignificant compared to that of the CPU.

The relationship between power consumption and CPU usage is a common subject of study in the computing field [9] [12]. In this context, several studies have been conducted over the years to determine what this relationship is. Even though some studies propose that different functions (such as polynomial and non-linear [24] [21]) can characterize the relationship between energy consumption and CPU usage, it is generally accepted that this behavior could be linearly represented. The research in [7] has significantly influenced power modeling for data centers and proposes this linear correlation and also the model (10) to represent it mathematically. In (10), P_u is the power consumption of the server as a function of the CPU usage, u. P_{idle} and P_{max} are constants, the average power consumption when the server is idle and at its maximum capacity, respectively.

$$P_u = (P_{max} - P_{idle})u + P_{idle} \quad (10)$$

Studies such as [15] have shown that this linear model can accurately describe a server's power consumption. Therefore, as a way of simplifying hands-on measurement, this article will consider the CPU as the only component responsible for a machine's energy consumption. We will also consider the relationship between energy consumption and CPU usage as linear, as it is classically considered by literature [9] [12].

However, this state of the art relationship is only representative of the energy consumption of a computer, and cannot be used at process level. The survey [9] extensively reviews power models and finds that only [25] proposes a model for the energy consumption of a process. Moreover, [25]'s model doesn't consider the CPU usage, and does not provide a relationship between that and the energy consumption of a process. Therefore, literature does not provide a model for the relationship of a process' energy consumption and the machine's CPU usage. In this context, this paper aims to empirically find such a model that works at process level.

Virtualization technologies allow multiple activities to be run independently on the same physical machine, facilitating isolated environments that are essential for accurate performance and energy consumption analysis [16]. VMs, a hypervisor-based type of virtualization, provide robust security and isolation at the hardware level [34]. However, the overhead associated with VMs can significantly impact the energy consumption profile of individual processes, thereby complicating precise measurement and analysis [4].

Containers, though not as secure, are light weight and can be rapidly instantiated [4] [34]. They also provide sufficient isolation for most application scenarios and have negligible impact on the host machine's energy consumption [22]. This makes containers particularly suitable for performance isolation and energy consumption research. Furthermore, Kubernetes is an open source system for automating deployment, scaling, and management of containerized applications [19]. The dynamic resource allocation it implements in containerized environments enhances the efficiency and flexibility of resource usage [10]. This further mitigates the overhead concerns typically associated with virtualization.

By using containerization, we can more accurately assess the energy consumption of individual processes, ensuring that the overhead introduced by the virtualization layer remains minimal and

Table 1: Summary of comparison between papers

Research questions	[33]	[13]	[35]	[30]	This work
Q1	✗	✓	✗	✓	✓
Q2	✓	✓	✓	✓	✓
Q3	✗	✗	✗	✗	✓
Q4	✗	✗	✗	✗	✓

does not distort the measurements. Thus, we chose containerization to investigate the energy consumption of a process, benefiting from the low overhead, rapid deployment, and effective resource management capabilities inherent to this technology.

The phenomenon of two or more simultaneous processes competing for computational resources is called "resource competition" [22]. When multiple containers operate on the same physical machine, they compete for processing resources. Thus, there is resource competition associated with processing resource sharing, and it is imperative to learn how that affects the energy consumption of each container. However, the containers energy consumption can be broken down to process level. Therefore, learning the relationship between a process' energy consumption and the resource competition it encounters enables the development of generalized solutions. These solutions would not only be applicable to containerized environments, but could also extend to other process-based applications.

4 Related Work

When looking for work related to ours, our search method consisted of searching the IEEEXplore platform with the following advanced search command, with keywords related to our research: *("All Metadata":energy consumption) AND ("All Metadata":monitoring) AND ("All Metadata":virtualization) AND (("All Metadata":cpu usage) OR ("All Metadata":energy efficiency))*. Filtering the articles from the last 9 years, from 2015 to 2024, we found that [33] and [35] were the resulting papers that related best to our work. Besides, we found other two articles that relate to our work, [13] and [30], when reviewing literature and reading through the survey [9]. This survey is relevant in the Energy Consumption Modeling field, since it thoroughly reviews models for energy consumption in data centers. [9] has 410 citations and is cited by 733 papers.

Work [35] uses test scenarios to assess how the energy consumption of Virtual Machines (VMs) behaves when sharing virtual network infrastructure and CPU cores. Throughout the article, it evaluates how the distribution of network traffic between VMs results in different values of power being dissipated by the machines altogether. Moreover, it also delves into how the competition for processing resources at CPU cores generates different energy consumption for the VM cluster. Nevertheless, unlike our contribution, this paper does not analyze the power consumption at the process level, only from the point of view of the VM.

Besides, [13] and [30] provide a similar analysis of how power consumption correlates to CPU utilization. This correlation is assessed by executing experiments in which they measure the total power consumed by the PC as it faces a gradual increase in CPU utilization over time. Afterward, [13] shows how linear and non-linear empiric functions can describe the behavior of the power dissipated on similar machines. On the other hand, [30] uses a polynomial function to approximate $\mathcal{W}(p)$. Still, differentiating from our work, the first one does not consider different hardware configurations and neither of them assesses process-level energy consumption.

Furthermore, [33] introduces a study into the power consumption behavior of Docker containers. This article investigates how the containers' instantiations create energy overhead and how their power dissipation behaves when facing gradually increasing loads over time in different applications. For example, they run containerized versions of Nginx servers and submit them for a test that increases the number of requests throughout the experiment, establishing a scenario of competition for processing resources related to our work. Even though this paper assesses energy consumption in containers and includes CPU-sharing elements, it does not propose any process-level power representation or model.

Finally, Table 1 presents a summary of the contributions of each paper mentioned alongside our production. Although the CPU use and energy consumption relation is a well-discussed topic in the literature, it remains to be introduced an analysis of process-level energy consumption regarding the total use of the CPU. Hence, this paper proposes a study that complements previous work done on this subject.

5 Proposal

To solve the problem of describing the impact of competition over the energy consumption of a reference process (a VNF, for instance), we propose an empirical method. In this sense, our method consists of running various experiments and observing how the power dissipation of a specific process behaves, by collecting data of certain metrics. In the end, with extensive experimentation and data collection, we can describe the behavior of power dissipation. Accordingly, our method relies on three pillars:

- **Baseline Process**: we establish a baseline process - the baseline process has a constant workload and serves as a reference for the competition.
- **Observed Metrics**:
  - resource usage - we observe the resource usage (more specifically, CPU) of the machine and of the baseline process. This allows us to measure the intensity of competition, through its resource usage; and
  - power consumption per process - we gather data on the power consumption of the baseline process.
- **Experimentation Scenarios**: we define scenarios in which we change variables, such as hardware (experiments are reproduced in different machines with different hardware); active cores (within the same machine); and competition

growth (e.g., none, gradual increase). This allows us to investigate how each of these variables affects the observed metrics.

6 Experiments environment

Machine name	CPU	Total RAM available	Disk Space available	Number of Threads
Controller	Intel(R) Core(TM) i7-4770	7.67 GB	1 TB	8
Worker 1	Intel(R) Core(TM) i5-3330	15.5 GB	1 TB	4
Worker 2	Intel(R) Core(TM) i7-2600	15.5 GB	500 GB	8
Worker 3	Intel(R) Core(TM) i7-2600	7.63 GB	500 GB	8
Worker 4	Intel(R) Core(TM) i5-8500	3.65 GB	1 TB	6
Worker 5	Intel(R) Core(TM) i7-2600	7.63 GB	500 GB	8

Table 2: Machines configurations

To correlate CPU competition usage and process energy consumption, it is necessary to execute tests to attempt to fit the obtained data into a mathematical model that represents this correlation. To achieve this, we created an environment in which we could carry out these experiments. In this context, we assembled 6 machines (hardware configurations in the table 2) and used them to create a Kubernetes Cluster (Client version 1.28.1 and Server version 1.28.6).

Creating a cluster allows automating tests on multiple machines with centralized telemetry capabilities. In this sense, we deploy pods running Docker images with specific test routines so that they can be run as pods on each computer. When it comes to this cluster architecture, the machines are labeled as Controller or Worker. The Controller is responsible for managing the cluster. That is, it makes sure the requirements defined by the user (such as scheduling pods and guaranteeing connection between nodes) are met. On the other hand, the Worker nodes are responsible for hosting the pods running the test routines. To execute the tests and fetch the data obtained from them, we created a four-agent scheme: Scaphandre pods to extract energy information from the Worker nodes, a Prometheus agent so that this data could be exported, a Grafana agent to plot graphs with the exported data, and a Python client to interact with the Controller API.

Then, a tool with process-level power telemetry is needed to gather the goal data. Even though there have been previous software that provided these capabilities [11] [8], Scaphandre [26] stands out due to its natural compatibility with Kubernetes and Prometheus, making it more suitable for our distributed measurements. By using the Powercap RAPL sensors integrated into Intel CPUs (which have been previously validated and used to assess its power consumption

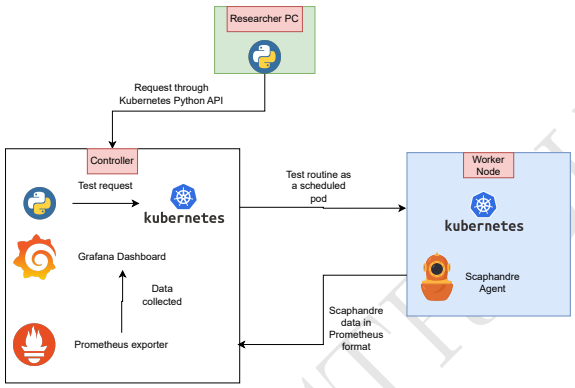

Figure 2: Diagram representing the test environment and its main agents

[23]), it can track the power dissipated by the host as a whole and estimate energy consumption by processes. Thus, by selecting a process on the machine (by tracking its PID, for instance) the agent can estimate how much energy the chosen element consumes, enabling the aforementioned process-level analysis.

Moreover, the Prometheus and Grafana agents act together to obtain this power information and expose them graphically on time-series dashboards (over the test duration). Finally, the Python client consists of a tool to interact with the controller through the Kubernetes API. Through this interaction, we could run tests remotely by authenticating and deploying the test pods.

7 Baseline

In order to validate our method, we ran two initial experiments. In both, we analyzed the power consumption of a baseline process, but in different scenarios. Through the command-line tool for Kubernetes, kubectl [18], we were able to use the Stress tool [3] to generate 2 constant loads, that is, the baseline process and its competition. These two tests were run on Worker 1. Scaphandre scraped power consumption data, that was stored in Prometheus, and Grafana was used to export it to a .csv file.

The first scenario was of no competition for resources, that is, with only the baseline process consuming resources (in this case, CPU). We found that its power consumption (shown in Figure 3) varies from 8.7 to 9.8 Watts, which is a 11.9% variation of the average value, proving that the consumption of the baseline process is in fact constant. Then, the second scenario consisted of running another process alongside the baseline, and identical to it. After some time, the second process was deactivated, enabling comparison between the first scenario (no competition) and the second (with competition). In Figure 4, the sample on the left side, with an average value of 12.5 Watts, corresponds to Scenario 2, and the sample on the right, with an average of 9.75 Watts, corresponds to Scenario 1.

This proves that our method enables us to describe the impact competition has on the energy consumption of a reference process, through empirical means. Furthermore, this raises the possibility

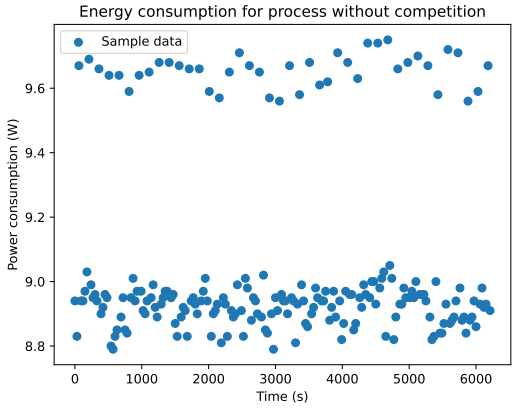

Figure 3: Power consumption of a constant process

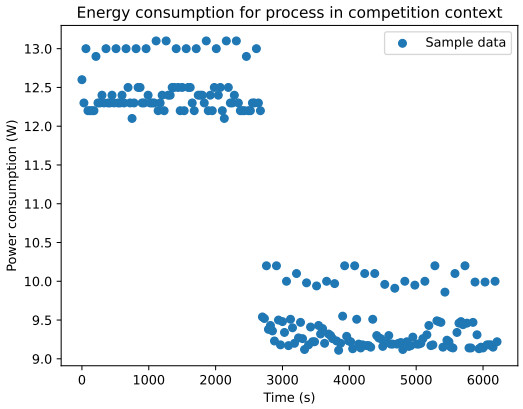

Figure 4: Power consumption with and without competition

that competition increases the energy consumption of a baseline process, contradicting our initial hypothesis that the energy consumption of a constant process is constant. With this motivation, we ran more experiments in order to study the behavior of a process' energy consumption as it faces resource competition.

8 Experiments and Results

Figure 5 displays sequentially the tests routine. As Section 6 described, we initially code the test routine and, with the communication with the Kubernetes Python API, the Controller is able to fetch the necessary information to set up the test. In this situation, the API, through kubectl commands, is able to create the necessary jobs and services to run the test. Afterwards, the Kubernetes agents on the Controller and Worker nodes are able to set up the pods with the test scripts. Besides, they also communicate in order to get the energy data collected by Scaphandre in the Worker running the routine. These sets are sent to the Controller, which stores them with the Prometheus volume. After the test is finished, the data stored in Prometheus is then compiled in a .csv file by the API, which is sent to the Researcher PC. The information is filtered and processed with statistical analysis, giving the final results which are discussed in this section.

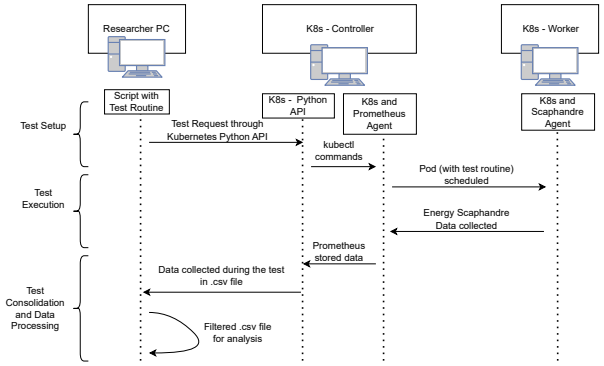

Figure 5: Sequential Diagram describing the test routine

To assess the competition effects on CPU energy consumption, we used both the CpuLimit [2] and Stress [3] tools to generate processing loads and test routines. While the Stress tool generated the load itself (such as running a C script in a loop), CpuLimit was responsible for controlling the load to achieve the expected percentage of usage required for the investigation.

In regards to the test routine, their main structure consisted of running a main process with a constant CPU load and gradually increasing the competition by instantiating smaller processes over time. During the experiment, the main process is monitored and its power consumption is exposed in Grafana dashboards. Then, the plotted data was analyzed statistically to evaluate whether the data would fit in a specific model for $\mathcal{W}(p)$.

8.1 Resource Competition Experiment - Gradual Increase

Initially, we proposed the previously detailed routine: the main constant process facing escalating competition for CPU resources while getting its energy measured. In our investigation, we propose that the competition starts at 0% and increases 5% at a time every 6 minutes (to gather enough data targeting reducing the effects of outliers), until the total processor usage reaches approximately 100%. These cycles were repeated 8 times to reduce the effect of possible outliers and the obtained data was saved in a CSV file. The obtained data was saved and published at Zenodo for public access[6]. Then, by using a Python program that receives a math model from the user (such as linear, quadratic, cubic, etc.), we fit the data into the model and evaluated its correlation by using a t-test. To get started, we executed this experiment on all Worker machines.

After plotting the data, we proposed that the main process behavior could be represented by either a linear or a n-root function. For Workers 1 and 4 (Figure 6 and Figure 8, respectively), the result was better described by a linear profile. On the other hand, Workers 2 (Figure 7), 3 (Figure 8), and 5 (Figure 10) were more accurately pictured by a n-root function. Besides, it is known by Table 2 that

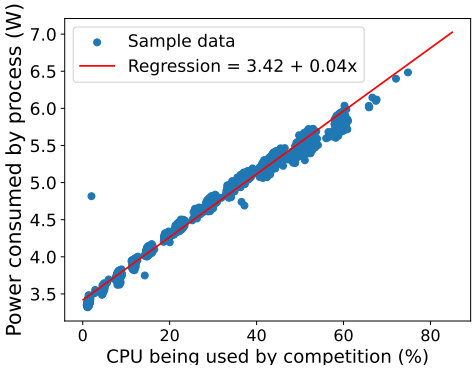

Figure 6: Worker 1 results

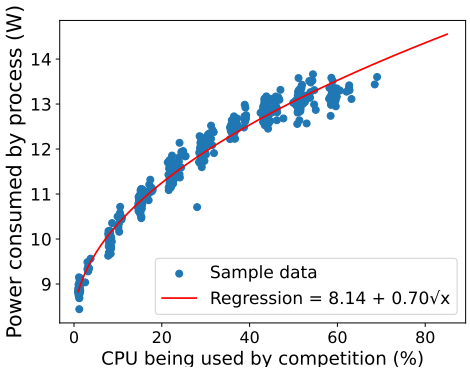

Figure 7: Worker 2 results

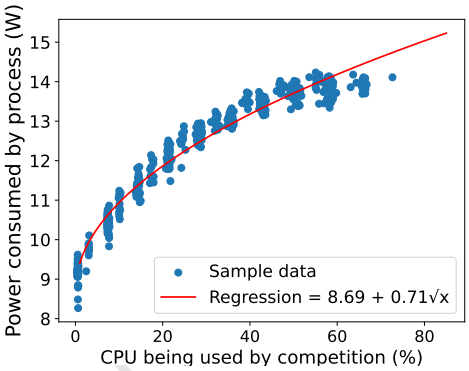

Figure 8: Worker 3 results

Workers 2, 3, and 5 have the same number of virtual cores on their CPUs (8), whilst Workers 1 and 4 have a lower number (4 and 6, respectively). As there has been previous studies which proposed a correlation between number of active threads and energy usage [17], we investigate the relationship between the number of virtual cores and the power consumption profile.

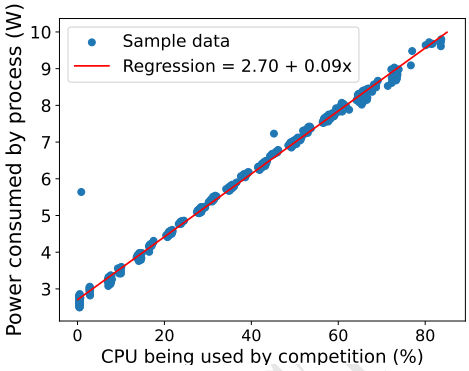

Figure 9: Worker 4 results

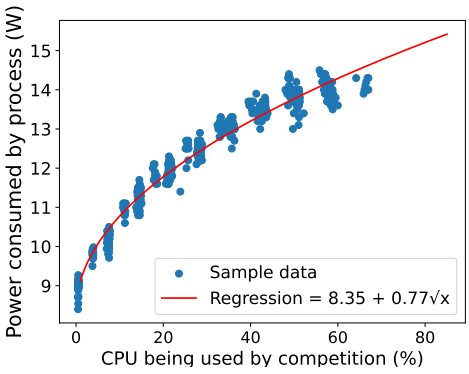

Figure 10: Worker 5 results

8.2 Resource Competition Experiment - Gradual Increase with capped CPU

Then, we suggested executing the same tests on 8-thread worker machines to investigate the effect of vCPUs. However, before running them, we would deactivate some of the cores to simulate a 4 or 6-thread computer and see if the function profile becomes linear. The resulting dataset from these experiments is also publicly available in Zenodo [6].

When reducing the number of threads from 8 to 4 for the machines Worker 2 (Figure 11), 3 (Figure 12), and 5 (figure 13), we got a linear profile for $\mathcal{W}(p)$, just like the results for Worker 1 (the machine with 4 cores originally). On the other hand, the results for capping the resources to 6 threads were mixed.

We executed this version of the test on Worker 2 and 3, and then, after gathering the data, we found out, as displayed in Figures 15 for Worker 2 and 16 for 3, that the best model for fitting this information was the n-root (differentiating from the original 6-core machine - Worker 4 - that provided a linear profile). However, we also did linear fits and the t-statistics were not significantly different than with the n-root model (even though they were slightly worse). This is graphically showcased by Figure 14 for Worker 2 and by Figure 17 for Worker 3. In this scenario, we propose that there is a correlation between the number of virtual cores and the profile for $\mathcal{W}(p)$.

8.3 Analysis & Discussion

With the obtained data, it is possible to draw conclusions related to the $\mathcal{W}(p)$ profile and the machine's CPU. First of all, for all machines with an n-root behavior, it is worth noting that not only was the function's type to represent the power consumption the same, but also the parameters were very similar (that is, a and b on $f(x) = a + b\sqrt[n]{x}$). On Workers 2,3 and 5, the variation in a did not surpass 7%, while for b, the maximum difference reached the 10% threshold. With that, as these machines had significant differences in available RAM and disk space, it seems that the CPU type overrules any of these hardware variations since their function $\mathcal{W}(p)$ had approximately the same parameters. This fact confirms the affirmation made in previous studies [7] [9] that the CPU can be considered the most relevant source of energy consumption on a machine. Moreover, regarding the thread's discussion, we propose a correlation between the function profile and the number of virtual cores available. In this case, we suggest that for lower numbers of cores, the $\mathcal{W}(p)$ behavior is linear (such as the results for Worker 1 and capped results for Workers 2,3 and 5 suggest). As the number of these components increases, they gradually progress to a n-root behavior as they reach 6 cores (Worker 4 provided a linear pattern, whilst capped versions of Workers 2 and 3 had a mixed shape). Then, as it reaches higher quantities of cores (up to 8, on our experimentation), the profile shifts completely to an n-root function (as Workers 2,3 and 5 results propose). Following this line, it is possible to assume that most modern hardware would follow the latter version of $\mathcal{W}(p)$.

A conclusion taken from this is that the power consumption of a process behaves differently on each machine. Then, when deciding where to instantiate a process (regarding attempting to use less energy), the fact that some machines have a linear or n-root profile brings intricacies when choosing the PC in which the process will dissipate less power. To understand them, we shall propose an explanation of the possible $\mathcal{W}(p)$ derivatives. When taking the linear version of the function ($\mathcal{W}_{lin}(p) = a + bp$), the derivative returns:

$$\dot{\mathcal{W}}_{lin} = \frac{d\mathcal{W}_{lin}(p)}{dp} = b \tag{11}$$

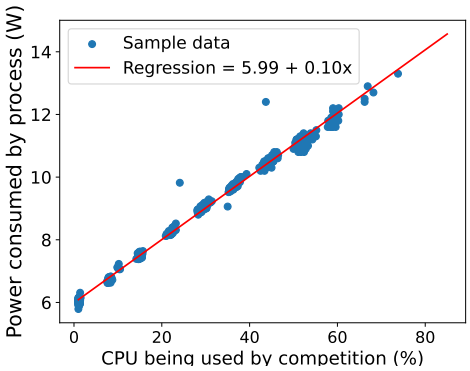

Figure 11: Worker 2 results with 4 cores

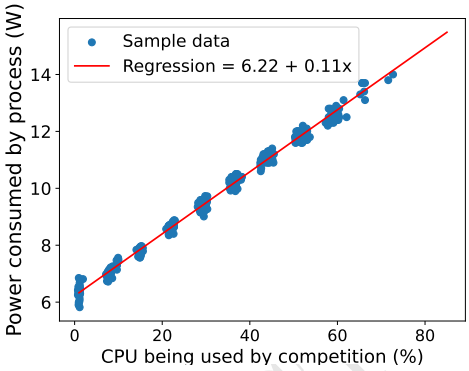

Figure 12: Worker 3 results with 4 cores

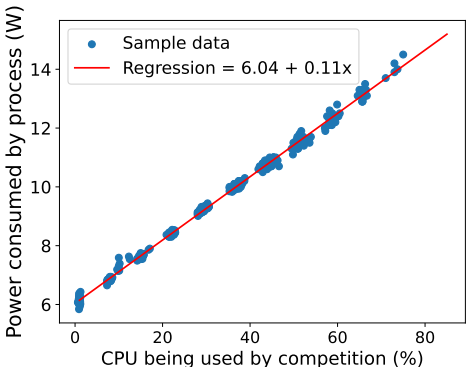

Figure 13: Worker 5 results with 4 cores

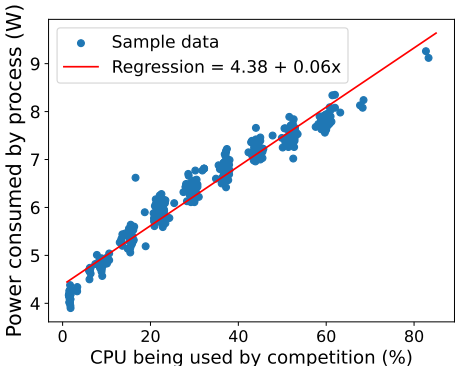

Figure 14: Worker 2 results with 6 cores and linear fit

Given that b is constant, the function increases at a constant rate. That is, *independent of how much CPU is being used by competition processes, the process energy consumption will increase with the same intensity.* On the other hand, with the n-root model ($\mathcal{W}_{rt}(p) = c + d\sqrt[n]{p}$), the derivative is:

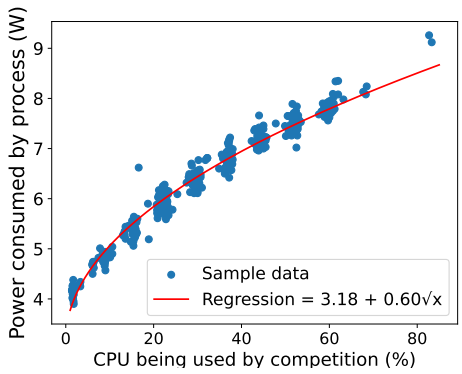

Figure 15: Worker 2 results with 6 cores and n-root fit

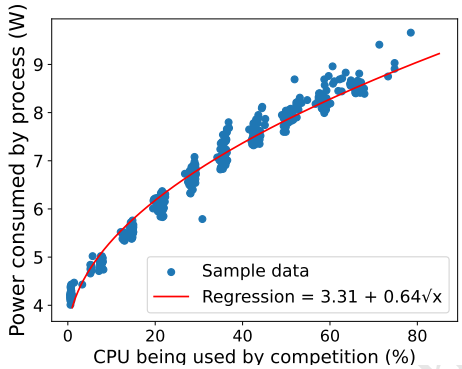

Figure 16: Worker 3 results with 6 cores and n-root fit

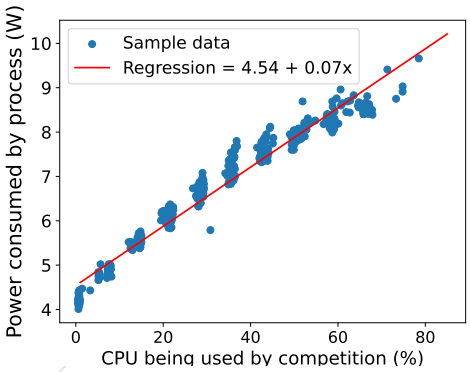

Figure 17: Worker 3 results with 6 cores and linear fit

$$\dot{\mathcal{W}}_{rt} = \frac{d\mathcal{W}_{rt}(p)}{dp} = \frac{d}{np^{1-\frac{1}{n}}} \tag{12}$$

In this case, the derivative depends on p. Moreover, it depends in a way that *for lower values for p, the derivative is greater whilst, for higher values, the derivative becomes tinier.* To exemplify the relevance of these different behaviors, we propose a hypothetical scenario. We assume that, for a certain process, $\mathcal{W}_{lin}(p_i) < \mathcal{W}_{rt}(p_i)$ for a certain $p_i \in [0, 100 - q[$ and let $D(p) = \mathcal{W}_{lin}(p) - \mathcal{W}_{rt}(p)$. Therefore, this shows $D(p_i) < 0$ and that means *it is more energetically friendly to deploy the process on the machine with the linear profile.* However, the derivative of $D(p)$ shows that (assume $k = 1 - \frac{1}{n}$):

$$\dot{D} = \frac{dD(p)}{dp} = b - \frac{d}{np^k} \tag{13}$$

$$\forall p > \sqrt[k]{\frac{d}{nb}}, \dot{D} > 0 \tag{14}$$

$$\therefore \exists\, p_j > p_i \,|\, D(p_j) = 0 \tag{15}$$

Thus, conclusion 15 provides scenario ($p > p_j$) in which $D(p) > 0$. Therefore, if $p_j < 100 - q$, there will be an interval for p (that is, $]p_j, 100 - q]$) for the competition *in which it will be better (energy-wise) to use the machine with the n-root behavior rather than the linear one for the same p.*

This conclusion could be useful for virtualization scheduling scenarios. As previously detailed, when deciding the network topology for the instantiation of NS, the agent responsible for selecting the machine could use the $\mathcal{W}(p)$ models to assess which machine would receive the user's process in question. Alongside performance metrics, the agent would be able to balance performance and energy consumption (with the model) to achieve the necessary Quality of Service without hurting the environment as much.

Besides, such results could be used in more generalized virtualization scenarios, such as pricing the utilization of cloud resources. Instead of considering only the type of hardware that is being used, cloud companies could also assess how much energy the user's processes are consuming. In this case, enterprises could more accurately evaluate how many resources are being consumed by the user, which would lead to a value that would represent the usage better than only considering hardware.

9 Considerations & Future work

In the resource competition context, we evaluated the state of the art. We learned that the relationship between energy consumption and CPU usage at the process level was not yet consolidated. Thus, we applied an empirical method that, given its results and analysis, successfully establishes a function $\mathcal{W}(p)$ to represent the energy consumed by a process.

The conducted experiments showed that the profile of a process' energy consumption as a function of the CPU used by the competing processes is dependent on the CPU's number of virtual cores. When there are 4 threads, that relationship is linear. For 6 cores, a transition between linear and n-root is seen. Then, when 8 threads are available, the behavior seen is of an n-root. Our experiments suggest that, for a higher number of threads, the n-root behavior remains.

In this study, we observe that a lower number of cores means a linear relationship between energy consumption and competition CPU usage, meaning the rate of energy consumption growth matches the rate of CPU usage growth. However, a higher number

of cores leads to a n-root behavior for that relationship. This means that at low levels of CPU usage, energy consumption grows at a higher rate, whereas at high levels of CPU usage, energy consumption grows at a lower rate. This leads to the conclusion that, when the competition is at low CPU usage, it is less energy-consuming to run a process on a machine with fewer cores. On the other hand, when the competition is at high CPU usage, it consumes less energy to run a process on a machine with more cores. These results significantly impact resource allocation and pricing in cloud services, as competition should be a factor considered when allocating new resources to minimize energy consumption and when charging users.

In future work, we aim to extend our analysis to machines with a higher number of CPU threads to further validate the n-root behavior observed. Additionally, extending our research to include VMs would be highly beneficial. By measuring and modeling the energy consumption profile of a process as a function of the competition it faces within VMs, we could uncover new insights that enhance our understanding of resource management, or pricing, in virtualized environments.

Acknowledgments

The authors would like to thank FAPESP for its cooperation through the thematic project SFI2 - Slicing Future Internet Infrastructures (2018/23097-3 - MCTIC/CGI) and for Scientific Initiation proposals 2023/13381-4 and 2023/13383-7 for the authors. Besides, we also acknowledge the National Council for Scientific and Technological Development (CNPQ) for providing through 140303/2021-9. Furthermore, we also thank UDESC and the LabP2D laboratory for the partnership.

References

[1] [n. d.]. Amazon pricing for VMs. https://aws.amazon.com/pt/ec2/pricing/on-demand/. Accessed: 2025-12-16.

[2] [n. d.]. CpuLimit repository. https://github.com/opsengine/cpulimit. Accessed: 2024-07-11.

[3] [n. d.]. Stress tool description. https://man.archlinux.org/man/stress.1. Accessed: 2024-07-11.

[4] Vaibhav Aggarwal and B. Thangaraju. 2020. Performance Analysis of Virtualisation Technologies in NFV and Edge Deployments. In *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. 1–5. doi:10.1109/CONECCT50063.2020.9198367

[5] Patrick Kwadwo Agyapong, Mikio Iwamura, Dirk Staehle, Wolfgang Kiess, and Anass Benjebbour. 2014. Design considerations for a 5G network architecture. *IEEE Communications Magazine* 52, 11 (2014), 65–75. doi:10.1109/MCOM.2014.6957145

[6] Campos. 2024. *Test results for Competition for CPU resources experiments*. doi:10.5281/zenodo.13221177

[7] Jeffrey Chase, Darrell Anderson, Prachi Thakar, Amin Vahdat, and Ronald Doyle. 2001. Managing Energy and Server Resources in Hosting Centres. *Operating Systems Review - SIGOPS* 35, 103–116. doi:10.1145/502034.502045

[8] Hui Chen, Youhuizi Li, and Weisong Shi. 2012. Fine-grained power management using process-level profiling. *Sustainable Computing: Informatics and Systems* 2, 1 (2012), 33–42. doi:10.1016/j.suscom.2012.01.002

[9] Miyuru Dayarathna, Yonggang Wen, and Rui Fan. 2016. Data Center Energy Consumption Modeling: A Survey. *IEEE Communications Surveys & Tutorials* 18, 1 (2016), 732–794. doi:10.1109/COMST.2015.2481183

[10] Zhijun Ding, Song Wang, and Changjun Jiang. 2023. Kubernetes-Oriented Microservice Placement With Dynamic Resource Allocation. *IEEE Transactions on Cloud Computing* 11, 2 (2023), 1777–1793. doi:10.1109/TCC.2022.3161900

[11] Thanh Do, Suhib Rawshdeh, and Weisong Shi. 2009. pTop: A Process-level Power Profiling Tool.

[12] Meryeme El Yadari, Ali Yahyaouy, Stéphane Le Masson, Khalid El Fazazy, and Hamid Gualous. 2022. Study of the correlation between server resources utilization and energy consumption. In *2022 10th International Conference on Systems and Control (ICSC)*. 125–130. doi:10.1109/ICSC57768.2022.9993950

[13] Xiaobo Fan, Wolf-Dietrich Weber, and Luiz Andre Barroso. 2007. Power provisioning for a warehouse-sized computer. *SIGARCH Comput. Archit. News* 35, 2 (jun 2007), 13–23. doi:10.1145/1273440.1250665

[14] Xiaobo Fan, Wolf-Dietrich Weber, and Luiz Andre Barroso. 2007. Power provisioning for a warehouse-sized computer. In *Proceedings of the 34th Annual International Symposium on Computer Architecture* (San Diego, California, USA) *(ISCA '07)*. Association for Computing Machinery, New York, NY, USA, 13–23. doi:10.1145/1250662.1250665

[15] Vishal Gupta, Ripal Nathuji, and Karsten Schwan. 2011. An analysis of power reduction in datacenters using heterogeneous chip multiprocessors. *SIGMETRICS Perform. Eval. Rev.* 39, 3 (dec 2011), 87–91. doi:10.1145/2160803.2160867

[16] Nancy Jain and Sakshi Choudhary. 2016. Overview of virtualization in cloud computing. In *2016 Symposium on Colossal Data Analysis and Networking (CDAN)*. 1–4. doi:10.1109/CDAN.2016.7570950

[17] Hiroki Kataoka, Dilawaer Duolikun, Tomoya Enokido, and Makoto Takizawa. 2015. Power Consumption and Computation Models of a Server with a Multi-core CPU and Experiments. In *2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*. 217–222. doi:10.1109/WAINA.2015.127

[18] Kubernetes 2024. Command line tool (kubectl). https://kubernetes.io/docs/reference/kubectl/. [Online; accessed 29-October-2024].

[19] Kubernetes 2024. Production-grade container orchestration. https://kubernetes.io/. [Online; accessed 23-July-2024].

[20] Adam Lewis, Soumik Ghosh, and Nian Feng Tzeng. 2008. Run-time Energy Consumption Estimation Based on Workload in Server Systems. *HotPower*.

[21] Hui li, Giuliano Casale, and Tariq Ellahi. 2010. SLA-driven planning and optimization of enterprise applications. *WOSP/SIPEW'10 - Proceedings of the 1st Joint WOSP/SIPEW International Conference on Performance Engineering*, 117–128. doi:10.1145/1712605.1712625

[22] Youhuizi Li, Jiancheng Zhang, Congfeng Jiang, Jian Wan, and Zujie Ren. 2019. PINE: Optimizing Performance Isolation in Container Environments. *IEEE Access* 7 (2019), 30410–30422. doi:10.1109/ACCESS.2019.2900451

[23] Unai Lopez-Novoa. 2019. Exploring Performance and Energy Consumption Differences between Recent Intel Processors. In *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. 263–267. doi:10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00088

[24] Xiaoyan Ma, Changgeng Zhang, Shurong Li, and Xinsheng Yang. 2021. A Data-Driven Based Energy Consumption Modeling for Heterogeneous Servers in Data Centers. In *2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2)*. 3109–3114. doi:10.1109/EI252483.2021.9713131

[25] Alexander Nowak, Tobias Binz, Frank Leymann, and Nicolas Urbach. 2013. Determining Power Consumption of Business Processes and Their Activities to Enable Green Business Process Reengineering. In *2013 17th IEEE International Enterprise Distributed Object Computing Conference*. 259–266. doi:10.1109/EDOC.2013.36

[26] Benoit Petit. 2023. *Scaphandre*. https://github.com/hubblo-org/scaphandre.

[27] Chuan Pham, Nguyen H. Tran, Shaolei Ren, Walid Saad, and Choong Seon Hong. 2020. Traffic-Aware and Energy-Efficient vNF Placement for Service Chaining: Joint Sampling and Matching Approach. *IEEE Transactions on Services Computing* 13, 1 (2020), 172–185. doi:10.1109/TSC.2017.2671867

[28] Meikel Poess and Raghunath Othayoth Nambiar. 2008. Energy cost, the key challenge of today's data centers: a power consumption analysis of TPC-C results. *Proc. VLDB Endow.* 1, 2 (Aug. 2008), 1229–1240. doi:10.14778/1454159.1454162

[29] Peter Rost, Ignacio Berberana, Andreas Maeder, Henning Paul, Vinay Suryaprakash, Matthew Valenti, Dirk Wübben, Armin Dekorsy, and Gerhard Fettweis. 2015. Benefits and challenges of virtualization in 5G radio access networks. *IEEE Communications Magazine* 53, 12 (2015), 75–82. doi:10.1109/MCOM.2015.7355588

[30] Cheng-Jen Tang and Miau-Ru Dai. 2011. Dynamic computing resource adjustment for enhancing energy efficiency of cloud service data centers. In *2011 IEEE/SICE International Symposium on System Integration (SII)*. 1159–1164. doi:10.1109/SII.2011.6147613

[31] Amir Varasteh, Basavaraj Madiwalar, Amaury Van Bemten, Wolfgang Kellerer, and Carmen Mas-Machuca. 2021. Holu: Power-Aware and Delay-Constrained VNF Placement and Chaining. *IEEE Transactions on Network and Service Management* 18, 2 (2021), 1524–1539. doi:10.1109/TNSM.2021.3055693

[32] Zihao Wang, Lei Zhuang, Feijie Zhou, and Ruimin Wang. 2023. Energy Efficient VNF Placement Algorithm Using Reinforcement Learning in NFV-Enabled Network. In *2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT)*. 625–629. doi:10.1109/ICCECT57938.2023.10141342

[33] Mehul Warade, Kevin Lee, Chathurika Ranaweera, and Jean-Guy Schneider. 2023. Monitoring the Energy Consumption of Docker Containers. In *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)*. 1703–1710. doi:10.1109/COMPSAC57700.2023.00263

[34] Junzo Watada, Arunava Roy, Ruturaj Kadikar, Hoang Pham, and Bing Xu. 2019. Emerging Trends, Techniques and Open Issues of Containerization: A Review.

IEEE Access 7 (2019), 152443–152472. doi:10.1109/ACCESS.2019.2945930

[35] Chi Xu, Ziyang Zhao, Haiyang Wang, Ryan Shea, and Jiangchuan Liu. 2017. Energy Efficiency of Cloud Virtual Machines: From Traffic Pattern and CPU Affinity Perspectives. *IEEE Systems Journal* 11, 2 (2017), 835–845. doi:10.1109/JSYST.2015.2429731

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009

PHIOT: A Spatio-Temporal Behaviour Modeling Framework for Phishing Detection in IoT Networks

Swagatika Sahoo
Department of Computer Science and Operations Research
University of Montreal
Canada
swagatika.sahoo@umontreal.ca

Nathan Cormerais
Department of Computer Science and Operations Research
University of Montreal
Canada
nathan.cormerais@umontreal.ca

Abdelhakim Senhaji Hafid
Department of Computer Science and Operations Research
University of Montreal
Canada
ahafid@iro.umontreal.ca

Abstract

The rapid growth of Internet of Things (IoT) deployments has significantly expanded the attack surface of cyber–physical systems, making them increasingly vulnerable to phishing attacks that exploit compromised or impersonated devices. Detecting such attacks is particularly challenging due to sparse, noisy, and heterogeneous behavioural data, as well as the ability of adversaries to mimic legitimate device activity. Existing detection approaches often rely on static features, isolated device analysis, or short-term observations, limiting their effectiveness against stealthy and slow-evolving phishing behaviour. In this paper, we propose PHIOT, a behaviour-based phishing detection framework tailored for IoT environments. PHIOT models multivariate sensor data as temporal sequences and learns latent representations of normal behaviour using an LSTM-based autoencoder trained exclusively on benign activity. Rather than relying on reconstruction error, anomaly detection is performed directly in the latent space by measuring the Mahalanobis distance to the distribution of normal embeddings, enabling the identification of subtle behavioural deviations. To address data sparsity and improve generalization, we introduce a targeted data augmentation strategy that generates diverse yet semantically consistent normal behaviour sequences. We evaluate PHIOT in an unsupervised anomaly detection setting, training exclusively on benign smart home behaviour and identifying phishing actors as deviations from normal activity. Experimental results demonstrate that combining latent-space modeling with Mahalanobis-based anomaly scoring significantly improves detection performance, achieving perfect specificity and high precision while detecting over half of phishing sequences. Moreover, PHIOT successfully identifies phishing actors performing atypical and previously unseen activities, highlighting its robustness and generalization capability. These findings suggest that PHIOT provides a promising direction for detecting stealthy phishing attacks in dynamic and sparse IoT environments.

CCS Concepts

• **Security and privacy → Artificial immune systems**.

Keywords

IoT Security, Phishing Detection, Spatio-Temporal Modeling, Anomaly Detection

1 Introduction

The rise of the Internet of Things (IoT) has led to a transformative shift in the way digital systems interact with the physical world. From smart healthcare devices and industrial automation to intelligent homes and critical infrastructure, IoT ecosystems have become integral to modern society. According to recent projections, the number of connected IoT devices is expected to exceed 30 billion by 2030 [1]. While this growth unlocks unprecedented convenience and efficiency, it also opens up new attack surfaces for cyber adversaries.

Among the most concerning threats in this space are *phishing attacks* [6, 8], which aim to deceive users or devices into revealing sensitive credentials or executing unauthorized actions. Unlike traditional phishing in web or email contexts, IoT-based phishing often exploits low-powered devices with limited interfaces, gaining access through forged identities or compromised credentials. Once an IoT device is infiltrated, attackers can impersonate it, issue malicious commands, and even spread laterally within the network leading to disruptions, data breaches, or physical damage in safety-critical systems [2, 9].

Detecting such attacks in IoT networks is non-trivial. Compromised devices often behave similarly to legitimate ones, exhibiting normal communication patterns to avoid detection. The decentralized and heterogeneous nature of IoT systems, combined with intermittent activity of the device due to power saving or event-based communication, results in sparse and noisy behavioural data. This makes it extremely difficult to distinguish phishing-induced behaviour from genuine operations. Traditional detection strategies that focus only on the behaviour of a single device are often insufficient, especially when attackers deliberately hide within normal communication patterns to avoid detection.

Although numerous approaches leverage machine learning and deep learning for phishing detection [3–5, 10], they have not considered the sequential structure and temporal dependencies that characterize the behaviour of IoT devices. Yet, this temporal dimension is critical, as phishing activity often unfolds gradually over time and may only become apparent when analyzing behavioural evolution across multiple time windows. Further, due to the high similarity between phishing and non-phishing behaviours especially when adversaries imitate benign devices most models struggle to learn discriminative patterns.

To overcome these challenges, this work proposes PHIOT, a behavior-based phishing detection framework specifically designed for IoT environments. PHIOT leverages an LSTM-based autoencoder trained exclusively on benign sequences to learn compact

latent representations of normal device behaviour. Instead of relying on reconstruction error, anomaly detection is performed in the latent space using Mahalanobis distance, enabling the identification of subtle deviations that are indicative of phishing activity. To address the scarcity of labeled phishing examples and improve generalization, PHIOT incorporates a targeted data augmentation strategy, generating diverse but semantically consistent normal behaviour sequences. This combination allows zero-shot detection of phishing sequences, including those performing previously unseen or atypical activities.

This dual focus on latent-space modeling and augmented temporal sequences enables PHIOT to capture both subtle and long-term deviations in device behaviour, improving robustness against stealthy and evolving phishing attacks.
Our main contributions are summarized as follows:

- This paper introduces PHIOT, a novel behavior-based phishing detection framework that performs anomaly detection directly in the latent space using Mahalanobis distance rather than traditional reconstruction error metrics. To the best of our knowledge, this is the first study to apply Mahalanobis-based latent space analysis for detecting phishing attacks in smart home IoT environments, effectively addressing the challenge of detecting subtle behavioral deviations that may not produce large reconstruction errors.
- Unlike prior literature, PHIOT addresses the challenge of data sparsity and limited labeled samples through a comprehensive data augmentation strategy combining jitter, scaling, time warp, magnitude warp, and window slice techniques, generating over 200 diverse sequences per activity while preserving temporal and multivariate structure.
- We demonstrate PHIOT's capability for zero-shot phishing detection by training exclusively on normal household activities and successfully identifying previously unseen phishing sequences, including those performing atypical activities such as *praying* that were never observed during training.
- Extensive experiments on a real-world smart home dataset demonstrate that PHIOT achieves superior detection performance, with the combination of data augmentation and hyperparameter tuning resulting in perfect precision (1.000) and specificity (1.000), while achieving a recall of 0.522 and F1-score of 0.686, significantly outperforming classical one-class baselines applied to the same latent representations.

2 PHIOT: Proposed Framework

This section describes the methodological framework developed to detect phishing behaviour in smart environments using temporal modelling of multivariate sensor streams. The proposed approach, named PHIOT, leverages an LSTM based autoencoder trained exclusively on normal household activities and performs anomaly detection by measuring the Mahalanobis distance between latent representations. Figure 1 showcases the overall architecture of the neural network used for this method.

2.1 Problem Statement

Sensor data collected from smart home environments take the form of multivariate time series. For each activity occurrence, the system records a sequence

$$\mathbf{x}_{1:T} = (\mathbf{x}_1, \ldots, \mathbf{x}_T),$$

where each $\mathbf{x}_t \in \mathbb{R}^d$ contains the d sensor readings at time t. These sequences capture the temporal dynamics of interactions between occupants and devices. Since labelled phishing events are rare, the problem is cast as unsupervised anomaly detection.

The goal is to learn a representation of normal behaviour and then determine whether a new sequence deviates from this learned distribution. Instead of relying on reconstruction error, PHIOT performs anomaly detection directly in the latent space of the autoencoder. Let $\mathbf{z} \in \mathbb{R}^k$ denote the latent vector produced by the encoder for sequence $\mathbf{x}_{1:T}$. We compute its Mahalanobis distance to the distribution of latent vectors extracted from normal training data:

$$D_M(\mathbf{z}) = \sqrt{(\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu})},$$

where $\boldsymbol{\mu}$ and Σ are the empirical mean and covariance of normal latent representations. A sequence is classified as anomalous if

$$D_M(\mathbf{z}) > \tau,$$

where τ is a threshold selected using the three sigma rule applied to the distribution of Mahalanobis distances observed on normal data. This formulation allows the detection of subtle behavioural deviations even when they do not produce large reconstruction errors.

2.2 Architecture Overview

PHIOT is protocol-agnostic, as it relies exclusively on physical-layer sensor data rather than network traffic or IoT protocol messages.

Figure 1 illustrates the end-to-end architecture of PHIOT's LSTM-based autoencoder used for temporal behaviour modelling. The model follows an encoder–decoder paradigm designed to capture sequential dependencies in multivariate sensor data while learning a compact latent representation of normal behaviour.

The input layer receives a fixed-length multivariate time series $\mathbf{x}_{1:T} \in \mathbb{R}^{T \times d}$, where each time step corresponds to the synchronized readings of multiple smart home sensors. These sequences are processed by the LSTM encoder, composed of stacked LSTM layers that iteratively model temporal dependencies and long-range correlations across sensor streams. The recurrent structure allows the encoder to retain contextual information over extended time horizons, which is essential for distinguishing subtle behavioural deviations from normal activity patterns.

The final hidden state of the encoder is passed through a fully connected layer that projects the temporal representation into a lower-dimensional latent space. This latent vector serves as a compact embedding summarizing the entire activity sequence. Unlike reconstruction-error-based approaches, PHIOT leverages this latent representation directly for anomaly detection, as it provides a more discriminative and noise-robust characterization of behaviour.

The LSTM decoder mirrors the encoder structure and reconstructs the original input sequence from the latent embedding. A fully connected layer first maps the latent vector back to the

decoder's hidden dimensionality, after which stacked LSTM layers generate the reconstructed sequence $\hat{\mathbf{x}}_{1:T}$. The reconstruction objective encourages the latent space to preserve temporal and multivariate structure while compressing redundant information.

Training is performed exclusively on benign activity sequences, ensuring that the learned latent distribution represents normal behaviour only. During inference, phishing sequences are encoded into the same latent space and evaluated using a Mahalanobis distance-based anomaly score. This design decouples representation learning from anomaly scoring, enabling PHIOT to detect phishing actors whose behaviour deviates from normal patterns even when reconstruction error remains low.

The architecture shown in Figure 1 enables PHIOT to jointly capture temporal dynamics, multivariate correlations, and distributional properties of benign behaviour, forming the foundation for robust zero-shot phishing detection in sparse and heterogeneous IoT environments.

Overall, this model assumes that phishing actors may perform seemingly legitimate or unusual activities to evade rule-based or semantic detectors; therefore, PHIOT focuses on detecting behavioral inconsistency rather than predefined malicious actions.

2.3 Proposed Approach

The PHIOT pipeline consists of four stages: data preprocessing, sequence construction, training of an LSTM autoencoder, and anomaly scoring in latent space.

Data preprocessing. Normal activities are loaded from more than two hundred CSV files generated through synthetic augmentation. All sequences associated with invalid or undefined activity identifiers are removed. For each activity occurrence, rows are grouped under a unique id (`ActGroupID`), and non-feature columns such as timestamps and metadata are discarded. The resulting feature matrices are converted into tensors of shape (T, d), where T varies across activities. Sequences are then truncated or zero-padded to a fixed length of 300 time steps to ensure uniformity. Normalisation is performed using the mean and standard deviation computed exclusively from the training split of normal data; the same parameters are reused for validation and phishing samples.

LSTM autoencoder. PHIOT adopts an autoencoder (see Figure 1) composed of a unidirectional LSTM encoder and LSTM decoder. The encoder processes the sequence using

$$(\mathbf{h}_t, \mathbf{c}_t) = \mathrm{LSTM}(\mathbf{x}_t; \theta),$$

and the final hidden state $\mathbf{h}_T$ is projected through a fully connected layer to produce the latent vector $\mathbf{z}$. The decoder receives $\mathbf{z}$, transforms it back to the hidden dimensionality, and reconstructs the full sequence through a second LSTM followed by a linear projection layer.

The model is trained on normal activity sequences using mean squared reconstruction loss:

$$\mathcal{L} = \sum_{t=1}^{T} \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|_2^2,$$

with the Adam optimiser, learning rate scheduling, and early stopping based on validation loss. Training uses mini-batches of size 128 over a maximum of 100 epochs.

Each model input is a multivariate time series corresponding to a single activity execution, spanning from its start to end time and aggregating synchronized readings from all available environmental sensors.

Latent-space modelling. After training, the encoder is applied to all normal sequences. The latent vectors from both the training and validation sets are concatenated to define the normal latent distribution. Let $\{\mathbf{z}^{(i)}\}_{i=1}^{N}$ denote the set of latent vectors extracted from normal data. The empirical mean and covariance are computed as

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{z}^{(i)}, \qquad \Sigma = \mathrm{Cov}(\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(N)}).$$

After training on benign activity sequences, the latent representations are assumed to follow a multivariate Gaussian distribution characterized by an empirical mean and covariance matrix. For any new activity sequence, its latent embedding is evaluated using the Mahalanobis distance. If this distance exceeds a threshold derived from the three-sigma rule, the sequence is flagged as anomalous and considered a potential threat.

Mahalanobis-based anomaly detection. For each new sequence, PHIOT extracts its latent vector and computes its Mahalanobis distance to the normal latent distribution. A threshold τ is set using the three sigma rule on the normal Mahalanobis scores:

$$\tau = \mathbb{E}[D_M] + 3\,\mathrm{Std}[D_M].$$

Sequences whose distance exceeds the threshold are flagged as anomalous. This approach captures anomalies that preserve low reconstruction error but occupy regions of latent space that are improbable under the normal distribution.

PHIOT does not attempt to semantically label activities as malicious or benign based on their type (e.g., walking, praying, or entering a room). Instead, it models normal behavior dynamics and detects potential threats through statistical deviations from learned benign patterns. This design choice enables zero-shot detection of phishing behaviors that may deliberately mimic legitimate activities.

3 Experiments

This section presents the experimental setup used to evaluate PHIOT, including data preparation, training procedure, anomaly scoring, baseline comparisons, and evaluation methodology. All experiments were performed using PyTorch on a single GPU.

3.1 Dataset and Preprocessing

Experiments rely on the publicly available *Dataset for Cyber–Physical Anomaly Detection in Smart Homes* [7]. This dataset provides multivariate time series recorded from a variety of smart home sensors, capturing daily activities of multiple occupants in controlled environments. The data includes both normal activities and annotated anomalous events, allowing the evaluation of unsupervised anomaly detection methods. In particular, the phishing-like sequences used for testing emulate malicious behaviours such as unauthorized device interactions or command injections.

Normal behaviour sequences were obtained from the original recordings, complemented by a synthetic augmentation procedure

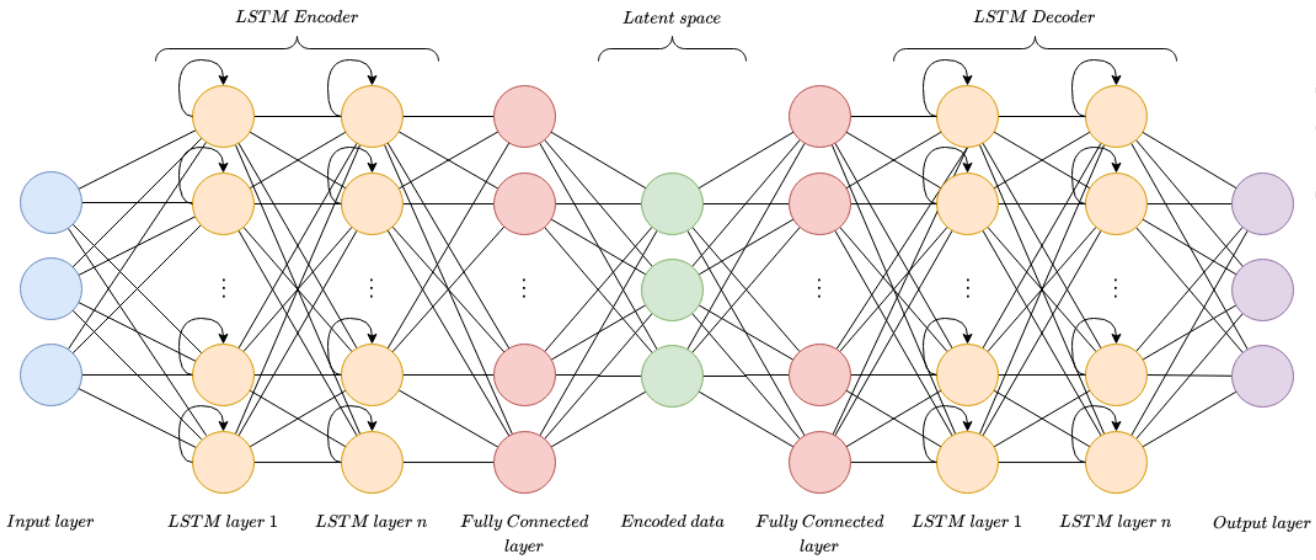

Figure 1: End-to-end architecture of the LSTM autoencoder for behaviour modelling.

that increases diversity and reduces the risk of overfitting. Each activity instance is associated with a unique identifier (`ActGroupID`), which allows all sensor readings corresponding to a single execution to be grouped into one temporal sequence. Non-numerical information such as timestamps, identifiers, or categorical metadata is removed, keeping only the raw sensor values. Since the duration of activities varies, sequences are padded or truncated to a fixed temporal length to ensure uniform input dimensionality.

Table 1 illustrates raw sensor data corresponding to two executions of the activity *Entering the home*. Each execution spans approximately 4 seconds, with timestamped readings collected from multiple sensors. All rows sharing the same `ActID` belong to the same activity type, while distinct `ActGroupID` values differentiate separate executions. For clarity, only a subset of variables (temperature and humidity) from the five sensors is shown; other sensor variables are omitted. The vertical ellipsis indicates that additional activities occurred between the two executions but are not displayed here. The resulting dataset is divided into:

- a training set containing only normal activity sequences;
- a validation set also composed solely of normal sequences, used for monitoring training progress;
- a phishing set used exclusively for testing, containing no overlap with training data, effectively evaluating PHIOT in a zero-shot detection scenario.

Normalization parameters (mean and standard deviation) are computed exclusively from the training set and subsequently applied to all other splits to prevent information leakage.

3.2 Data Augmentation

To increase the diversity of normal behaviour sequences and improve the robustness of PHIOT, we applied a targeted data augmentation procedure to the original dataset [7]. This approach generates additional sequences from each activity while preserving the underlying temporal and multivariate structure of the sensor data.

Data augmentation is employed to expand the manifold of observed benign behaviors by generating realistic temporal variations, thereby reducing the likelihood that rare but benign activities are mistakenly detected as anomalies.

The augmentation pipeline operates on sequences grouped by activity (`ActID`, e.g., Entering the home) and by activity execution (`ActGroupID`). An `ActGroupID` corresponds to a single execution of an activity, where all sensor readings between the activity's start and end timestamps share the same identifier. For each activity, the pipeline generates over 200 augmented sequences. Several complementary techniques are applied to each base sequence:

- **Jitter:** Gaussian noise is added to each sensor channel, scaled by the channel's standard deviation, to simulate small measurement variations.
- **Scaling:** Sensor readings are multiplied by random factors drawn from a normal distribution around 1, introducing variations in signal amplitude.
- **Time Warp:** Sequences are stretched or compressed along the temporal axis to simulate variations in activity speed.
- **Magnitude Warp:** Smooth, nonlinear curves are applied to the sequence to vary the amplitude over time, introducing realistic temporal fluctuations.
- **Window Slice:** Sequences are cropped or interpolated to a uniform length, ensuring consistent input dimensions for model training while also generating partial observations.

During augmentation, multiple techniques are applied randomly to each sequence, producing diverse variations that retain the semantic integrity of the original activity. The resulting augmented dataset increases the effective size of the training set, providing the

Timestamp	Temperature_1	Humidity_1	$\cdots$	Temperature_5	Humidity_5	ActGroupID	ActID
2022-10-20 14:52:07	25.53	26.06	$\cdots$	21.48	23.90	4	1.0
2022-10-20 14:52:08	25.46	26.09	$\cdots$	21.49	23.85	4	1.0
2022-10-20 14:52:09	25.42	26.14	$\cdots$	21.53	23.79	4	1.0
2022-10-20 14:52:10	25.42	26.12	$\cdots$	21.40	23.93	4	1.0
			$\vdots$				
2022-10-21 16:27:42	23.48	26.57	$\cdots$	21.33	23.89	5	1.0
2022-10-21 16:27:43	23.56	26.64	$\cdots$	21.40	23.85	5	1.0
2022-10-21 16:27:44	23.75	26.13	$\cdots$	21.39	23.86	5	1.0
2022-10-21 16:27:45	23.75	26.17	$\cdots$	21.34	23.89	5	1.0

Table 1: Example of raw multivariate sensor data corresponding to two executions of the activity *Entering the home*. The table shows a subset of variables (e.g., temperature and humidity) from five sensors for illustration. Each row is a timestamped sensor snapshot. Distinct `ActGroupID` values denote different activity executions, with intermediate activities omitted for clarity.

LSTM autoencoder with a richer set of normal behaviours. This process is critical for improving PHIOT's ability to distinguish subtle anomalies and to detect phishing sequences, including those performing previously unseen or atypical activities.

The augmented sequences were saved in the same format as the original dataset, ensuring compatibility with the existing preprocessing and model training pipeline. The augmented data was exclusively used for training and validation, while phishing sequences remained unseen until evaluation, enabling a zero-shot assessment of PHIOT's anomaly detection capabilities.

3.3 Training Procedure

The PHIOT model is trained as an LSTM-based autoencoder using only normal behaviour sequences. The training objective is the reconstruction of the input sequence, and optimization is performed using a standard gradient-based technique.

Model hyperparameters were selected via grid search on a validation set containing only benign activity sequences. The search space included LSTM hidden size, latent dimension, dropout rate, learning rate, batch size, sequence length, and padding strategy.

Sequence length and padding strategy were treated as critical hyperparameters due to the large variability in activity durations, as improper temporal alignment can significantly degrade representation quality in recurrent models.

To ensure stable learning, several regularization and training-control mechanisms are employed:

- **Early stopping** prevents overfitting by terminating training when the validation loss stops improving.
- A **learning rate scheduling strategy** gradually lowers the step size when progress plateaus, allowing the model to refine its representation during later stages of training.
- **Mini-batch training** stabilizes gradients over long sequences.
- A lightweight form of **dropout** within the LSTM layers encourages robustness and reduces co-adaptation of recurrent units.

After training, only the encoder is used for generating latent representations required for anomaly scoring.

3.4 Latent-Space Anomaly Scoring with Mahalanobis Distance

Although the autoencoder is trained using reconstruction loss, anomaly detection in PHIOT relies exclusively on the latent representations produced by the encoder. The distribution of normal embeddings is modeled using their empirical mean and covariance, enabling Mahalanobis-based anomaly scoring. This method allows PHIOT to detect subtle deviations that may not lead to large reconstruction errors, effectively leveraging both the expressive power of autoencoders and the statistical properties of the latent space.

The use of Mahalanobis distance is particularly effective in sparse and heterogeneous IoT environments, where traditional reconstruction-based thresholds may fail to distinguish normal variability from truly anomalous behaviour. Moreover, our results demonstrate that this approach allows zero-shot detection of phishing-like sequences not seen during training, highlighting the generalization capacity of PHIOT.

3.5 Baselines

To contextualize the performance of PHIOT, we compare its anomaly detection ability to two classical one-class baselines applied directly to the same encoder-generated embeddings:

- **Isolation Forest**, which identifies anomalies based on tree-based partitioning depth;
- **One-Class SVM** with a radial basis function kernel, which learns a boundary enclosing normal embeddings.

Applying these methods to PHIOT's latent representations ensures that comparisons isolate the anomaly scoring strategy rather than the feature extraction process.

3.6 Evaluation Metrics

Evaluation is performed on all sequences from the augmented dataset. Since the model was trained exclusively on normal sequences, reported metrics for normal (negative) behavior reflect in-distribution performance and may overestimate generalization. Phishing sequences (positive) were not seen during training and are used solely to assess the model's ability to detect previously unseen anomalies. Metrics include:

- **Precision** and **Recall** to quantify false alarm rate and detection sensitivity;
- **F1-score** to capture the balance between precision and recall;
- **Accuracy** and **Specificity** for overall classification behaviour;
- **ROC-AUC** and **PR-AUC** to evaluate separability independent of threshold choice.

Score distributions and confusion matrices are analyzed to examine the separation of normal and phishing sequences, providing a transparent view of PHIOT's discriminative capacity.

4 Results

PHIOT demonstrates strong potential for detecting anomalous activity sequences in smart home environments, particularly in challenging zero-shot scenarios where phishing sequences were never observed during training. Figure 2 reports the performance of each experimental configuration using standard anomaly detection metrics. Accuracy and specificity quantify the model's ability to correctly identify normal behavior, while precision, recall, and F1-score are computed with respect to the phishing (anomalous) class. Bold values indicate the best performance achieved for each metric across all configurations. This comparison highlights the trade-off between maintaining low false-positive rates on normal traffic and effectively detecting phishing sequences.

Data augmentation proves to be a key factor in improving detection performance. By applying the techniques described in Section 2, augmented sequences increase the diversity of normal behaviours available for training, allowing the model to better distinguish subtle deviations associated with phishing activity. This is reflected in the improved recall from 0.217 (non-augmented) to 0.420 (augmented). While augmentation initially reduces precision (0.937 to 0.408) due to a higher number of false positives, hyper-parameter tuning on the augmented dataset restores precision and specificity to perfect values while improving recall to 0.522, resulting in an F1-score of 0.686. These results highlight that combining augmentation with careful model tuning is essential to achieve robust detection in sparse and heterogeneous IoT environments.

Figure 2 shows the Mahalanobis distance of each sequence from the learned normal latent distribution. Distances below the dashed red three-sigma threshold correspond to sequences that are considered normal, while sequences whose distance exceeds this threshold are flagged as anomalous. The green shaded region highlights the range of distances regarded as normal variability under the learned distribution. Sequences appearing above the threshold represent deviations that are statistically unlikely under normal behavior, and are therefore identified as potential phishing activity.

The figure demonstrates that normal sequences predominantly lie below the threshold, while phishing sequences are concentrated above it. This separation in latent space highlights the effectiveness of combining LSTM-based representation learning with Mahalanobis distance for detecting subtle anomalies.

The confusion matrix in Figure 3 complements this visualization, showing that all 2,613 normal sequences are correctly classified (perfect specificity and precision), while 36 out of 69 phishing sequences are correctly identified. The moderate recall indicates that some phishing sequences remain challenging, but the use of latent embeddings with Mahalanobis scoring substantially improves detection compared to naive reconstruction-error approaches.

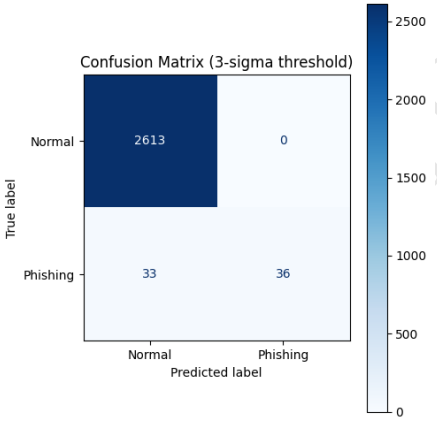

Figure 3: Confusion matrix for the experiment corresponding to using PHIOT, trained on the augmented dataset.

Analysis of individual sequences provides further insight. For instance, PHIOT successfully detected phishing actors performing atypical activities, such as *praying*, which were not observed during training. This zero-shot detection capability demonstrates that the framework generalizes well to previously unseen behaviours and can flag potential security breaches even when malicious actors mimic normal activity in novel ways.

Importantly, experimental results demonstrate that PHIOT adopts a conservative detection strategy. While recall remains moderate, precision and specificity reach 100% in the best configuration, indicating that benign behaviors—including rare or atypical ones—are not falsely classified as phishing. This behavior is desirable in safety-critical smart environments where false positives can be disruptive.

Taken together, the results indicate that the combination of LSTM-based latent representations, Mahalanobis-based anomaly scoring, and targeted data augmentation provides a robust and promising approach for detecting subtle, stealthy, or previously unseen anomalies in smart environments.

5 Conclusion

This paper presented PHIOT, a latent space-based framework for detecting phishing attacks in IoT environments using LSTM autoencoders and Mahalanobis distance anomaly scoring. By modeling temporal dependencies in multivariate sensor data and performing detection in latent space rather than relying on reconstruction error, PHIOT effectively identifies subtle behavioral deviations indicative of phishing activity. While the specific sensor modalities depend on the dataset, the proposed methodology—temporal encoding via LSTM autoencoders followed by latent-space anomaly detection—is general and can be reproduced on other smart environment datasets with aligned sensor streams. As with all anomaly detection systems, it is not possible to guarantee coverage of all conceivable normal behaviors during training. User behavior in smart environments is

Experiment	Accuracy	Precision	Recall	Specificity	F1-score
Augmented + Isolation Forest	96.4%	0.037	0.014	0.990	0.021
Augmented + One-Class SVM	87.8%	0.492	0.472	0.900	0.477
Non-augmented + PHIOT (1)	91.6%	0.937	0.217	0.998	0.352
Augmented + PHIOT (2)	**98.7%**	**1.000**	**0.522**	**1.000**	**0.686**

Table 2: Performance comparison across experimental configurations and baselines.

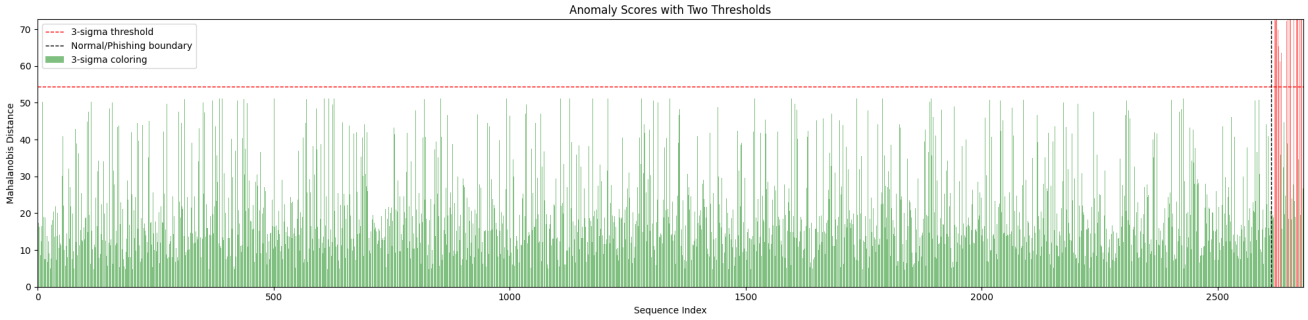

Figure 2: Mahalanobis distances of all sequences with respect to the normal latent distribution.

inherently diverse and may evolve over time. However, our targeted data augmentation strategy successfully addressed the challenge of limited training data by generating over 200 diverse sequences per activity while preserving semantic integrity. Experimental results demonstrated that combining augmentation with careful hyperparameter tuning achieved an F1-score of 0.686 with perfect precision and specificity. Notably, PHIOT successfully detected phishing actors performing previously unseen activities in zero-shot scenarios, validating its generalization capability. Future work will focus on four key directions: (1) incorporating spatial dependencies through graph neural networks to model device interactions and detect coordinated attacks, (2) developing adaptive threshold mechanisms for evolving behavior patterns in long-term deployments, (3) creating lightweight variants suitable for resource-constrained edge devices to enable real-time on-device detection, and (4) incorporating cyber-level interactions such as network traffic alongside physical sensor data for improved robustness and resilience against sophisticated phishing attacks. Additionally, evaluating PHIOT across diverse IoT environments beyond smart homes, including industrial control systems and healthcare settings will assess its broader applicability and scalability for securing heterogeneous cyber-physical systems.

References

[1] 2023. Number of connected IoT devices worldwide. *Statista* (2023). https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/

[2] Utku Akyazi, Oguz Yildirim, and Huseyin Polat. 2022. Phishing attacks in IoT: Threats and countermeasures. In *Proceedings of the 2022 IEEE Symposium on Computers and Communications*. IEEE.

[3] Lázaro Bustio-Martínez, Miguel A. Álvarez-Carmona, Vitali Herrera-Semenets, Claudia Feregrino-Uribe, and René Cumplido. 2022. A lightweight data representation for phishing URLs detection in IoT environments. *Information Sciences* 603 (2022), 42–59. doi:10.1016/j.ins.2022.04.059

[4] Gonzalo De La Torre Parra, Paul Rad, Kim-Kwang Raymond Choo, and Nicole Beebe. 2020. Detecting Internet of Things attacks using distributed deep learning. *Journal of Network and Computer Applications* 163 (2020), 102662. doi:10.1016/j.jnca.2020.102662

[5] S. B. Gopal, C. Poongodi, D. Nanthiya, T. Kirubakaran, D. Logeshwar, and B. Kulavishnu Saravanan. 2022. Autoencoder based Architecture for Mitigating phishing URL attack in the Internet of Things (IoT) using Deep Neural Networks. In *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)*. 427–431. doi:10.1109/ICDCS54290.2022.9780673

[6] Surbhi Gupta, Abhishek Singhal, and Akanksha Kapoor. 2016. A literature survey on social engineering attacks: Phishing attack. In *2016 international conference on computing, communication and automation (ICCCA)*. IEEE, 537–540.

[7] Yasar Majib, Mohammed Alosaimi, Andre Asaturyan, and Charith Perera. 2023. Dataset for cyber–physical anomaly detection in smart homes. *Frontiers in the Internet of Things* Volume 2 - 2023 (2023). doi:10.3389/friot.2023.1275080

[8] Zulfikar Ramzan. 2010. Phishing attacks and countermeasures. *Handbook of information and communication security* (2010), 433–448.

[9] Heena Rathore, Deepak Sharma, and James J. Park. 2018. Real-time intrusion detection system for IoT using customized hybrid model. *Computers & Electrical Engineering* 68 (2018), 157–172.

[10] Ashina Sadiq, Muhammad Anwar, Rizwan A. Butt, Farhan Masud, Muhammad K. Shahzad, Shahid Naseem, and Muhammad Younas. 2021. A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0. *Human Behavior and Emerging Technologies* 3, 5 (2021), 854–864. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/hbe2.301 doi:10.1002/hbe2.301

Autonomous Cargo Box Delivery System

Konstantin Aprosin
aprosin.ki@gmail.com
UFSC
Florianopolis, SC, Brasil

Robert Reis
robertrs959@gmail.com
UFSC
Florianopolis, SC, Brasil

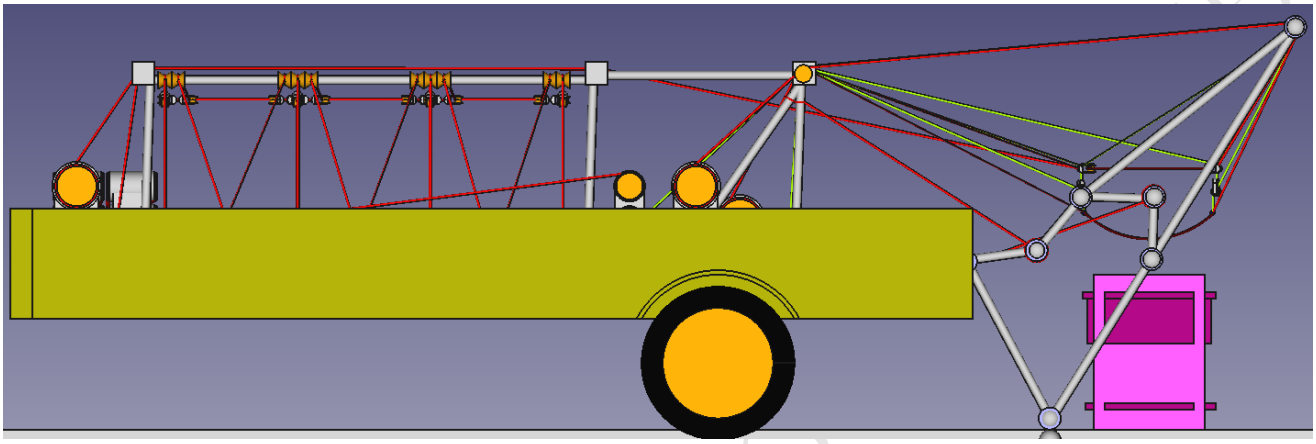

Figure 1: Cargo box loading equipment for an autonomous vehicle

Abstract

This article is devoted to operation process and hardware of box delivery system, that operating by using of autonomous vehicles network. It is considered questions of the operation process in the autonomous vehicles network and questions of cargo boxes downloading and uploading technics. Special attention is paid to loading hardware of autonomous vehicle

Keywords

Autonomous vehicles, vehicles network, box delivery, logistics automation, autonomous delivery system

ACM Reference Format:
Konstantin Aprosin and Robert Reis. 2026. Autonomous Cargo Box Delivery System. In *Proceedings of (International Workshop on ADVANCEs in ICT Infrastructures and Services).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Box delivery services is important part of city infrastructure, that is used as by businesses also by households. Millions people work in this field on over the world. This service is provided by as big international companies also by local delivery operators.

Automation of this service allow to reduce human labor using and free big amount of people from heavy and unqualified duties. This article describes one of ways of the delivery service automation, that is based on using of autonomous vehicles. Presented here autonomous delivery system is potentially able to provide transportation of a cargo box between two geographical points without any human activity.

2 State of art

Present time box delivery services are provided as by specialized companies also by organizations, that performs other duties, but has own transportation departments. Big transnational corporations, with high capitalization [1] are operating in box delivery business, so this area has enough financial resources for automation.

2.1 Autonomous vehicles for box delivery

Last years autonomous vehicles technologies have came to state, when autonomous vehicles are ready for using on common roads [4]. Box delivery application for autonomous vehicles are also considered present time [7],[2]. However it is mainly considered using of specialized vehicles (figure 2). There is no technology, that would use ordinary vehicles for autonomous box delivery.

2.2 Existing cranes and manipulators

For using ordinary vehicles in box delivery tasks it is necessary to have some method for loading boxes to vehicle. It is possible to use specific wheeled drones for it (as shown on figure 2) or it is possible to use existing manipulators (figure 3). Here it is not considered

International Workshop on ADVANCEs in ICT Infrastructures and Services, Florianopolis, SC-Brazil

ACM ISBN 978-1-4503-XXXX-X/2026/03
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Figure 2: Existing solution for box delivery

using some drones for loading of autonomous vehicle, because it is searching for simple and inexpensive solution.

Main problem of existing manipulators is necessity to provide two quality simultaneously. From one hand, the manipulator should be able to lift and carry enough weight, from other hand, the manipulator should be able to high precision positioning to lock the cargo box. Power and high precision mechanics usually are expensive. More over, this complicated mechanics should be tolerant to high vibration in time of transportation on non-ideal roads. So loading system is one of biggest problems for origination of totally autonomous box delivery system.

Figure 3: Existing manipulators for cars

3 Structure of the autonomous delivery system

Any autonomous box delivery system has to parts, physical and virtual. The first one is majority of devises and installations, that is used for transportation of an object from one geographical point to another, the second one is majority of data flows, that provides correct operation of the physical part.

The virtual part of the delivery system is a data transfer process. Where participants of the process start physical processes, using data from the virtual part. In the delivery process participate following agents:

- Delivery customer
- Deliveryman
- Cargo box
- Cargo vehicle
- Delivery process software

Interactions of the agents are presented at the diagram on figure 4

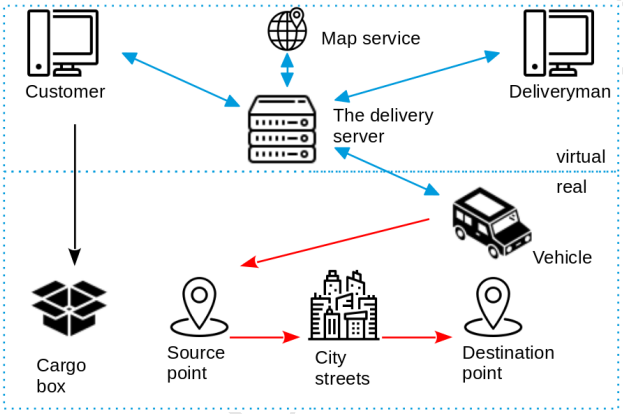

Figure 4: Box delivery process

Virtual part of the box delivery process is realized as some software. It might be a messenger or a blockchain, but anyway it consists from:

- Lager of delivery customer identities (presented as private keys)
- Lager of deliveryman identities (presented as private keys)
- Lager of cargo box identities (presented as pairs of identifiers: permanent and temporary)
- Geo-location service
- Text and image message chart with recording

The virtual part provides communication between customer and deliveryman, authentication for all of participants, as for subjects (customer and deliveryman) also for objects (cargo box and autonomous vehicle). Also it provides global path planing for all autonomous vehicles. It is important to point, that delivery process subjects might be as humans also bots, and in last case delivery process is autonomous on all stages.

Physical part of the delivery system consists from cargo boxes and autonomous vehicles. The cargo boxes contain payload. Autonomous vehicles perform downloading, transportation and uploading of the cargo boxes. Majority autonomous cargo vehicles are joined to a single autonomous delivery system, that maintain own delivery process for each cargo box, but with single plating of all operations.

4 Delivery process

The delivery process is a sequence of physical actions, that result is removing a cargo box from a source to a destination point. All elements of the delivery system are involved in the delivery process.

Physical part of the delivery process additionally includes some external elements, that are not parts of the delivery system. They are source and destination points for loading of the cargo box and roads, that are used by the autonomous vehicle for the cargo box transportation.

The delivery process might be presented as sequence of stages. On each stage participants of virtual part of the delivery process starts some physical action.

Main stages of the delivery process are described below.

- A delivery customer initiates a cargo box delivery process. To initiate the process, the delivery customer identity must be associated to at least one cargo box identity. The customer points a delivery time, a s delivery processource and a destination addresses and a cargo box identifier (or several identifiers, if it is necessary to deliver several boxes).
- A deliveryman first checks association of the cargo box with the customer and next accepts the delivery order.
- The customer sends to the deliveryman a photo of the cargo box on the source place and a photo of the destination place, that points a destination position for the cargo box.
- At the pointed time of the delivery a cargo vehicle sends message to access the source place. When the cargo vehicle comes to the source place, it send photo of the cargo box to the deliveryman, the deliveryman compare the photo from the customer and the photo from the vehicle. Each photo should contain QR code with permanent identifier of the cargo box, that was declared by the customer. If the photos corresponds one to another, the vehicle is allowed to begin loading process.
- When the cargo vehicle comes to the destination place, it also sends message to access the place. After coming to the destination place, the cargo vehicle also sends the destination place photo to the deliveryman, that compares the photo with the photo from the customer. If the place is recognized and unloading is possible in the place, the cargo vehicle unloads the cargo box to the place. After success unloading the cargo vehicle sends message to the customer. If unloading in the destination point is impossible, the cargo vehicle returns to the source place and unload the box there and sends message about unsuccess delivery.

5 Cargo box loading system

To provide mass usage of autonomous delivery system, it components should be cheap and wide used now. Autonomous driving technology potentially allows to use usual cars as autonomous cargo vehicles. If the usual car will be equipped by loading system, it will be able to perform functionality of autonomous cargo box delivery unit.

The loading system provides lifting and moving the cargo box inside luggage compartment of a car.

5.1 Technologies of the loading system

Technologies, that is used in the loading system, should be wide used in other fields, so to perform loading of the box it is used three existing technologies:

- Composite crane booms
- Blocks and wires
- Winches

5.1.1 Composite crane booms. Crane booms are used to provide pivot point in lifting process, so in operation time it should be solid construction. However, solid constructions are not convenient in using. So modern cranes have composite booms, that consists from several solid sections, joined by sheaves [6]. Industrial robotic arms, that also lift some objects, have relative constructions [3]. Such crane booms or robotic arms might be easily folded because of its parts are joined by sheaves. This technology allow to make crane boom, that will be folded inside luggage compartment of a car.

5.1.2 Blocks and wires. Blocks and wires used together allows redirect tension forces and move objects, connected by wires. Lifting systems with wires and blocks are wide used in marine and construction fields. Relating technology is named rigging [5]. Cargo box downloading task is low weight rigging task. So, for this task, it possible to use cheap high extended polyamide wires and cheap plastic blocks.

5.1.3 Winches. Winches are used in the same fields as blocks and wires. The winch consists from drum with turned wire and electrical motor, that rotate the drum. The winches are primary sources of tension force for cranes. Usually it is used winches with single drum. However in the cargo box loading system it is necessary to use specific winches with two drums and single motor, that might be joined to each drum via individual clutch.

5.2 Elements of the loading system

Designed by presented rules, loading system may be realized different ways, but all of the realizations will have the same main elements:

- Crane boom
- Box lock
- Cargo box

5.2.1 Crane boom. The crane boom is used to provide pivot point for lifting cargo box from a ground. At the time of loading an end of the crane boom is shifted outside the luggage comportment. At the time of car movement all part of the crane boom are inside the luggage comportment. The crane boom should be realized as several solid pivots, that is joined one to each other and with car by wires.

5.2.2 Box lock. A cargo box joining to a crane boom by a box lock, that is connected to the crane boom by wires. The box lock is mechanical device that is able to join the standard cargo box. The lock geometry corresponds to geometry of the standard cargo box. The lock provides fixation of the cargo box during loading procedures, the lock releases the box on transportation point inside car or on ground at destination point. The box lock is provided by 4 MEMS sensors (magnetometer + accelerometer), that are placed to corners of the box lock quadrat. This sensors are used for high precision positioning of the lock to the cargo box. The box lock might be realized as quadrat solid frame also as quadrat from tensioned wires with blocks.

5.2.3 Cargo box. A cargo box is a box with standard size, that is using for placing payload inside. The cargo box is provided by special handles, for locking by crane boom. Also the cargo box is provided by two pairs of permanent magnets for precision positioning of the box lock. The pairs of magnets are situated on the opposite crones of the box. One pair of the magnets are positive, another pair is

negative. At the sides of the cargo box QR codes are printed. The QR codes contains information about ID of the cargo box.

5.3 Realization of the loading system

In this realization, it is used tall cargo box, that is lifted by handles in bottom part (the cargo box is marked by purple color on figure 5). Vertical stability in the lifting process is provided by additional wires in top part of the box. The lifted cargo box is moved inside luggage compartment by special winches, that do not participate in lifting process.

Cargo box lock consists from wires ans blocks only. High precision winches, that provides joining to cargo box, are set inside luggage compartment. To the lock is added by special wires for rotation of the frame.

Also in this realization, the crane boom contains many blocks and wires, but no one motor, so it is easy folded and unfolded.

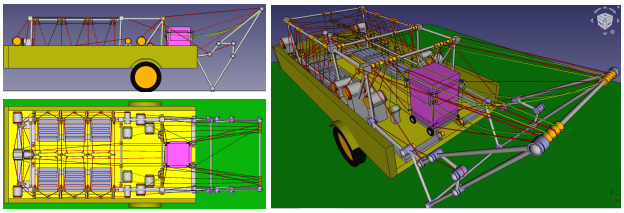

Figure 5: Realization of the loading system

6 Positioning algorithm for the cargo box loading system

In the presented loading system it exists only one process, that requires high precision positioning by means sensors data. The process is positioning of bars of the cargo box lock. In this procedure the cargo box is placed on unknown place over the crane boom and is placed with unknown angle. In all other processes position of the cargo box is known.

The positioning process is performed using magnetic sensors data (figure 6).

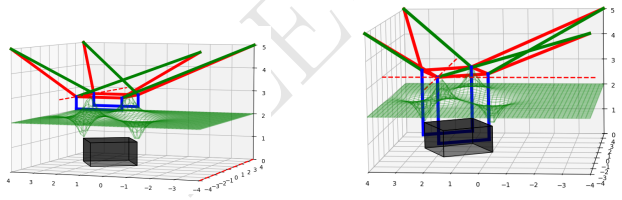

Figure 6: Field of permanent magnets on top of the cargo box

It is used 4 magnetometers on the ends of two bars, that will be joined to the cargo box. When the bars is not lifted down the magnetometers are used for cargo box corners search. When the bars are lifted down the magnetometers are used for detection of the cargo box bottom.

The positioning process starts as transverse moving (figure 7, left side).

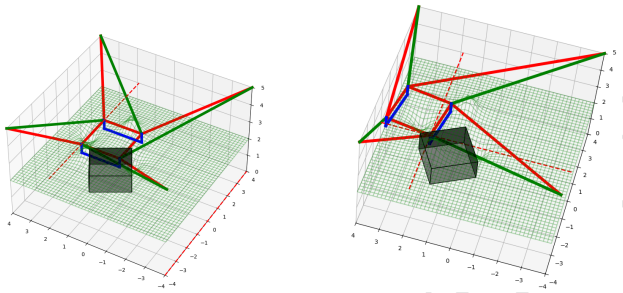

Figure 7: Searching for cargo box first corner - movement of the frame

When this process output some increasing of magnetic field, transverse moving is replaced by longitudinal moving in coordinates of the magnetic field local maximum. In the result of longitudinal moving it is found maximum of magnetic field, that corresponds to one of the cargo box corners. So one of corners of the frame corresponds to one of the cargo box corners.

Next frame begins horizontal rotation relative to the found cargo box corner (figure 7, right side). When all of corners of the frame and of the box are corresponds, the frame stops rotation.

Next bars begin lifting down, until it will be detected magnetic fiend of magnets, that are installed to the bottom of the cargo box. On this place are handles for bars joining (figure 6, left side).

7 Outcomes

It is presented conception of cargo box delivery system. In the conception it is overworked cargo box loading system, as key process of the transportation technology.

Presented variant of realization is able to download and upload cargo boxes with height, that is close to luggage compartment height. This is close to optimum of luggage compartment space using. Final variant of realization is not optimal by number of winches. Most part of winches might be replaced by wire breaks and clutches with electrical motors.

References

[1] 2025. *Largest courier companies by market cap.* Retrieved Sep 1, 2025 from https://companiesmarketcap.com/delivery-services/largest-delivery-companies-by-market-cap/

[2] Ran Wang Chengyuan Huang, Miaojia Lu and Rong Zhang. 2024. Understanding customer preferences for autonomous delivery vehicles in instant delivery: Exploring the impact of delivery and personal attributes. *International Journal of Transportation Science and Technology* (2024). doi:10.1016/j.ijtst.2024.12.001

[3] Darren M.Dawson Frank L.Lewis and Chaouki T.Abdallah. 2004. *Robot manipulator control theory and practice.* Marcel Dekker.

[4] Bo Deng Jingyuan Zhao, Wenyi Zhao. 2023. Autonomous driving system: A comprehensive survey. *Expert Systems With Applications* 242 (December 2023). doi:10.1016/j.eswa.2023.122836

[5] J.A. Klinke. 2016. *Rigging Handbook: The Complete Illustrated Field Reference.* ACRA Enterprises.

[6] L.K. Shapiro and J.P. Shapiro. 2010. *Cranes and Derricks, Fourth Edition.* McGraw Hill LLC.

[7] Surya Ramachandiran Sharan Srinivas and Suchithra Rajendran. 2022. Autonomous robot-driven deliveries: A review of recent developments and future directions. *Transportation Research Part E* 165 (August 2022). doi:10.1016/j.tre.2022.102834

ITS@OpenRAN – Towards an Intelligent Transport System support on the Open Radio Access Network

Edson T. de Camargo
Federal Technology University of Paraná (UTFPR)
Toledo, PR, Brazil
Software/Hardware Integration Lab (LISHA) – Federal University of Santa Catarina (UFSC)
Florianópolis, SC, Brazil
edson@utfpr.edu.br

Antônio Augusto Fröhlich
Software/Hardware Integration Lab (LISHA) – Federal University of Santa Catarina (UFSC)
Florianópolis, SC, Brazil
guto@lisha.ufsc.br

Abstract

Vehicle-to-everything (V2X) communication is a key use case driving 5G and 6G networks. However, the rigidity of traditional radio access networks creates significant barriers to new proposals and innovation. The vision of open radio access networks (Open RAN), proposed by the O-RAN Alliance, offers opportunities to leverage intelligent, real-time control of V2X communication. Yet, integration between V2X and O-RAN remains a work in progress. This paper presents initial steps toward ITS@OpenRAN, an architecture designed to integrate 5G-V2X communication into Open RAN. The proposal includes xApps that use information about vehicles connected to the 5G base station (gNB) and apply the required quality of service for V2X communication. In this approach, RSU functionalities are incorporated as a service within the gNB, specifically in the Distributed Unit (DU).

CCS Concepts

• **Networks** → **Network services**; **Network architectures**.

Keywords

Intelligent Transportation System, Open RAN, O-RAN, Vehicle-to-everything, V2X, ITS

1 Introduction

In a society where information and communication technologies are increasingly prevalent, transportation systems are undergoing significant transformation. Intelligent Transportation Systems (ITS) use sensing, analysis, control, and communication technologies in ground transportation to improve safety, mobility, and efficiency [3]. Data from sensors such as video cameras, inductive circuits, and radars are collected to monitor traffic conditions. This data is processed to identify patterns, predict congestion, and detect incidents. Based on this analysis, control can be exercised, making it possible to change traffic lights or set variable speed limits, for example. Communication is also a key component in the implementation of intelligent transportation systems.

Vehicle-to-Everything (V2X) communications allow a vehicle to interact with other vehicles and elements or actors in its vicinity, such as road infrastructure – including the road itself and its signage – pedestrians, cyclists, and more [6]. Typical V2X applications involve message exchanges defined by the Intelligent Transport Systems (ITS) standards from ETSI: Cooperative Awareness Messages (CAMs), Collective Perception Messages (CPMs), and Decentralized Environmental Notification Messages (DENMs) [12]. A CAM [4] is a standardized message that enables vehicles to share their current status, such as position, speed, and direction, with each other and with entities near the road. These messages are sent periodically to maintain a consistent and shared understanding of the traffic environment beyond the range of a single vehicle's sensors, increasing safety and supporting applications such as collision prevention. CPM messages, as defined by standards such as ETSI EN 103 562, share information about the environment and allow other vehicles or infrastructure connected to the V2X network to extend their own environmental models and improve situational awareness beyond what their own sensors can detect. DENM [5] is a standardized service that enables connected vehicles to send and receive messages about hazardous road conditions or events, such as accidents, roadworks, or obstacles.

A key component in V2X communication is the Roadside Unit (RSU), a fixed device installed along roads that acts as a central communication hub between vehicles and road infrastructure. RSUs are equipped with wireless communication technologies, such as Dedicated Short-Range Communications (DSRC) and cellular V2X (C-V2X), to connect with nearby vehicles that have compatible communication devices. RSUs often process and analyze data collected from various sources, including sensors embedded in the roadway, traffic cameras, and other infrastructure components. This information can be used to manage traffic flow, enhance road safety, and provide real-time updates to drivers.

Dedicated Short-Range Communications (DSRC) and cellular V2X (C-V2X) are the two main V2X communication strategies [8]. DSRC is supported by the IEEE 802.11p standard, while C-V2X is promoted by the Long Term Evolution (LTE) and New Radio (5G NR) standards. In the context of 5G NR technology, 5G-V2X builds on previous 3GPP efforts to enable vehicular communications and receives particular attention from the intelligent transportation systems community due to its potential to provide high speed and low latency for critical safety services such as collision avoidance, while also enabling advanced infotainment and coordination for smoother traffic flow. It leverages technologies like Sidelink (direct communication) for a fully connected mobility ecosystem. However, 5G-V2X presents a complex set of challenges, mainly due to its constantly changing temporal and spatial dynamics.

Traditional Radio Access Networks (RANs) face challenges in achieving the flexibility needed to optimize advanced control mechanisms and improve performance metrics to meet the diverse and

demanding requirements of the 5G-V2X environment [9]. Additionally, RANs are predominantly composed of monolithic units supplied by a limited number of vendors and are viewed by operators as black boxes. This results in high implementation and maintenance costs, limits innovation, and restricts the entry of new players into the market [10].

To address the limitations of traditional RANs, the Open Radio Access Network (O-RAN) and its implementation by the O-RAN Alliance aim to promote virtualized RANs, where disaggregated components are connected through open interfaces and optimized by intelligent controllers. This creates a new paradigm for RAN design [10]. The O-RAN vision enables components from different vendors to interoperate, fostering innovation, flexibility, and cost reduction. In the 5G-V2X context, O-RAN capabilities can support advanced use cases, as its core paradigms provide an ideal framework for orchestrating vehicular communication [2]. However, as noted by Linsalata et al. [8, 7], research on the potential synergies between O-RAN and V2X remains unexplored. In other words, how to integrate V2X and O-RAN and fully leverage O-RAN in the ITS and V2X world – whether for CAM or CPM exchange, in DENM service, or in the RSU itself – remains a path to be paved.

This paper presents initial steps toward ITS@OpenRAN, an architecture to integrate 5G-V2X communication into O-RAN through the development of intelligent software applications (called V2X xApps) that leverage the wealth of information related to vehicles connected to the 5G base station (gNB). These software applications are responsible for performing network functionalities, such as slicing and applying necessary quality of service requirements in V2X communication. In this proposal, the RSU functionalities will be incorporated as a service in gNB.

The remainder of this article is organized as follows. Section 2 provides a brief overview of the O-RAN architecture and related work. Section 3 describes our proposed architecture, and Section 4 presents the conclusion.

2 O-RAN Architecture Overview

This section briefly introduces the O-RAN architecture and its main components, and presents recent efforts to integrate O-RAN with V2X.

The O-RAN Alliance, established in 2018, aims to implement disaggregated, virtualized, software-based components connected through open, well-defined interfaces that are interoperable across different vendors over 3GPP LTE and NR RANs. As shown in Figure 1, the O-RAN Alliance separates base station functions into a Central Unit (CU), Distributed Unit (DU), and Radio Unit (RU). O-RAN also connects these units to intelligent controllers via open interfaces that transmit RAN telemetry and enable the deployment of control actions and policies. Two Intelligent RAN Controllers (RICs) manage and control the network on near-real-time (10 milliseconds to 1 second) and non-real-time (more than 1 second) timescales [10]. The logical division of O-RAN allows different functionalities to be deployed in various locations on the network and on different hardware platforms. For example, CUs can be virtualized on white box servers at the network edge, while RUs are generally implemented on specialized hardware and deployed close to RF antennas.

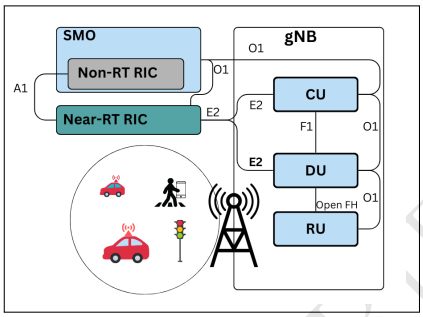

Figure 1: O-RAN Architecture adapted from [10].

Figure 1 also shows the Near-real time RIC (Near-RT-RIC) and the Non-real time RIC (Non-RT-RIC). The near-RT RIC is deployed at the network edge and operates control loops with a periodicity between 10 ms and 1 s, interacting with DUs and CUs in the RAN. It includes multiple applications that support custom logic, called xApps, as well as the services required to execute them. An xApp is a microservice used to manage radio resources through specific service interfaces and models. It receives data from the RAN and, if necessary, computes and sends control actions back. Non-RT RIC is a component of the Service Management and Orchestration (SMO) framework and complements Near-RT RIC for intelligent RAN operation and optimization on a timescale greater than 1 second. Non-RT RIC provides guidance, enrichment information, and machine learning model management to Near-RT RIC via rApps, the software applications hosted in Non-RT RIC. Furthermore, Non-RT RIC can influence SMO operations, allowing it to indirectly govern all O-RAN architecture components connected to SMO by making decisions and applying policies that affect thousands of devices.

O-RAN also provides open interfaces that connect different components of its architecture, as shown in Figure 1. Among these interfaces, the E2 interface connects the NearRT-RIC to the RAN nodes. The A1 interface links the RIC controllers, and the O1 interface connects all other RAN components for management and orchestration of network functions. The F1 interface connects the CU to the DU.

2.1 Related Work

Recent work has explored the potential of O-RAN in V2X [8, 7, 9, 11], as described next.

Linsalata et al. [8] discuss integration strategies and highlight the challenges and opportunities of leveraging O-RAN to enable real-time V2X control. One strategy proposes that communication with the RIC from the base station, where the O-RAN elements are located, should also be possible from the RSUs. Specifically, they argue that an E2 termination can be included in RSUs to enable their dynamic control and access to the extensive information related to autonomous and connected vehicles that they provide. E2 messages can therefore be multiplexed with other communications in the RSU control plane. The authors also identify research avenues in resource allocation, beam selection and management, and mitigating signal jamming through retransmission and multi-hop

mechanisms, for example, by utilizing nearby connected vehicles and the RSU.

The same authors in [7] propose a novel architecture that establishes a low-frequency O-RAN-based control plane to ensure reliable and efficient multi-hop connectivity between connected autonomous vehicles over millimeter waves. They also examine and test the technological feasibility of this integrated architecture and expand the existing network simulator 3 (ns-3) modules, resulting in a simulation framework for experimenting with the O-RAN-enabled V2X system.

Mobi-O-RAN [9] is an architecture that integrates V2X and O-RAN by introducing a joint resource admission and provisioning control structure adapted to O-RAN environments. The authors propose a reinforcement learning model with constraints to dynamically balance user admissions and allocate resources to admitted users. Simulation results show substantial performance gains compared to traditional methods.

Rehman et al. [11] argue that, unlike traditional RANs, the disaggregated architecture and RIC of O-RAN allow for real-time adaptation to the challenges of non-line-of-sight communication (NLOS), a problem still unresolved in many existing V2X systems. Therefore, they propose an adaptive relay framework that leverages near real-time RIC to dynamically enhance V2X reliability through dynamic relay selection and interference-aware resource coordination.

3 Proposed Architecture

According to 3GPP Technical Specification 23.287 [1], an RSU is not an architectural entity but an implementation option. The specification illustrates this by collocating V2X application logic or a server with certain 3GPP system entities, as shown in Figures 2 and 3. Figure 2 shows a UE-type RSU, which combines a UE with V2X application logic. PC5 refers to the direct communication interface (sidelink). Figure 3 shows a gNB-type RSU, where the RSU consists of a gNB and a V2X Application Server. The Uu interface is the air interface connecting the UE to the gNB.

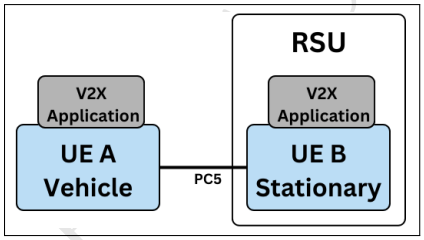

Figure 2: RSU includes a UE and the V2X application [1].

We propose implementing RSU functionalities in both the UE and the gNB to make RSU information accessible to the O-RAN intelligent controllers. By standard, a base station (the gNB for 5G-V2X) is equipped with E2 terminations to enable data collection and control. However, to allow communication between the RAN controllers and a stationary UE, an extension to the E2 interface would be necessary, as noted by Linsalata et al. [8]. Another issue is that the E2 interface does not support mobility.

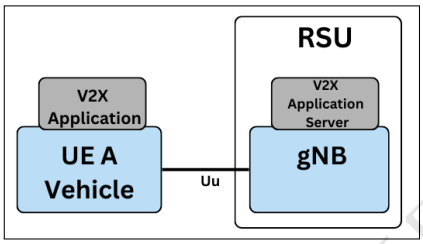

Figure 3: RSU includes a gNB and an V2X Server [1].

Another important issue is O-RAN's support for ITS. To achieve this, it is necessary to implement support for CAM and CPM messages as defined in the ETSI TS 102 and 103 standards, along with the ETSI TS 122 and 302 family of standards, which define the context of V2X over 5G, and DENM to allow connected vehicles to send and receive messages about hazardous road conditions or events, such as accidents, roadworks, or obstacles.

Based on the extensive information provided by the RSU, a set of specialized xApps and rApps, called V2X xApps and V2X rApps, can interpret and act on data received directly from ITS elements, such as network slicing. Network slicing efficiently divides the physical network into slices and naturally facilitates various V2X use cases [9]. The xApp manages slices for V2X communication according to the different data traffic present in the urban environment and its own performance, security, and latency requirements. Different QoS requirements are applied to each slice to support and prioritize CAM and CPM message traffic, meeting the necessary latency, transfer rate, and packet loss metrics. In this context, an xApp can receive data on vehicle position and mobility, as well as channel status and interference profile. The xApp is further enriched with information from the base station to allow dynamic control and leverage the wealth of information related to connected vehicles.

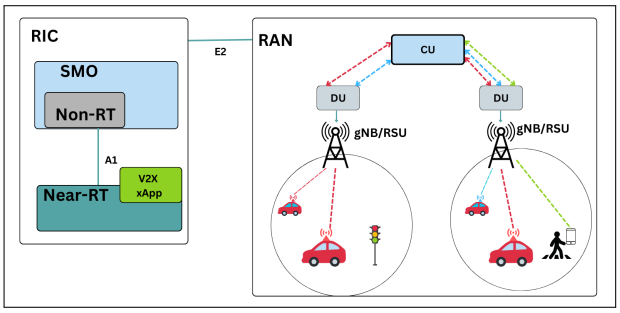

Figure 4: Proposed ITS@Open-RAN Architecture

Figure 4 shows a high-level overview of the proposed architecture, which builds on the previously described Open-RAN architecture. In the lower layer, base stations (gNBs) provide connectivity to endpoints – vehicles, pedestrians, and other elements of intelligent transportation systems – that connect to the 5G infrastructure. The RSU, a service to be implemented in this proposal, is integrated with the gNB as a service in the DU. The DU connects to a Central Unit (CU). RSUs, DUs, and CUs are connected to the Near-RT

RAN controller via the E2 interface. The controller includes a V2X xAPP responsible for implementing network slicing and quality of service functionalities for the UEs connected to the respective DU. The colored arrows in the figure represent different types of traffic, each corresponding to a network slice. The V2X xApp for the O-RAN and 5G-V2X context inherits the benefits of the O-RAN architecture and enables integration with solutions from various automotive vendors, thanks to open interfaces and a preference for open software and hardware. This approach also encourages innovation and accelerates the time to market for new network services for intelligent transportation systems.

The proposed architecture, which integrates ITS and O-RAN, aims to address use cases such as the following. At a busy urban intersection, traffic control units, connected traffic lights, and vehicles equipped with V2X communication are present. A small car (vehicle A) approaches the intersection but is hidden behind a large delivery truck (vehicle B) in front of it. A pedestrian is crossing the street at a crosswalk but is not visible to vehicle A because the truck blocks its view. An RSU with a 5G antenna monitors the area using sensors (e.g., cameras, LiDAR) and is connected to the V2X network. The goal is to prevent the hidden vehicle A from crossing the crosswalk when its view is obstructed by the truck. To achieve this, CAM and CPM messages will be used and must be transmitted within strict time limits to avoid the scenario shown in Figure 5[1] through the implemented V2X xApps, where the car crosses the crosswalk and/or stops too late.

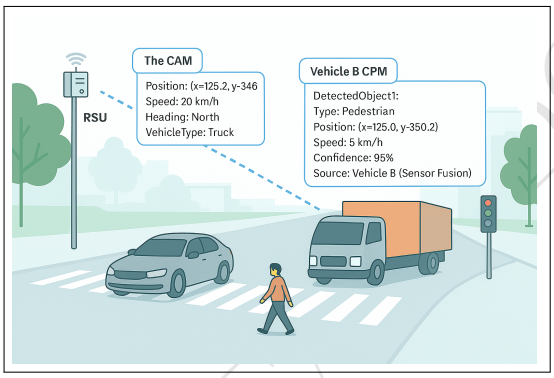

Figure 5: Scenario ITS and O-RAN.

4 Conclusion

The integration of O-RAN and 5G-V2X provides a flexible, scalable, and cost-effective solution, but several challenges remain. Bringing V2X to Open-RAN, along with introducing and improving new concepts and mechanisms, will significantly expand the application scope of V2X solutions and accelerate progress toward Intelligent Transportation Systems. Conversely, solutions proposed for Open-RAN gain traction when considering the technical specifications that govern V2X communication, such as those defined for CAM and CPM messages, as well as issues specific to the automotive domain, including mobility and security of connected vehicles. In our proposal, the RSU is implemented in one of the O-RAN elements, providing specialized xApps with valuable environmental information. Next steps include simulating, implementing, and validating the proposed architecture using simulators.

[1]Image created with the help of AI tools.

5 Acknowledgments

This work was partially funded by Fundação de Apoio da UFMG (Fundep), through Linha VI – Conectividade Veicular, a prioritary program from Mover (Mobilidade Verde e Inovação), project Auto5G (29271.02.01/2022.01-00).

References

[1] 3GPP. 2025. Architecture enhancements for 5g system (5gs) to support vehicle-to-everything (v2x) services. 3GPP. ETSI TS 123 285 V19.0.0. (Sept. 2025). https://www.etsi.org/deliver/etsi_ts/123200_123299/123285/19.00.00_60/ts_123285v190000p.pdf.

[2] Bharat Agarwal, Ralf Irmer, David Lister, and Gabriel-Miro Muntean. 2025. Open ran for 6g networks: architecture, use cases and open issues. *IEEE Communications Surveys & Tutorials*, 1–1. doi:10.1109/COMST.2025.3562429.

[3] George Dimitrakopoulos and Panagiotis Demestichas. 2010. Intelligent transportation systems. *IEEE Vehicular Technology Magazine*, 5, 1, 77–84.

[4] ETSI. 2014. Part 2: specification of cooperative awareness basic service. ETSI. ETSI EN 302 637-2. Technical report. V1.3.2. (Jan. 2014). https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.02_60/en_30263702v010302p.pdf.

[5] ETSI. 2019. Specifications of decentralized environmental notification basic service. ETSI. ETSI EN 302 637-3. V1.3.1. (Apr. 2019). https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.02_60/en_30263702v010302p.pdf.

[6] Mario H. Castañeda Garcia, Alejandro Molina-Galan, Mate Boban, Javier Gozalvez, Baldomero Coll-Perales, Taylan Şahin, and Apostolos Kousaridas. 2021. A tutorial on 5g nr v2x communications. *IEEE Communications Surveys & Tutorials*, 23, 3, 1972–2026. doi:10.1109/COMST.2021.3057017.

[7] Francesco Linsalata, Eugenio Moro, Franci Gjeci, Maurizio Magarini, Umberto Spagnolini, and Antonio Capone. 2024. Addressing control challenges in vehicular networks through o-ran: a novel architecture and simulation framework. *IEEE Transactions on Vehicular Technology*, 73, 7, 9344–9355. doi:10.1109/TVT.2024.3355202.

[8] Francesco Linsalata, Eugenio Moro, Maurizio Magarini, Umberto Spagnolini, and Antonio Capone. 2024. Open ran-empowered v2x architecture: challenges, opportunities, and research directions. In *2024 IEEE Vehicular Networking Conference (VNC)*, 113–116. doi:10.1109/VNC61989.2024.10576004.

[9] Eugenio Moro, Francesco Linsalata, Maurizio Magarini, Umberto Spagnolini, and Antonio Capone. 2025. Advancing o-ran to facilitate intelligence in v2x. *IEEE Network*, 1–1. doi:10.1109/MNET.2025.3553581.

[10] Michele Polese, Leonardo Bonati, Salvatore D'oro, Stefano Basagni, and Tommaso Melodia. 2023. Understanding o-ran: architecture, interfaces, algorithms, security, and research challenges. *IEEE Communications Surveys & Tutorials*, 25, 2, 1376–1411.

[11] Abdul Rehman, Husnain Shahid, Barbara M. Masini, Piergiuseppe Di Marco, and Fakiha Munawar. 2025. O-ran-enabled adaptive relaying for robust v2x communication in urban nlos scenarios. *IEEE Access*, 13, 200946–200956. doi:10.1109/ACCESS.2025.3633945.

[12] Jin Yan. 2024. *Towards Dependable 5G-NR Sidelink Communication*. Ph.D. Dissertation. Sorbonne Université.

Dynamic Map-based Data-Centric Approach for Tourism and Cultural Heritage Preservation Digital Twins

João Spínola Falcão
joaofalcao2004@gmail.com
Universidade Salvador (UNIFACS)
Salvador, Brazil

João G. Perrone Hohlenwerger
joaoperrone1831@gmail.com
Universidade Salvador (UNIFACS)
Salvador, Brazil

Daniel C. Santos
dann.costasantos@gmail.com
Universidade Salvador (UNIFACS)
Salvador, Brazil

Lucas Almeida de Sousa
lucas.ads@fieb.org.br
Universidade Salvador (UNIFACS)
Salvador, Brazil

Nazim Agoulmine
nagoulmine@gmail.com
Université Paris-Saclay - Évry
Évry, France

Joberto S. B. Martins*
joberto.martins@gmail.com
Universidade Salvador (UNIFACS)
Salvador, Brazil

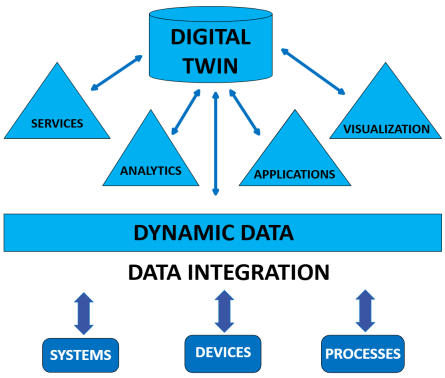

Figure 1: TEASER - Dynamic and Customized Map-based Data-centric approach for a Specialized Tourism and Cultural Heritage Digital Twins

ABSTRACT

Tourism is an essential and growing economic activity worldwide, bringing benefits such as job creation, revenue generation, and tax revenue, and driving economic prosperity. Tourism activity may also have negative impacts on cities, such as over-tourism and pressure on housing and real estate, which are increasingly recognized as problems communities must address. Cultural heritage is an essential asset of cities and countries that must be preserved. Cultural heritage, as an asset, is commonly explored through tourism activities and may have negative impacts, including physical degradation, commodification, and loss of authenticity, among others. Tourism and cultural heritage management are common elements of smart city digital transformation strategies, in which the well-being of citizens and the maintenance of public assets are among the goals. A digital twin is a data-driven virtual representation of a physical object, system, or environment. It typically integrates real-time data and computational models to simulate, monitor, analyze, and predict the behavior and performance of the represented entity. However, although digital twin technology has been widely adopted in manufacturing, Industry 4.0, and urban planning for smart cities, there remains a gap in specialized digital twins for tourism and cultural heritage management. This paper proposes a QGIS-based, data-centric approach to digital twin frameworks that supports the management, development, and deployment of tourism and cultural heritage services and applications in smart cities. The data-centric approach is embedded in a specialized digital twin focusing on the Salvador Historic Center - Pelourinho, a highly important cultural asset and tourism spot for the city of Salvador. Currently, Pelourinho faces a persistent challenge in sustaining tourism flux while safeguarding its cultural and heritage assets. Preliminary results indicate that the data-centric approach adopted by Pelourinho's DT facilitates data visualization, integrates data silos, and adequately supports management, enabling managers to address heritage preservation and conservation issues, control over-tourism, and implement urban resilience and climate adaptation measures.

CCS CONCEPTS

• **Computer systems organization** → **Embedded and cyber-physical systems**; • **Computing methodologies** → *Modeling and simulation.*

KEYWORDS

Tourism, Cultural Heritage, Digital Twin, Digital Maps, QGIS, Data-Centric, Smart City, Digital Transformation.

*All authors contributed equally to this research.

1 INTRODUCTION

Tourism is an important economic activity with a huge impact on economies around the world (Council, 2025). As discussed in (Santos et al., 2025), 1.4 billion international tourist arrivals were recorded in 2024, generating 10.9 trillion dollars in economic activity, an amount equivalent to 10% of the Global GDP (Gross Domestic Product). In addition, tourism activity created jobs for 357 million people, equivalent to 10.6% of the global workforce (Council, 2025).

Tourism global indicators suggest that tourism activity is increasing and is commonly associated with impacts on cultural heritage and the well-being of inhabitants (Santos-Júnior et al., 2020).

Cultural heritage preservation is a fundamental aspect of cities, including new smart cities, and digital transformation management strategies (Martins et al., 2024). In this context, Digital Twins (DTs) can be used for tourism management and to support the preservation of cultural heritage and historic sites in general (e.g., Serbouti et al., 2025; Dang et al., 2023; Akyol and Avci, 2025). In summary, digital twins customized for cultural heritage create virtual replicas of historic sites that support in situ management and preservation initiatives.

The Historic Center of Salvador, Brazil, also known as Pelourinho (Figure 2), is recognized as a UNESCO World Heritage Site. It faces significant challenges related to planning, climate change adaptation, optimization, and the facilitation of tourist flux, as well as urban heat islands. These issues are exacerbated by the lack of management of tourism flux, the scarcity of green spaces, the high density of buildings, and inadequate land use, which compromise not only the environment but also the region's historical and cultural heritage.

Artificial intelligence (AI) is another important component explored in current approaches for tourism management and cultural heritage **AI for tourism management**. AI support requires a data-rich setup, in which the quantity and quality of the data are paramount for providing effective AI-based solutions (ERCIM, 2025; Sánchez-Martín et al., 2025; Almeida et al., 2025).

The research gap in the scenario with DTs being used to manage tourism and cultural heritage is therefore threefold:

- There is a gap of solutions based on digital twins (DTs) that consider tourism, cultural heritage, and social issues concomitantly;
- Digital twins for cultural heritage are mostly focused on simple digitization and visualization tools, and lack incorporating data-rich models; and
- There is a gap for digital twins (DTs) that embed artificial intelligence tools to address smart city multi-faceted issues (and related datasets), such as tourism, social well-being, and cultural heritage.

As such, the main research question addressed in this paper is as follows:

- Is it possible to integrate a data-rich strategy into a specialized digital twin framework that holistically deals with tourism and cultural heritage urban issues?

The research gap in this scenario is how to incorporate several data-rich datasets from different domains and support new strategies using DT technology for urban development and smart cities.

This paper's objective is to propose a data-rich strategy integrated into a specialized QGIS-based digital twin to support integrated tourism and cultural heritage management actions in the Pelourinho Historic Center.

The paper addresses the following contributions:

- Develop a data-centric approach for a new type of specialized digital twin;
- Integrate IoT sensors and predictive analytics with artificial intelligence to address the impact of tourism and climate change on the preservation of the historical heritage of Pelourinho; and
- Develop a specialized DT enabling real-time site monitoring, scenario simulations, and predictive maintenance, offering a replicable, data-centric approach to tourism and heritage preservation.

The paper innovates by adopting QGIS maps georeferencing and providing the necessary flexibility to incorporate, on demand, dynamic data with various structural elements such as Internet of Things (IoT) sensors, heritage, historic, and social data related to the real-world scenario of historical centers, while capturing and exporting dynamic data for these elements. Practical examples of incorporated structural components include cameras, on-the-fly people and tourism flux, vehicle flux, temperature and humidity sensors, cultural and historic data elements, among others.

This paper is organized as follows: the introduction section 1 presents tourism and cultural heritage preservation using DT technology. Section 2 introduces the digital twins in the context of tourism and cultural heritage. Section 3 presents the data-centric

Figure 2: Salvador Historic Center (Pelourinho) (Source: Bacelar, 2025).

approach used by the digital twin, followed by a use case on tourism flux and a discussion on its embedding within the digital twin architecture. Finally, Section 4 concludes the discussion with final considerations on the DT data-rich approach and future work.

2 DIGITAL TWINS FOR TOURISM AND CULTURAL HERITAGE PRESERVATION IN SMART CITIES

The smart city strategy is an "umbrella" trend that considers various pillars for urban development, such as mobility, infrastructure, security, health, and energy, to mention some (Shao and Min, 2025; Farid et al., 2021). Smart city strategies aim to promote the well-being of city inhabitants while simultaneously optimizing urban management and services.

Technology and data form the core of smart city project development. The Internet of Things (IoT), artificial intelligence, and digital twins, to name a few, are fundamental technologies of smart city development across nearly all areas, including tourism and cultural heritage preservation (Menaguale, 2023).

2.1 Digital Twins

A digital twin is a digital representation of a physical asset, process, or system that accurately replicates its data, behavior, and interaction with other assets. DTs enable real-time monitoring, situation simulation, and data analysis, providing valuable insights into the performance and behavior of the modeled counterparts (Mazzetto, 2024; Afif Supianto et al., 2024).

Digital Twins emerge as a promising tool for urban development. They allow virtual modeling of the urban environment and enable real-time data analysis to support tailored strategies and policies for smart, sustainable, and resilient cities (Afif Supianto et al., 2024).

Digital twins for tourism and cultural heritage preservation are a current trend (Almeida et al., 2025). A specialized digital twin for tourism and cultural heritage that handles dynamic data and integrates data across different applications is shown in Figure 1. One fundamental aspect of this proposal is the data-rich approach, which is pivotal for tourism and cultural heritage and relies heavily on artificial intelligence.

2.2 Digital Twin for Tourism and Heritage Preservation - The Pelourinho Historic Center Issues

The Historic Center of Salvador is one of the most significant cultural and architectural heritage sites in Brazil and was designated a UNESCO World Heritage Site. Despite its historical, symbolic, and economic significance, Pelourinho faces a set of persistent, interrelated urban challenges that directly affect both tourism sustainability and heritage preservation. These challenges are intensified by the area's complex spatial configuration, its high tourist appeal, and the limitations of traditional urban management approaches.

One of the central issues concerns the lack of systematic management of tourist flows. Tourism activities in Pelourinho are highly concentrated in specific streets, squares, and time periods, leading to spatial and temporal imbalances. The absence of continuous monitoring mechanisms results in overcrowding during peak hours and underutilization of other areas, increasing pressure on historic buildings, public spaces, and urban infrastructure. This phenomenon contributes to overtourism, reducing the quality of the visitor experience and negatively impacting residents' well-being.

Another critical problem concerns human density and congestion in heritage-sensitive spaces. Excessive visitor concentrations accelerate physical degradation processes, such as the wear of pavements, facades, and public equipment, while also increasing risks to safety, accessibility, and emergency response. The lack of real-time data and predictive tools limits public authorities' capacity to anticipate and mitigate these effects.

Pelourinho also faces significant environmental challenges, particularly those related to urban heat and noise pollution. The high density of built structures, limited vegetation coverage, and intense pedestrian activity contribute to the formation of urban heat islands, which affect thermal comfort for both residents and tourists. In parallel, cultural events, street performances, vehicular circulation in surrounding areas, and large tourist crowds generate elevated noise levels, compromising the acoustic comfort of humans and animals and potentially affecting the structural integrity of historic buildings over time.

In addition to these issues, data fragmentation and institutional silos constitute structural limitations to integrated urban management. Information on tourism, cultural heritage, environmental monitoring, and urban infrastructure is typically dispersed across multiple institutions and formats, hindering comprehensive analysis and coordinated decision-making. This fragmentation restricts the ability to establish correlations between tourism dynamics, environmental conditions, and heritage conservation indicators.

Furthermore, Pelourinho is increasingly exposed to climate change-related risks, including rising temperatures and more frequent extreme weather events. These factors exacerbate existing vulnerabilities of historic structures and public spaces, demanding adaptive and resilient management strategies supported by data-driven tools.

Given this context, Pelourinho constitutes a complex urban environment in which tourism pressure, environmental stress, and challenges of heritage preservation coexist and interact. Addressing these issues requires an integrated approach that combines real-time monitoring, historical data, spatial analysis, and predictive modeling. In this sense, a specialized Digital Twin emerges as a suitable technological solution to support the analysis, simulation, and management of tourist flows, environmental impacts, and preservation strategies in the Historic Center. By centralizing heterogeneous data and enabling dynamic representations of urban processes, the Pelourinho Digital Twin aims to provide actionable insights for sustainable tourism management and the long-term preservation of cultural heritage.

3 DIGITAL TWIN WITH DYNAMIC AND CUSTOMIZED DATA-CENTRIC AND QGIS MAP-BASED APPROACH

The QGIS-based data-centric approach of the digital twin of Pelourinho (Pelourinho's DT) (Figure 3) provides several key features for managers, urban developers, and urban planners within the context of a smart city strategy. The supported key features are:

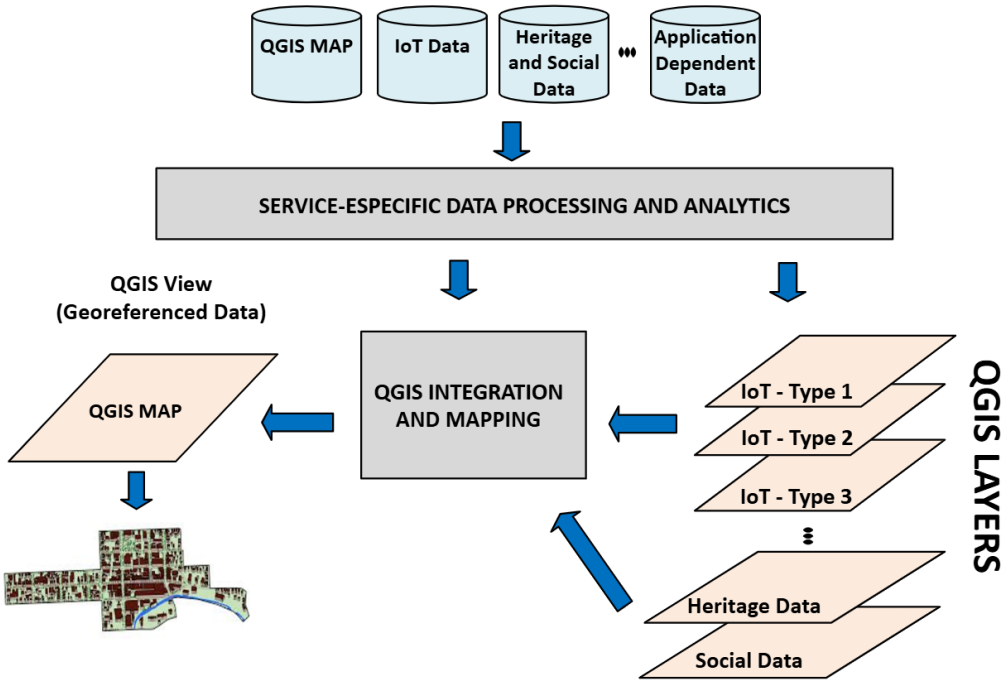

Figure 3: Pelourinho's DT Data-Centric Approach.

- The integration of heterogeneous and diverse data sources from IoT physical and logical sensors, social data from institutions dealing with Pelourinho's settlement, and historical and cultural heritage data elements;
- Flexible data analytics processing with multiple artificial intelligence tools and algorithms; and
- Use of georeferenced data to facilitate spatial-data correlation and management visualization with dynamic and flexible input parameters.

3.1 The Pelourinho's DT Databases and Data-Centric Approach

The main principle of the data-centric approach adopted in this research is to shift from advanced data models toward improving the quality and quantity of the data. One of the main arguments for this approach comes from the fact that Pelourinho's DT uses artificial intelligence, and AI is moving to data-centric AI since the algorithms need data with quantity and quality (Zha et al., 2025; Siegl et al., 2016).

The Pelourinho's DT data-centric requirements are as follows:

- Dynamically integrate time-series data acquired with the help of various types of IoT sensors for distinct DT's situational analysis;
- Support diverse and specific types of data necessary for analyzing the situation of Pelourinho's tourism, cultural heritage, and climate change; and
- To georeference all data relative to the QGIS map to support real-location in Pelourinho's in-site management and impact analysis.

The Pelourinho's database is composed of the following elements (Figure 3):

- A geographic QGIS database;
- An IoT-based database of sensed parameters relevant for Pelourinho's tourism and climate change impact analysis;
- A heritage-related historical database and social data database; and
- An application-defined database for supporting Pelourinho's DT customization for different smart city management strategies and applications.

The geographic and georeferenced QGIS database stores all data in accordance with QGIS map-processing requirements and operations.

The IoT-based database stores all physical and sensor parameters associated with the services and applications supported by the DT, including tourism and climate change applications. All data is time-referenced to allow time-series analysis supported by the DT.

The application-specific database includes data parameters, knowledge, and information concerning the custom application and services supported by the DT.

QGIS does not explicitly support a dynamic, data-centric approach. As such, the Pelourinho's DT employs a database-to-QGIS

mapping approach, as illustrated in Figure 3. In summary, it works as follows:

- The used database stores IoT and other types of data, allowing a generic set of service-specific data processing and data analysis by distinct AI algorithms;
- Data stored in the databases map to the layers' structure used by the QGIS maps; and
- The deployed database structure ensures item Time-series mappings and relations.

3.1.1 Pelourinho's DT Data-Centric Deployment. The deployment of Pelourinho's Digital Twin adopts a data-centric approach, in which data serves as the architecture's structuring element. At the same time, applications and visualization mechanisms act as access and analysis layers.

The database used in Pelourinho's DT is the open-source PostgreSQL with the PostGIS extension, which supports spatial data storage and manipulation (Salunke and Ouda, 2024).

A PostgreSQL database was chosen due to QGIS's support for PostGIS, including spatial capabilities for 2D, 3D, and 4D, spatial relationships, and spatial analytics. In addition, it is high-performance with large datasets, ensuring the DT's scalability for large-scale application scenarios.

As illustrated in Figure 3, the overall data deployment is based on a set of logically distinct yet integrated databases (IoT, cultural heritage, social, and application-dependent data) that feed a Service-Specific Data Processing and Analytics (SSDPA) module and, subsequently, the QGIS-based integration and visualization layer.

The service-specific data processing and analytics (SSDPA) module is part of the DT main architecture. Regarding data manipulation, the SSDPA supports spatial SQL queries and database-defined views.

The main characteristics of the data-centric deployment adopted in the Pelourinho's DT are as follows:

- A central spatial database stores different types of data;
- Service-specific pre-processing is performed directly on the database using spatial SQL;
- QGIS acts as an integration and visualization layer; and
- The database tables and views reflect the QGIS layer structure used for georeferenced operations.

3.2 A Use Case: Pelourinho's Tourism Flux

The use case diagram presented in this study formalizes a practical application of the Pelourinho Digital Twin focused on the analysis and management of tourist flux within the Historic Center of Salvador. The diagram defines the main actors, the system boundaries, and the interactions that enable the Digital Twin to operate as a monitoring, analytical, and decision-support platform.

Three primary actors interact with the Pelourinho Digital Twin System. The Urban Manager (Public Authority) represents municipal and heritage management institutions responsible for regulating urban space, tourism activities, and preservation policies. The Tourism Planner acts as a strategic user focused on organizing events, designing tourist routes, and managing visitation patterns. The Researcher/Analyst represents academic and technical users who employ the Digital Twin to investigate spatial, temporal, and environmental dynamics associated with tourism.

At the core of the diagram lies the Pelourinho Digital Twin System, which encapsulates the data-centric infrastructure, analytical models, and visualization tools described in the architectural framework. The central use case, Monitor Tourist Flow, constitutes the system's primary functionality. It enables stakeholders to visualize and track the spatial and temporal distribution of tourists across streets, squares, and points of interest, using dynamic maps and indicators generated from integrated data sources.

This core functionality explicitly includes two subordinate use cases: Estimate Tourist Density and Identify Congestion Hotspots. The first relies on the fusion of heterogeneous data collected from different sources, such as camera detections, passive WiFi/Bluetooth signals, and statistical estimates, to generate reliable density measures. The second use case builds on these estimates to detect areas of excessive concentration, thereby supporting early identification of overcrowding risks in heritage-sensitive spaces.

The use case Simulate Tourism Flow Scenarios extends the monitoring process by enabling prospective analyses. Through scenario simulation, stakeholders can evaluate "what-if" situations, such as cultural events, changes in pedestrian routes, or seasonal variations in visitation. This function highlights the predictive capacity of the Digital Twin, distinguishing it from conventional monitoring systems.

Another relevant use case, Assess Environmental Impacts, connects tourist flow patterns with environmental variables, particularly urban noise levels and microclimatic conditions. By correlating human density with acoustic pressure and temperature, the Digital Twin supports integrated assessments of comfort, sustainability, and heritage preservation.

Finally, the use case Support Decision-Making aggregates the outputs of monitoring, estimation, simulation, and environmental assessment. It represents the ultimate purpose of the Digital Twin: to provide structured, data-driven insights that inform urban governance, tourism planning, and preservation strategies. This use case depends on the results of multiple analytical processes, reinforcing the systemic and integrative nature of the proposed Digital Twin.

Overall, the use case diagram demonstrates how the Pelourinho Digital Twin uses a data-centric approach to tourism management. By clearly defining actors, system functions, and their relationships, the diagram illustrates how heterogeneous data are transformed into actionable knowledge, supporting both real-time management and strategic planning in a complex historic urban environment. The use case diagram can be identified below

.

3.3 The Data-Centric Approach embedded in the Pelourinho's DT Architecture

The Pelourinho's digital twin architecture is illustrated in Figure 5. It is composed of five layers, applications, and components (de Souza and Martins, 2025). The architecture's layers are hierarchical and interact together to represent, monitor, and optimize the physical environment. The physical layer represents the real world, including physical assets (buildings, urban infrastructure, equipment, vehicles, people, and social-related data, among others).

The proposed architecture for the Pelourinho Digital Twin was developed from a systemic vision that integrates heterogeneous

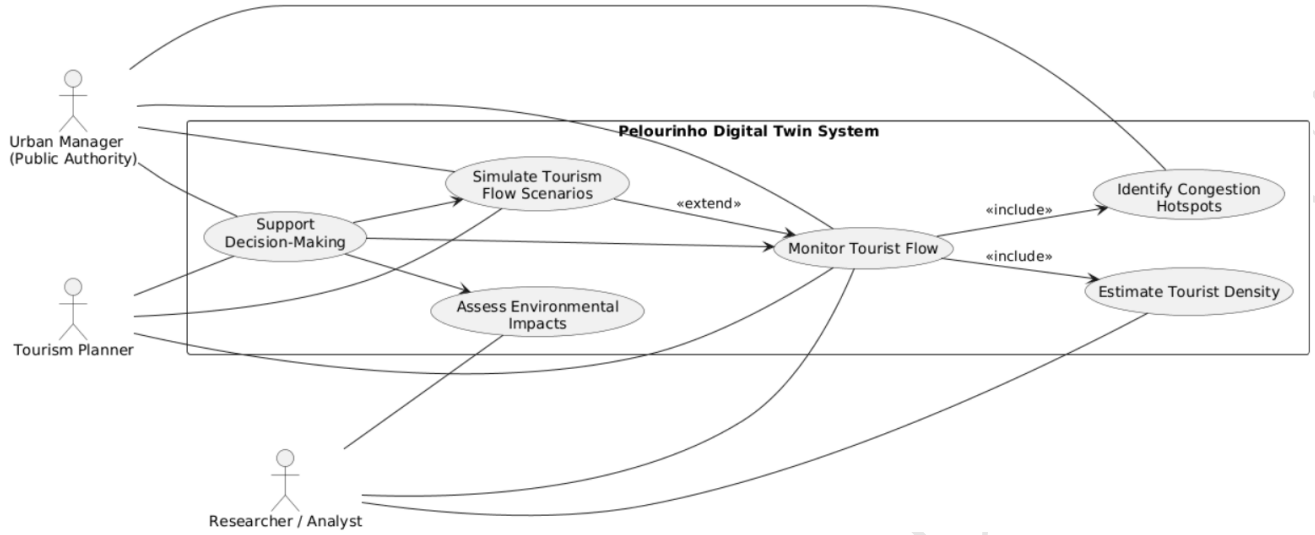

Figure 4: Case Diagram - Pelourinho's DT Tourism Flux Service

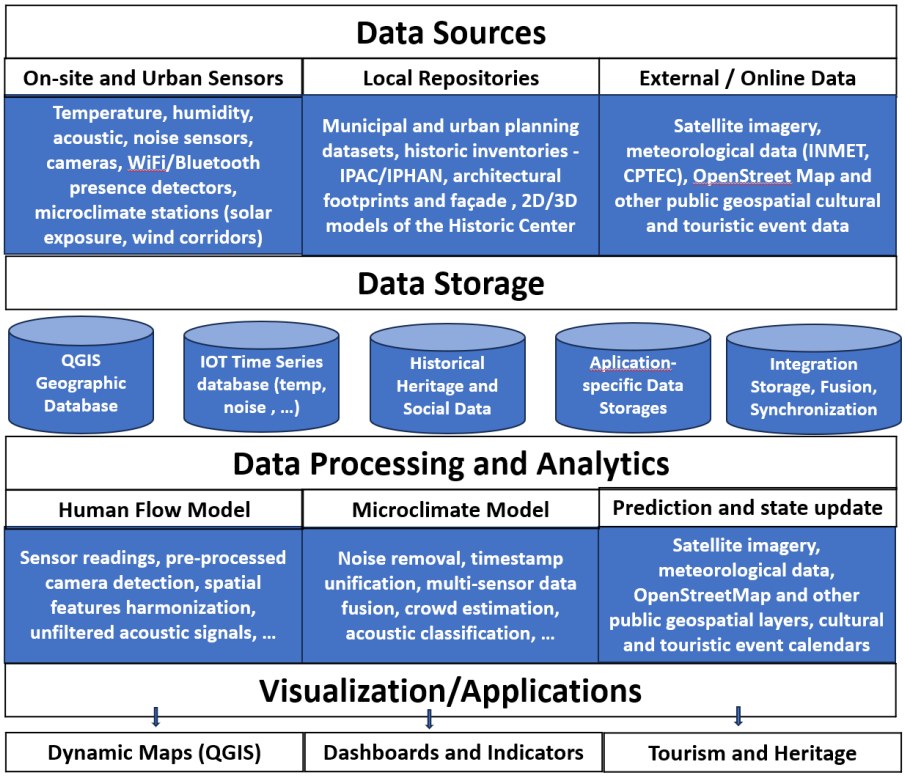

Figure 5: Pelourinho's Digital Twin Architecture Layers

data sources, a centralized processing core, and analytical models for simulating urban phenomena. The figure developed in the research's conceptual model illustrates this organization by presenting, in a hierarchical manner, the elements comprising the physical

environment, the data flows, and the computational layers that support the operation of the digital twin.

The architecture is structured in three main blocks. The first layer corresponds to the real world, where sensors and devices distributed throughout the Historic Center capture essential variables such as temperature, humidity, noise level, people flow, and microclimatic conditions. This data is complemented by institutional digital sources, such as the urban registry; heritage inventories from IPAC (*Instituto do Patrimônio Artístico e Cultural da Bahia*)[1] and IPHAN (*Instituto do Patrimônio Histórico e Artístico Nacional na Bahia*); and existing 2D/3D models, as well as external resources, including satellite images, national meteorological data, and public geospatial databases. This layer is characterized by the diversity of formats, resolutions, and periodicities, which reinforces the need for a robust integration mechanism.

The second layer represents precisely this mechanism: the Digital Twin Data Hub, which constitutes the core of the data-centric strategy. Inspired by the logic presented in data-centric architectures for heritage preservation, the Pelourinho Data Hub integrates three essential components:

• The raw data zone, which directly receives sensor records, camera detections, acoustic data, and pre-processed spatial features;

• The processing and harmonization pipeline, responsible for noise removal, inconsistency correction, temporal synchronization, georeferencing via QGIS base layers, and merging multiple sources, especially for estimating population density;

• The structured repositories, composed of geographic databases, time series databases, and application-specific datastores. This set is the organizing element of the entire architecture, as it ensures spatial, temporal, and semantic coherence to the data used by the digital twin models.

The third layer comprises analytical models and the simulation core, in which structured data are transformed into dynamic representations of the territory. At this level, human flow, microclimate, and sound-propagation models operate, using the Data Hub's geospatial and temporal databases to generate maps, indicators, forecasts, and simulated scenarios. This layer also includes predictive analytics modules and continuous updating mechanisms, ensuring that the digital twin reflects the most recent state of Pelourinho.

Finally, the top layer of visualization and applications provides dynamic maps, dashboards, and interfaces that support urban monitoring, cultural heritage management, and planned intervention in public space.

This organization demonstrates that the data-centric strategy is not an additional element, but the structural axis that articulates all the components of the architecture. While the real world provides diverse and fragmented data, and analytical models require consistent databases to function, it is in the Data Hub that the critical mediation between these two dimensions occurs.

The overall architecture is sustainable only because the data-centric strategy ensures that data is received, structured, spatialized, and integrated, enabling the digital twin to operate with precision, continuous updates, and simulation capabilities. In summary, the presented proposal encompasses all the necessary elements: sensors, institutional databases, processing, models, and applications, and illustrates how data centralization is the fundamental component that sustains the Pelourinho Digital Twin.

[1]IPAC and IPHAN are governmental institutions in Bahia and Brazil that lead Cultural Heritage across the state and country.

4 FINAL CONSIDERATIONS

Specialized digital twins are a trend in cultural heritage and tourism management in smart cities and digital transformation contexts, where over-tourism and the preservation of cultural heritage assets must be planned and enforced.

A data-rich approach is a must for specialized digital twins. The QGIS map-based data-centric approach developed for Pelourinho's DT enables the integration of heterogeneous data—spatial, temporal, and semantic—into a central repository, ensuring consistency, scalability, and flexibility for diverse uses of the Digital Twin.

In the case of Salvador Historic Center (Pelourinho), the data-rich approach embedded in the Pelourinho's DT architecture facilitated the management and decision-making processes of managers involved in the tourism economy and in the preservation of cultural heritage assets, activities that are typically distributed over diverse institutions.

Future work includes developing a new set of services and applications for the Salvador Historic Center (Pelourinho) for tourism and cultural heritage, and mitigating the impact of social and cultural events in Pelourinho.

In the context of tourism, we have examples such as urban heat-island modeling, climate-risk scenarios, real-time visitor flow monitoring, and dynamic route recommendation systems. In the context of cultural heritage and monument preservation, we have examples such as façade degradation forecasting based on heat stress, digital inventory of heritage assets, simulation of restoration scenarios, and green areas planning and deployment.

ACKNOWLEDGMENTS

The authors thank the ANIMA Institute for scholarship support 2025/2026.

REFERENCES

Afif Supianto, A., Nasar, W., Margrethe Aspen, D., Hasan, A., Karlsen, A. S. T., & Torres, R. D. S. (2024). An Urban Digital Twin Framework for Reference and Planning. *IEEE Access*, *12*, 152444–152465. https://doi.org/10.1109/ACCESS.2024.3478379

Akyol, G., & Avci, A. B. (2025). Digital twins in heritage conservation and visitor engagement: Comparative case studies from four historic sites. *Periodica Polytechnica Architecture*. https://doi.org/10.3311/PPar.40513

Almeida, D. S. d., Abreu, F. B. e., & Boavida-Portugal, I. (2025). Digital twins in tourism: A systematic literature review. https://doi.org/10.48550/arXiv.2502.00002

Bacelar, J. (2025). *Guia geográfico - turismo no brasil e no mundo* [Guia geográfico - salvador - bahia - pelourinho]. Retrieved December 14, 2025, from http://www.bahia-turismo.com/index.htm

Council, W. T. a. T. (2025). Travel & Tourism Economic Impact 2024: Global Trends. Retrieved June 10, 2025, from https://wttc.org/research/economic-impact

Dang, X., Liu, W., Hong, Q., Wang, Y., & Chen, X. (2023). Digital twin applications on cultural world heritage sites in china: A state-of-the-art overview. *Journal of Cultural Heritage*, *64*, 228–243. https://doi.org/10.1016/j.culher.2023.10.005

de Souza, L. A., & Martins, J. S. B. (2025). Cidades inteligentes e gêmeos digitais: Solução inovadora para o planejamento do centro histórico de salvador. *Anais da XII Semana de Análise Regional e Urbana (SARU)*, 1–6.

ERCIM, E. C. i. I. T.
bibinitperiod A. M. (2025). AI for cultural heritage. *ERCIM News*, (141), 44.

Farid, A. M., Alshareef, M., Badhesha, P. S., Boccaletti, C., Cacho, N. A. A., Carlier, C.-I., Corriveau, A., Khayal, I., Liner, B., Martins, J. S. B., Rahimi, F., Rossetti, R., Schoonenberg, W. C., Stillwell, A., & Wang, Y. (2021). Smart City Drivers and Challenges in Energy and Water Systems. *IEEE Potentials*, *40*(1), 6–10.

Martins, J. S. B., Perin, A. O., Castro, H. U., & Filho, E. B. O. (2024). Cidade Inteligente na Contemporaneidade com Internet das Coisas e Inteligência Artificial. *Gestão & Planejamento - G&P*, *25*(1). https://doi.org/10.53760/g&p

Mazzetto, S. (2024). A Review of Urban Digital Twins Integration, Challenges, and Future Directions in Smart City Development. *Sustainability*, *16*(19), 8337. https://doi.org/10.3390/su16198337

Menaguale, O. (2023). Digital twin and cultural heritage – the future of society built on history and art. In N. Crespi, A. T. Drobot, & R. Minerva (Eds.), *The digital twin* (pp. 1081–1111). Springer International Publishing. https://doi.org/10.1007/978-3-031-21343-4_34

Salunke, S. V., & Ouda, A. (2024). A performance benchmark for the PostgreSQL and MySQL databases. *Future Internet*, *16*(10), 382. https://doi.org/10.3390/fi16100382

Sánchez-Martín, J.-M., Guillén-Peñafiel, R., & Hernández-Carretero, A.-M. (2025). Artificial intelligence in heritage tourism: Innovation, accessibility, and sustainability in the digital age [Publisher: Multidisciplinary Digital Publishing Institute]. *Heritage*, *8*(10), 428. https://doi.org/10.3390/heritage8100428

Santos, J. P. B., França, L. C., Lima, B. L., Reis, R. B., Spínola, C., & Martins, J. S. B. (2025). Evidence analysis of tourism and geographic location correlation with syphilis incidence [Publisher: Routledge]. *Current Issues in Tourism*, 1–20.

Santos-Júnior, A., Almeida-García, F., Morgado, P., & Mendes-Filho, L. (2020). Residents' quality of life in smart tourism destinations: A theoretical approach [Publisher: Multidisciplinary Digital Publishing Institute]. *Sustainability*, *12*(20), 8445. https://doi.org/10.3390/su12208445

Serbouti, I., Chenal, J., Tazi, S. A., Baik, A., & Hakdaoui, M. (2025). Digital transformation in african heritage preservation: A digital twin framework for a sustainable bab al-mansour in meknes city, morocco. *Smart Cities*, *8*(1), 29. https://doi.org/10.3390/smartcities8010029

Shao, J., & Min, B. (2025). Sustainable development strategies for smart cities: Review and development framework. *Cities*, *158*, 105663. https://doi.org/10.1016/j.cities.2024.105663

Siegl, P., Buchty, R., & Berekovic, M. (2016). Data-centric computing frontiers: A survey on processing-in-memory. *Proceedings of the Second International Symposium on Memory Systems*, 295–308. https://doi.org/10.1145/2989081.2989087

Zha, D., Bhat, Z. P., Lai, K.-H., Yang, F., Jiang, Z., Zhong, S., & Hu, X. (2025). Data-centric artificial intelligence: A survey. *ACM Comput. Surv.*, *57*(5), 129:1–129:42. https://doi.org/10.1145/3711118

Received 20 December 2025; revised 30 January 2026; accepted XX XXXXX 2026

Giselle: A RAG-Based Generative AI Platform for Mental Health Care

Fabio Jose Gomes de Sousa[1], Ronaldo Fernandes Ramos[2], César Olavo de Moura Filho[3], Ivana C. H. Cunha Barreto[4], Luiz Odorico Monteiro de Andrade[5], and Antonio Mauro Barbosa de Oliveira[6]

[1] Federal Institute of Ceara, Maracanaú, Ceara, Brasil
fabio.jose@ifce.edu.br
[2] Federal Institute of Ceara, Fortaleza, Ceara, Brasil
ronaldo@ifce.edu.br
[3] Federal Institute of Ceara, Fortaleza, Ceara, Brasil
cesar.olavo2011@gmail.com
[4] Fundação Oswaldo Cruz, Eusébio, Ceara, Brasil
ivana.barreto@fiocruz.br
[5] Fundação Oswaldo Cruz, Eusébio, Ceara, Brasil
odorico.monteiro@fiocruz.br
[6] Federal Institute of Ceara, Fortaleza, Ceara, Brasil
mauro@ifce.edu.br

The launch of GPT-3 by OpenAI in 2020 revolutionized the field of Natural Language Processing (NLP) by introducing highly advanced generative AI models capable of performing complex tasks. This breakthrough accelerated the global adoption of artificial intelligence in various fields, including automation, customer support, and content creation.However, in the healthcare sector, the use of generative models requires caution, as errors or "hallucinations"can have serious consequences. A promising approach to improving the safety and accuracy of these tools is Retrieval-Augmented Generation (RAG), which enables the model to access external information in real time, ensuring up-to-date content, solution customization, and greater transparency—without the need for retraining—while significantly reducing the risk of hallucinations.In this context, this study explores the application of RAG in the Giselle Saúde project, an initiative funded by Embrapii/MCTI, which leverages generative AI to detect signs of depression in the elderly.

1 Introduction

In 2020, OpenAI introduced the Generative Pre-Training Transformer 3 (GPT-3), marking a major milestone in the field of Natural Language Processing (NLP) [3]. This model fundamentally advanced the state of the art by exhibiting the capability to comprehend and execute a wide range of linguistic tasks with minimal or no task-specific training data (few-shot and zero-shot learning), thereby achieving performance comparable to, and in some cases surpassing, that of the most advanced supervised artificial intelligence systems available at the time. [2]

Following this milestone, there was a rapid expansion in the adoption of generative artificial intelligence across multiple sectors, driving the development of new tools, process automation, and the enhancement of services. However, for large language models (LLMs) to operate effectively, the formulation of well-structured and contextually appropriate prompts is essential [6]. This challenge becomes particularly critical in the healthcare domain, where errors or hallucinations generated by conversational agents may lead to severe consequences. [8]

Among the strategies proposed to improve the quality and reliability of LLM outputs, Retrieval-Augmented Generation (RAG) stands out for four primary reasons: (1) it enables the integration of external knowledge sources, allowing access to information not included in the model's original training data; (2) it facilitates system governance by enabling straightforward addition, removal, or updating of content [7]; (3) it reduces operational costs by eliminating the need for model retraining to achieve customization; and (4) it enhances transparency by supporting the traceability of information sources, thereby mitigating the occurrence of hallucinations [1] [5].

In this context, this study proposes the adoption of the Retrieval-Augmented Generation (RAG) technique as a prompt engineering strategy to mitigate hallucinations in large language models (LLMs), within the scope of a research project funded by Embrapii/MCTI that employs generative artificial intelligence for the detection of depressive symptoms in older adults.

2 Releated Work

The RAG2 approach aims to reduce the semantic gap between retrievers and artificial intelligence models, as well as retrieval systems, arising from differences in their training objectives and architectural designs [9]. This discrepancy leads LLMs to passively accept the documents provided by retrievers, which may result in misunderstandings during the generation process. To address this limitation, the authors propose RAG2 as an enhanced framework that directly incorporates retrieval-related information into the generation process. Specifically, RAG2 leverages fine-grained retriever features and employs an R2-Former to effectively capture such information. In addition, a retrieval-aware prompting strategy is introduced to integrate these signals into LLM generation. Notably, RAG2 is particularly well suited for low-resource scenarios, in which both the LLMs and the retrievers remain unchanged.

The work of Patrick Lewis [4] highlights his seminal contribution to the development of the Retrieval-Augmented Generation (RAG) technique, which enhances the accuracy of artificial intelligence models by enabling access to external textual sources, such as corporate documents or news websites. This approach contributes to the mitigation of errors commonly referred to as 'hallucinations' and facilitates access to up-to-date information. In recent years, major technology companies, including Microsoft, Google, Amazon, and NVIDIA, have adopted this technique. Currently serving as Director of Machine Learning at Cohere, Lewis has continued to advance RAG methodologies by promoting mechanisms that ensure AI systems explicitly cite their sources, thereby enabling human verification of generated information and fostering comprehensive traceability of knowledge.

It is important to note that the RAG implementation adopted in the Giselle project relied on documents previously curated by the healthcare team and was developed using the Python programming language, in conjunction with the LangChain framework and the ChromaDB vector database. This approach may be considered a traditional implementation, given that, at the time of development, there was a limited availability of specialized frameworks for RAG-based systems. However, it is worth highlighting that this RAG will be an important feature of the Giselle health platform to mitigate hallucinations from LLM, working in conjunction with a prompt developed by the health team.

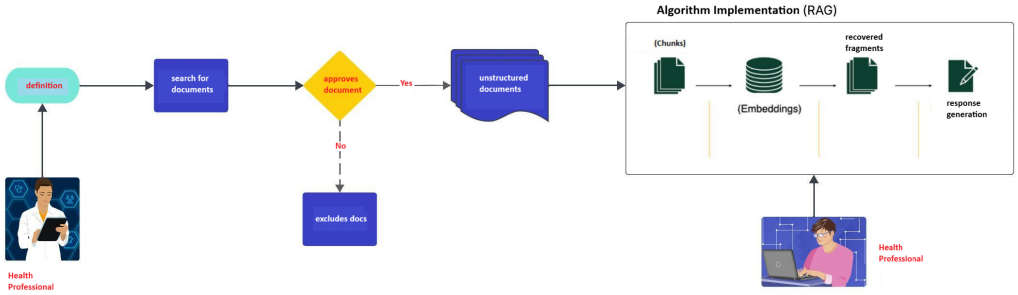

Figura 1: Flowchart of the RAG Development Process.

3 Proposed Methodology

The Giselle Saúde project adopts a methodological approach centered on the application of generative artificial intelligence to support the identification and monitoring of depressive indicators in older adults. The system is implemented as a conversational agent designed to conduct empathetic and supportive dialogues, guided by a specialized prompt developed collaboratively with healthcare professionals.

A central methodological challenge involves the continuous refinement of this prompt to ensure response accuracy and to mitigate hallucinations produced by the large language model (LLM). To address this issue, the Retrieval-Augmented Generation (RAG) technique was employed, integrating information retrieval mechanisms with natural language generation to enhance the contextual relevance and factual consistency of model outputs.

The RAG pipeline was constructed using a curated corpus of domain-specific documents, previously selected and validated by healthcare professionals. These documents were embedded and indexed in a vector database, enabling semantic retrieval of relevant content in response to user queries. The retrieved information was then incorporated into the prompt provided to the LLM, allowing the generation process to be grounded in authoritative and up-to-date sources.

The development and implementation of the RAG-based system followed a multidisciplinary methodology, involving close collaboration between healthcare experts and information technology professionals. This collaborative process ensured both clinical relevance and technical robustness, supporting the deployment of a reliable and transparent AI-driven solution for mental health support among older adults. Figure 1 shows the flowchart for this work.

3.1 Scope Definition

The work began with the definition of the scope of the documents to be retrieved from scientific databases and official platforms related to the topic, which, in this study, focuses on the mental health of older adults.

3.2 Document Search

After defining which documents were relevant for the construction of the database, the healthcare team conducted a systematic search across multiple data sources, including scientific databases (articles, dissertations, and theses), academic journals, and official websites addressing mental health, social well-being, and geriatric depression scales for older adults.

```python
# Iterar sobre todos os arquivos PDF no diretório fornecido
for file in glob.glob(f"{directory}/*.pdf"):  # Ajuste para '*.pdf'
    try:
        loader = PyMuPDFLoader(file)
        docs = loader.load()
        print(f"Carregado o arquivo {file} com sucesso.")
    except Exception as e:
        print(f"Erro ao carregar o arquivo {file}: {e}")
        continue

    try:
        text_splitter = CharacterTextSplitter(chunk_size=1000, chunk_overlap=200)
        splits = text_splitter.split_documents(docs)
        document_splits.extend(splits)
        print(f"Dividido o arquivo {file} em {len(splits)} partes.")
    except Exception as e:
        print(f"Erro ao dividir o documento {file}: {e}")
        continue
```

Figura 2: Loading and Breaking Documents into Chunks.

3.3 Document Approval

Once the documents related to the topic were identified, the healthcare team performed a rigorous selection process, retaining only the most relevant materials and discarding those deemed less pertinent to the study objectives.

3.4 Structured Documents

The subsequent step involved incorporating each selected document into an unstructured database, which would later be used in the RAG generation process.

3.5 RAG Algorithm Implementation

After assembling the unstructured dataset on older adults' mental health—curated by the healthcare team—the information technology team initiated the implementation of the RAG system. Initially, the technological tools to be employed were defined, including the GPT-3.5 Turbo API, the Python programming language, the ChromaDB vector database, and the LangChain and Streamlit libraries.

At this stage, the algorithm loads all documents in PDF format and subsequently divides them into smaller segments (chunks) to enable vectorization and storage in the database. Figure 2 illustrates the code segment responsible for this process, using a chunk-size of 1000 characters and a chunk-overlap of 200 characters. This configuration ensures that each text segment contains sufficient contextual continuity, allowing the model to maintain semantic coherence during data processing.

```python
def create_persistence(directory="textos"):        You, há 7
    document_splits = []

    # Verificar se o diretório existe
    if not os.path.exists(directory):
        print(f"Diretório {directory} não encontrado.")
        return

    # Configurar o banco de dados Chroma
    settings = Settings(
        anonymized_telemetry=False,
        is_persistent=True,
        persist_directory=".chroma/",
    )
```

Figura 3: Document Persistence Function.

3.6 ChromaDB Vector Database

Figure 3 illustrates the persistence function, which defines the storage directory and the document vector container (document splits = [ ]) used to receive the loaded documents, as well as the configuration parameters of the ChromaDB vector database.

3.7 Retrieved Chunks

At this stage, the system makes the data available in numerical (vectorized) form for querying. For each user query, the most relevant text chunks are retrieved based on semantic similarity measures, enabling efficient and context-aware information retrieval.

3.8 Response Generation

In the final stage, the large language model (LLM) generates responses grounded in the information retrieved in the previous step, producing outputs that are more accurate and contextually informed. Figure 4 presents the main execution function (main), which imports the rag function—responsible for implementing the preceding stages—as well as the Streamlit library, used to provide a graphical interface for displaying user queries, generated responses, and the associated contextual information.

```python
import streamlit as st
import rag


if prompt := st.chat_input("Escreva sua pergunta?"):
    # Display user message in chat message container
    st.chat_message("user").markdown(prompt)

    with st.spinner("Aguardando resposta ..."):
        ctx, response = rag.generate_response(prompt)

    with st.expander("Contexto armazenado"):
        st.write(ctx)

    st.chat_message("assistant").markdown(response)
```

Figura 4: Main function - streamlit.

4 Results and Discussion

By leveraging the RAG approach, the Giselle Saúde system queries a specialized knowledge base whenever a user question is received, generating grounded and context-aware responses. This strategy eliminates the need for retraining the large language model (LLM), thereby making the adaptation process more agile and efficient.

Figure 5 presents the architecture of the Giselle platform, updated to incorporate the RAG approach, as described in the following steps:

4.1 Prompt Service

This microservice manages the specialized prompt employed by the conversational agent. It interfaces with a dedicated database (DB) that stores prompt versions, dialogue rules, and parameters defined by healthcare professionals. The outputs generated by this service can be consolidated into reports, which are subsequently delivered to healthcare professionals to support clinical monitoring and assessment.

4.2 Retrieval-Augmented Generation (RAG) Service

The RAG service enhances the quality and reliability of language model responses by retrieving relevant information from a vector database (DB). By grounding the generation process in domain-specific and curated knowledge sources, this service improves contextual relevance and reduces the incidence of hallucinations. Similar to the Prompt Service, the RAG component contributes to the generation of structured reports for healthcare professionals.

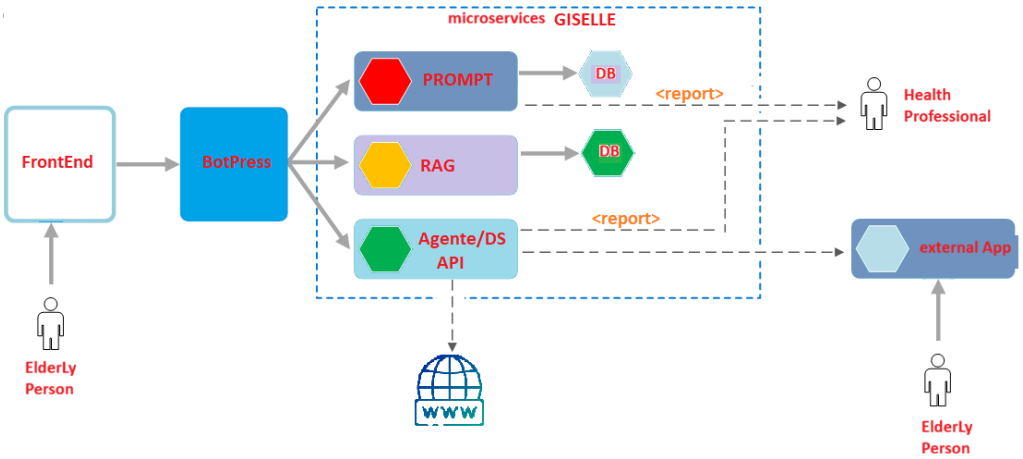

Figura 5: Giselle's microservices architecture

4.3 Agent / Data Services (DS) API

This microservice functions as an integration and mediation layer, enabling communication with external applications and resources available on the web (WWW). It facilitates interoperability with third-party systems, allowing both data ingestion and dissemination. External applications may also interact with older adults through this service, expanding the ecosystem of care.

Healthcare professionals receive analytical reports produced by the microservices, enabling continuous monitoring of mental health indicators and supporting informed clinical decision-making. In parallel, external applications can exchange information with the platform, maintaining the older adult at the center of a connected and supervised care ecosystem.

Figure 6 presents as part of the experimental evaluation, the Giselle system enhanced with the RAG approach was queried regarding factors associated with depression in older adults. The system generated consistent and contextually grounded responses based on the curated document corpus, explicitly indicating the sources from which the information was retrieved. These results were positively evaluated by the healthcare team, comprising physicians, a psychologist, a geriatrician, and a nutritionist.

5 Conclusions

The ability of the Retrieval-Augmented Generation (RAG) approach to incorporate new data in real time, without the need for costly model retraining, demonstrates its effectiveness in enhancing the accuracy and reliability of generated responses. Based on these findings, the Giselle Saúde platform emerges as a robust and scalable solution for supporting older adults' mental health, enabling more informed, adaptive, and human-centered interactions. These results highlight the potential of RAG-based architectures to improve the trustworthiness and applicability of generative artificial intelligence systems in sensitive healthcare contexts.

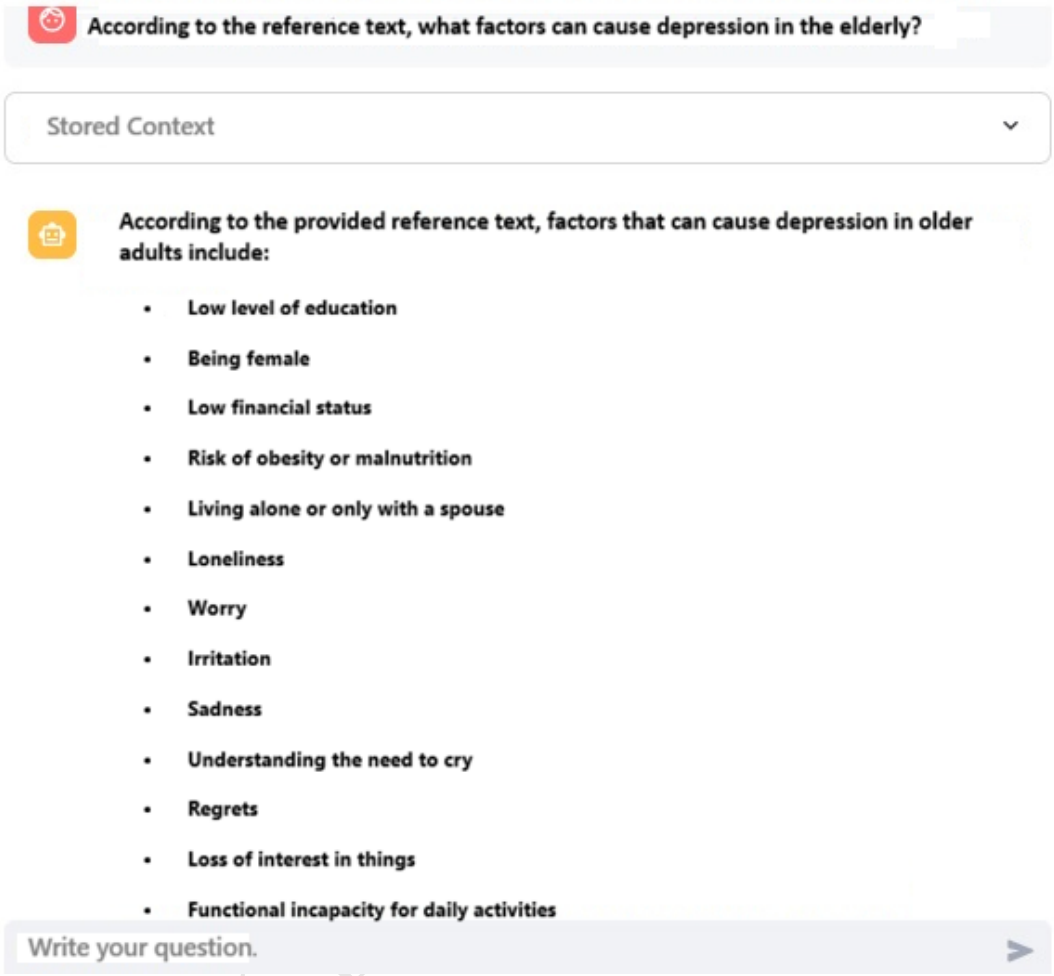

Figura 6: System query using RAG

6 Future Work

This RAG component will be migrated to a new integration platform, n8n, which will enable the Giselle Saúde chatbot to query the knowledge base via an API whenever required, with improved agility and reliability. The system architecture has already been defined, and the update process is currently underway. In addition to RAG, the Giselle platform has a prompt, developed by the healthcare team using modern prompt engineering techniques, and, in the future, a dataset to minimize or elucidate hallucinations in LLM.

Referências

[1] John W. Ayers, Adam Poliak, Mark Dredze, et al. Comparing physician and artificial intelligence chatbot responses to patient questions posted to a public social media forum. *JAMA Internal*

Medicine, 183:589–596, 2023.

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, et al. Language models are few-shot learners. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, Vancouver, Canada, 2020.

[3] I., P. et al. Enhancing speech emotion recognition using dual feature extraction encoders. *Sensors*, 23(14), 2023.

[4] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Veselin Stoyanov. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

[5] F. R. and A. P. Emotion detection for supporting depression screening. *Multimedia Tools and Applications*, 82(9):12771–12795, 2023.

[6] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, et al. A systematic survey of prompt engineering in large language models: Techniques and applications. Technical report, Department of Computer Science and Engineering, Indian Institute of Technology Patna, 2024.

[7] J. Singh et al. Attention-enabled ensemble deep learning models and their validation for depression detection: A domain adoption paradigm. *Diagnostics*, 13(12), 2023.

[8] L. Tran, J. Smith, and H. Nguyen. Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 10(1):1234, 2020.

[9] Fangzhou Ye, Shuang Li, Yifan Zhang, and Lei Chen. R^2ag: Incorporating retrieval information into retrieval augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11584–11596, 2024.

Sk-Iterative: A Greedy Scheduling Algorithm with Spatial Reuse for Dense Wireless Networks

Chrystopher N. Bravos
Elias Procópio Duarte Jr.
{cnb18,elias}@inf.ufpr.br
Federal University of Parana (UFPR)
Curitiba, Paraná, Brazil

Fábio Engel de Camargo
fabioe@utfpr.edu.br
Federal University of Technology – Parana (UTFPR)
Toledo, Paraná, Brazil

Flávio Assis
fassis@ufba.br
Universidade Federal da Bahia (UFBA)
School of Computing
Salvador, Bahia, Brazil

Abstract

The density of wireless networks has been consistently increasing. Dealing with an increasing number of devices per area unit is a pressing issue in the context of the Internet of Things (IoT), as well as in cellular networks (5G and B5G). The Signal-to-Interference-plus-Noise Ratio (SINR) model is particularly relevant in this context, as it facilitates spatial reuse, which allows multiple devices to transmit simultaneously within the same coverage area. This model considers the cumulative interference from competing transmissions, enabling scheduling that maximizes simultaneous communications. Given that the scheduling problem in SINR networks is NP-hard, heuristics are necessary for practical solutions. This work introduces Sk-Iterative, a greedy scheduling algorithm with spatial reuse to solve the problem. The algorithm schedules links produced with the DTE (Down-To-Earth) heuristics. Sk-Iterative was implemented and evaluated via simulation. Results confirm the efficiency of the scheduling strategy, showing that it produces schedules that are close to the optimal.

Keywords

Wireless Networks, Spatial Reuse, Link Scheduling, Greedy Heuristics, Dense Wireless Networks

1 Introduction

There is currently a clear trend toward dense wireless networks [28]. This is true in the Internet of Things [8, 11], in 5G and B5G cellular networks [4], sensor networks [23] and several other contexts [21]. Both reliability and efficiency are at stake in a very dense network [22]: as the transmission space is shared, the result is increased interference between simultaneous transmissions. One common way to address this issue is by separating transmissions into distinct time slots using TDMA (Time Division Multiple Access) scheduling. In its traditional form, TDMA scheduling only allows for a *single* transmission per time slot [27].

The Signal-to-Interference-plus-Noise Ratio (SINR) model accounts for the physical effects of transmissions to represent cumulative interference [1, 10, 20, 24]. This model incorporates factors like path loss, background noise, and mutual interference. It enables "spatial reuse", where simultaneous transmissions can coexist within the same time slot, as long as those transmissions do not exceed a predefined interference threshold. Nonetheless, scheduling with spatial reuse under the SINR model is proven to be NP-complete [13]. As a result, heuristics are needed to compute feasible schedules in practice.

The Down-to-Earth (DTE) scheduling approach, proposed by [3], is a heuristic aimed at enabling spatial reuse in wireless networks by generating a set of links that connect all devices, giving each and every device the same opportunity to transmit and receive messages. This process consists of two stages: (i) devices broadcast their positions and identifiers within the network, using a traditional method that assigns one device to a single time slot; (ii) each device identifies its closest neighbor, calculates the minimum transmission power needed to communicate safely, and adds additional links to ensure a strongly connected transmission graph, which guarantees directed paths between all devices. The scheduling algorithm then processes these links to assign them to consecutive time slots, with the primary goal of minimizing the total number of time slots used to enhance communication opportunities by maximizing spatial reuse.

This paper introduces the Sk-Iterative algorithm, a greedy heuristic algorithm for scheduling with spatial reuse under the SINR model. The Sk-Iterative algorithm takes as input the set of directed links generated by the DTE heuristic. The algorithm outputs a schedule of the links, assigning them to consecutive time slots. Using a greedy strategy, the algorithm first identifies a set of "candidate" links that can be scheduled. It then expands this set by selecting additional links to ensure that all links are assigned to at least one candidate time slot. Finally, a Set Covering Problem (SCP) algorithm is applied to the candidate time slots, producing the final schedule.

The Sk-Iterative algorithm is evaluated through simulations. In experiments conducted on networks where devices are randomly distributed within a Euclidean plane,Sk-Iterative was able to reduce the schedule size by up to 90% in networks with 100 devices, compared to scheduling without spatial reuse. When compared to the optimal algorithm, Sk-Iterative produced results that were very close to the best possible, typically requiring only one additional time slot on average.

The remainder of this work is organized as follows. Section 2 describes the SINR model. Section 3 introduces the DTE heuristic. Section 4 provides a detailed description of the Sk-Iterative algorithm. Section 5 presents the simulation results, including comparisons with other algorithms. Finally, Section 6 concludes the paper.

2 The SINR Model

Devices in a wireless network communicate over a shared physical medium and are subject to various interferences, both from the environment and from competing transmissions [16]. Unlike point-to-point networks, where all processes can communicate simultaneously, wireless networks require a medium access control strategy to enable communication. One of the most commonly

used strategies is TDMA, which schedules transmissions across distinct time intervals. In its traditional form, only one transmission is assigned per time slot.

To achieve a more accurate abstraction of real transmission behavior in wireless networks, the physical interference model known as Signal-to-Interference-plus-Noise Ratio (SINR) has been proposed. The SINR model accounts for physical effects on transmissions, allowing for the development of efficient scheduling algorithms. Specifically, it calculates the interference among multiple potential transmissions, considering factors such as path loss and background noise to enable multiple simultaneous transmissions within the same time slot—this is known as spatial reuse.

It is intuitive to understand that a signal emitted from a source weakens as it travels through the medium. More specifically, the path loss of the power of the transmitted signal is inversely proportional to the distance it travels [26]. Equation 1 illustrates the relationship between the transmitted and received power in relation to distance, where i and j represent the transmitter and receiver, respectively, $(d(i, j))$ is the distance between the devices, and (P_T) is the transmitted power.

$$P_{ij} = \frac{P_T}{d(i,j)^\alpha} \tag{1}$$

The equation also includes constant α, known as the path-loss exponent. This constant varies depending on the communication medium, and empirical measurements have shown that it can range from 1.6 to 6 in indoor environments and from 2 to 4 in urban settings [12].

The sensitivity of the receiver's antenna is a crucial factor for effective communication. Proper data reception requires that the signal reaches the receiver with a predefined minimum intensity. This value, typically expressed in decibel-milliwatts (dBm), ranges from -85 dBm as established by the IEEE 802.11g standard to -120 dBm for more recent receivers [19].

Note that just exceeding the receiver's sensitivity is not enough. A signal experiences various fluctuations due to unwanted interference that cannot be controlled, known as background noise [3]. This noise, inherent to the environment, is typically treated as a random constant N_0, expressed in decibels.

To achieve efficiency, communications must be coordinated to avoid mutual interference, which hinders proper signal reception [12]. The total interference P_I is the accumulation of all interferences caused by simultaneous transmissions at a receiver i. Equation 2 demonstrates how to calculate the total interference for a network with τ devices transmitting simultaneously, where i and j denote the transmitter and receiver, respectively.

$$I_i = \sum_{\substack{k \in \tau, \\ k \neq i,j}} \frac{P_{T_k}}{d(k,i)^\alpha} \tag{2}$$

For example, consider the scenario depicted in Figure 1. In this situation, three devices, a, b, and c, are transmitting simultaneously with a power output of 1mW to their respective receivers d, e, and f, all within an urban environment ($\alpha = 4$). Table 1 presents the partial and total interference calculated for each transmission.

As observed, the distance between devices is a key factor in determining the level of interference. In other words, the closer

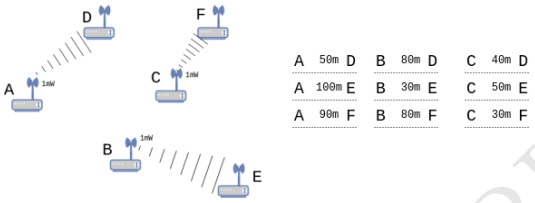

Figure 1: Parallel communication scenario.

Table 1: Mutual interference computed for the example in Figure 1.

	I_a	I_b	I_c	P_I
$a \rightarrow d$	–	$\cong$ 2.4e−08mW	$\cong$ 3.9e−07mW	$\cong$ 4.1e−07mW
$b \rightarrow e$	1e−08mW	–	1.6e−07mW	$\cong$ 1.7e−07mW
$c \rightarrow f$	$\cong$ 1.5e−08mW	$\cong$ 2.4e−08mW	–	$\cong$ 3.9e−08mW

the devices are to each other, the greater the interference they cause. The power used is also crucial for communication; thus, it is important to choose a transmission power that is sufficient for effective communication (i.e., reception by the receiver) while also maintaining an acceptable level of parallel transmissions.

The SINR model takes into account the three properties of signals mentioned earlier (path loss, background noise, and mutual interference) to determine whether the communication will be successful. Equation 3 illustrates how to compute the SINR value for a communication between transmitter i and receiver j in the presence of τ simultaneous transmissions.

$$SINR(i,j) = \frac{\frac{P_{T_i}}{d(i,j)^\alpha}}{N_0 + (\sum_{\substack{k \in \tau, \\ k \neq i}} \frac{P_{T_k}}{d(k,i)^\alpha})} \tag{3}$$

The resulting SINR value must be lower than the so-called SINR threshold for the communication between i and j to be successful. The SINR threshold, denoted by γ, is expressed in decibels (dB). Thus, a transmission occurs correctly if the SINR value is at least equal to γ. The SINR threshold varies for each device, as it depends on factors such as antenna capability and processing power, which are linked to the technology used. Therefore, a successful communication, according to the SINR model, satisfies the inequality shown in Equation 4.

$$SINR(i,j) \geqslant \gamma \tag{4}$$

For defining the transmission power to be employed by the devices, the most common strategy is oblivious power assignment. This strategy establishes the power levels that are employed beforehand [27]. The specific value for the power to be assigned can be computed in three different ways, described next. The uniform assignment is the simplest approach, where all devices are assigned the same power level. The linear assignment, on the other hand, determines the power level based on the path loss between the transmitter and receiver. Finally, the square root assignment computes the power in proportion to the square root of the path loss.

Equation 5 computes the transmission power P_T, ensuring that it meets the Signal-to-Interference-plus-Noise Ratio (SINR) threshold (*gamma*), computed as shown in Equation 4. Equation 5 also employs the background noise (N_0), and path loss ($d(i,j)^\alpha$), which characterizes a linear power assignment. However, the computed value does not consider simultaneous transmissions, meaning that any interference would make concurrent communications impossible.

$$P_T = \gamma \cdot N_0 \cdot d(i,j)^\alpha \tag{5}$$

To address this issue, making simultaneous transmission in the same area possible, a margin value (β) is introduced into the expression. This margin increases the power to levels that are high enough to handle interference. The resulting expression is shown in Equation 6.

$$P_T = \gamma \cdot N_0 \cdot (d(i,j) + \beta)^\alpha \tag{6}$$

By applying the linear assignment power assignment strategy to the example in Figure 1, and considering an SINR threshold of 1 dB and a margin $\beta = 10$, the new values for transmitted power (P_T), received power (P_R), total interference (P_I), and the SINR ratio computed ($SINR$) are presented in Table 2.

Table 2: SINR values obtained with linear power assignment.

	P_T	P_R	P_I	$SINR$
$a \rightarrow d$	$\cong$ 1.6e−02mW	$\cong$ 2.56e−09mW	$\cong$ 1.33e−09mW	$\cong$ 0.40dB
$b \rightarrow e$	$\cong$ 3.2e−03mW	$\cong$ 3.95e−09mW	$\cong$ 6.72e−10mW	$\cong$ 3.73dB
$c \rightarrow f$	$\cong$ 3.2e−03mW	$\cong$ 3.95e−09mW	$\cong$ 3.2e−10mW	$\cong$ 4.76dB

Table 2 shows that the power values computed with the linear assignment strategy result in better SINR ratios for the three transmissions. As the value for each transmission is computed based on its corresponding path loss, the overall result is significantly more balanced in comparison with the uniform strategy shown previously, in which all devices are assigned the same value.

Time Division Multiple Access (TDMA) is a classical scheduling method that defines transmission slots for a given time interval. Several variations of the basic TDMA strategy have been proposed. One such variation is the Spatial Reuse TDMA (STDMA) [2], which allows multiple transmissions within the same interval while managing interference among devices. The goal of STDMA is to minimize the total number of time intervals needed for all devices to communicate, thereby maximizing spatial reuse.

The scheduling problem in the SINR model is NP-complete [13]. Thus there is no polynomial-time algorithm that can find the optimal schedule. As a result, various approximation algorithms have been proposed to solve the problem in practice.

Scheduling algorithms can be classified in multiple ways. They can be either topology-dependent, or topology-independent. They can also be distributed or centralized. In a distributed strategy the devices themselves compute the schedule. On the other hand, in the centralized strategy a single central unit computes the whole schedule. Additionally, scheduling algorithms may have different targets: some compute the scheduling for devices, while others focus on scheduling transmissions represented by links (pairs of devices that communicate).

The algorithm proposed in this work is classified as distributed (as it requires all devices to participate in computations), topology-dependent (it relies on network topology information for computing the schedule), and does link scheduling.

3 The Down-To-Earth Heuristic

The Down-To-Earth (DTE) heuristic determines the set of communication links to be scheduled. DTE also defines the transmission power for each device. Experimental evidence from the simulation of SINR scheduling algorithms indicates that enabling spatial reuse in dense wireless networks is very challenging [3]. DTE addresses this issue by restricting each device to communicate *only* with its nearest neighbor, which reduces interference and improves the potential for spatial reuse.

To further mitigate interference, each device transmits at the *minimum power* required to ensure reliable communication with its nearest neighbor, *augmented by a small safety margin* to tolerate concurrent transmissions. This margin reduces the likelihood of signal corruption due to additional interference, enabling spatial reuse when feasible. The links generated by DTE form a transmission graph that connects devices to their nearest neighbors while ensuring that all devices can both transmit and receive messages. Moreover, the resulting graph must be strongly connected, i.e., every device must have a directed path to every other device. Achieving this property may require adding additional edges beyond nearest-neighbor connections.

The DTE heuristic takes as input a wireless network composed of a set of devices distributed in the Euclidean plane. A 1-hop network is assumed, i.e., all devices lie within mutual coverage range and can therefore communicate directly with any other device. Note however that although any device *can* in principle communicate with any other other device, the DTE heuristic allows each node to communicate with a *single* device, the closest one. The purpose is to increase the chances that multiple devices will be able to communicate simultaneously in that 1-hop network.

The DTE heuristic can be described in two main steps. The first step enables each device to obtain the positions of all other devices. In this step, every device first broadcasts its identifier and location to the network. Initially, each device is aware only of its own identifier and spatial coordinates. We emphasize that the adopted system model is static: device membership is fixed over time and nodes are immobile. In dynamic settings, additional mechanisms for topology discovery [17, 18, 25] and monitoring [5–7] would be required. Under the static assumption, however, topology discovery can rely on standard CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), ensuring that each node can successfully disseminate its information. Once global position information is available, devices define their communication neighbors and adjust their transmission power accordingly.

The key intuition behind the Down-To-Earth (DTE) heuristic is to restrict each device i to communicate only with its nearest neighbor j in the plane. Accordingly, DTE initially adds a single directed

edge (i,j) from i to j to the transmission graph. However, relying exclusively on nearest-neighbor edges may leave some devices without outgoing (transmission) or incoming (reception) edges, preventing them from sending or receiving messages. Moreover, the resulting transmission graph must be strongly connected, i.e., there must exist a directed path between every ordered pair of vertices in the graph.

To enforce strong connectivity of the transmission graph, a Minimum Spanning Tree (MST) algorithm is executed. MST structures are widely used in distributed protocols, particularly those supporting network-wide information dissemination (e.g., broadcast), because they enable message propagation at minimal communication cost. Formally, an MST is a tree that spans all vertices of a weighted graph while minimizing the total sum of edge weights. Since devices have access to the global network topology, several efficient MST algorithms can be applied. In this work, we adopt Kruskal's algorithm [15].

In summary, the DTE heuristic first determines the set of communication links. These links ensure that every device (1) has at least one outgoing link, enabling that device to transmit messages, and (2) has at least one incoming link, enabling the reception of messages. The resulting set of links forms a strongly connected transmission graph. Subsequently, each transmitter adjusts its transmission power. Recall that this design exploits the fact that each device knows the positions of the neighbors to which it transmits.

At least one vertex will have two receivers: Since the proposed approach relies on tree, all non-leaf vertices in the tree necessarily have at least two outgoing edges: to its parent and its son in the tree. Thus that non-leaf vertex will make transmissions to those two (or more) receivers. In those cases, the power level is determined so that the communication with the the farthest receiver is possible, which ensures that the communication with the closest receiver will of course also succeed. The transmission power is then set to be linearly proportional to this distance. By assigning power according to Eq. 7, all outgoing transmission links are guaranteed to operate with sufficient power for reliable communication, using a single power level per device.

As mentioned above, the device sets to itself the minimum transmission power plus an additional margin, as shown in Eq. 7. This expression is derived from Eq. 4 plus of the margin, denoted by γ_{spare} in the equation. Thus γ_{spare} ensures that the computed power value is not exactly equal to the SINR threshold and allows spatial reuse.

$$P_T = (\gamma + \gamma_{spare}) \cdot N_0 \cdot d(i,j)^{\alpha} \tag{7}$$

The next step consists of scheduling the links defined in the transmission graph. A subset of links can be assigned to each time interval provided that their mutual interference does not prevent successful reception. Every link must be assigned to some interval. The resulting schedule is performed sequentially and repeated infinitely, enabling all devices to communicate over time, periodically. This strategy requires message routing to guarantee that transmissions originating from any device can reach any other device. An optimal solution would examine all feasible link combinations under the SINR model; however, this yields exponential complexity, making it impractical even for moderate values of n, the number of devices. The Sk-Iterative algorithm presented in the next section provides an efficient heuristic solution to this problem.

4 The Sk-Iterative Algorithm

This section introduces the Sk-Iterative algorithm (*Stochastic k-Iterative*) for link scheduling in dense wireless networks under the SINR model. The algorithm allocates transmissions to discrete time slots. Devices are assumed to be distributed over a bounded region in the Euclidean plane, and scheduling is performed over the directed links generated by the DTE heuristic. Each link must be assigned to at least one time slot, and the total number of slots is denoted by (t). Through spatial reuse, multiple links may share the same time slot, and the objective is to minimize t by maximizing concurrent transmissions. After t time slots, all transmitters have been scheduled at least once; the procedure is then repeated *ad infinitum*.

The Sk-Iterative algorithm takes as input a set L of links to be scheduled and a parameter k, which specifies the maximum number of simultaneous transmissions allowed per time slot, expressed as a percentage of the total number of links. Empirical evaluation using the optimal algorithm indicates that the maximum number of links assigned to a single time slot typically ranges from 10% to 20% of the total number of links. Accordingly, in this work, k is set to $0.2|L|$.

The algorithm consists of two phases. In the first phase, sets of up to k links that can be assigned to a single time slot are constructed. These sets, denoted by K_{set}, represent candidate slot assignments. In the second phase, the *Set Covering Problem* (SCP) is solved over the candidate assignments [14]. The Set Covering Problem (SCP) is a classic optimization problem that finds the smallest collection of subsets that contain every element of a larger "universe" set. Thus in the scheduling problem, the universe set is the set of all links, and the subsets are of links assigned for each time slot.

In the first phase, candidate links are selected randomly for each time slot. Initially, k links are drawn at random from L. The first link is assigned without restrictions, whereas each subsequent link is assigned only if its transmission remains successful under the interference generated by the links already placed in the slot. Interference feasibility is evaluated using the SINR model. If a link cannot be assigned, the next candidate link is tested. After all k links have been evaluated, the time slot is finalized and a new slot is constructed. This procedure is repeated for λ iterations, yielding λ candidate time slots.

To illustrate this process, consider a network with eight devices distributed within a 100×100 m^2 area. The scheduling is to be done with $k = 2$ devices per time slot. The parameters are set as follows: SINR threshold (γ) = 20 dB, threshold spare (γ_{spare}) = 50 dB, background noise ($N_0 = -90$ dBm), and path-loss exponent $\alpha = 4$. With $\lambda = 30$ iterations in the first phase, the candidate time slots generated by Sk-Iterative include as example $\{0 \rightarrow 5, 0 \rightarrow 3\}$, $\{3 \rightarrow 1, 4 \rightarrow 6\}$, $\{5 \rightarrow 2, 5 \rightarrow 0\}$, $\{5 \rightarrow 0, 5 \rightarrow 2\}$, $\{5 \rightarrow 2, 4 \rightarrow 6\}$, $\{5 \rightarrow 0, 5 \rightarrow 6\}$, $\{3 \rightarrow 0, 3 \rightarrow 1\}$, $\{0 \rightarrow 5, 0 \rightarrow 3\}$, and $\{6 \rightarrow 4, 6 \rightarrow 5\}$. It is possible to conclude that the algorithm does produce a diverse set of candidate combinations, even for a small network.

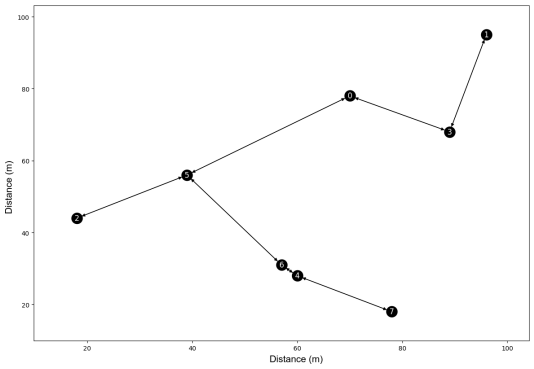

Figure 2: Example network with 8 devices.

Across the λ iterations, all links in L are considered as candidates for assignment into time slots. After these iterations, a second phase is executed, employing the *Stochastic k-Greedy* (Sk-Greedy) strategy [9] to generate additional candidate time slots for the final schedule. The reason for that second phase is that some links may have not been assigned to any time slot. The goal is to guarantee that every link is eventually assigned to at least one time slot.

The Sk-Greedy algorithm operates as follows. Recall that this phase assigns links that have not been assigned to any time slot in the previous phase. The first link is assigned without restrictions. Each subsequent link is assigned to a time slot only if its transmission remains successful under the interference generated by the links already placed in the same time slot. If a link cannot be assigned, the algorithm evaluates the next candidate. After all k links have been tested—whether assigned or not—the time slot is closed and a new one is created. A main difference to the Sk-Iterative approach is that links that have been assigned are removed from further consideration in subsequent candidate time slots. This procedure continues until all links have been allocated to at least one candidate time slot.

After the candidate time slots have all been defined, Sk-Iterative computes the final schedule by solving a *Set Covering Problem* (SCP). SCP is defined as follows: let $U = \{u_1, \ldots, u_m\}$ denote the universe of elements, and let $S = \{s_1, \ldots, s_n\}$ be a collection of subsets such that $s_i \subseteq U$ and $\bigcup s_i = U$. The objective is to find the smallest subcollection $X \subseteq S$ that covers U. Since the SCP is NP-hard, a heuristic method is adopted, namely the greedy strategy proposed by Grossman [14]. This approach iteratively selects the subset that covers the largest number of currently uncovered elements, adding it to the solution until all elements are covered. As a result, the final schedule covers all links initially defined for the transmission graph.

5 Simulation Results

The Sk-Iterative strategy was evaluated through simulation. This section reports the results regarding the size of the resulting schedule and compares it against other approaches.

First, the impact of the number of iterations (λ) on the schedule size is evaluated. The algorithm was executed on networks with 30, 50, and 100 devices, varying λ from 0.5 to 5 times the number of links to be scheduled, in increments of 0.5. The evaluation assumes $\gamma = 20$ dB, $\gamma_{spare} = 50$ dB, $N_0 = -90$ dBm, and $\alpha = 4$, and employs the greedy approach to solve the SCP.

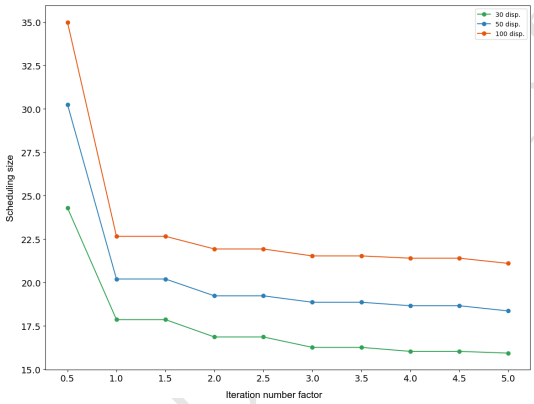

Figure 3: Average size of the scheduling obtained with the Sk-Iterative algorithm for different numbers of iterations (λ).

The results are shown in Figure 3. A strong impact is observed when increasing λ from 0.5 to 1.0 times the number of links to be scheduled. For larger values, the curve quickly flattens, and from $\lambda = 3.0$ times the number of links onward it becomes nearly constant, with a negligible slope. Therefore, the remaining experiments consistently adopt $\lambda = 3$ times the number of links to be scheduled throughout this work.

Next, results are presented comparing Sk-Iterative against the optimal algorithm and a strategy based on Sk-Greedy. Since computing an optimal schedule for large networks is computationally expensive, it is only feasible for smaller and less dense instances. For this reason, the comparison experiments were conducted separately (one set comparing against the optimal algorithm and another set comparing against the Sk-Greedy version). In both cases, the DTE heuristic was applied to obtain the set of links to be scheduled, using the same parameters adopted previously: $\gamma = 20$ dB, $\gamma_{spare} = 50$ dB, and $\alpha = 4$. Networks were generated randomly over a planar area of 100×100 m^2.

In the first experiment, Sk-Iterative is compared against the optimal algorithm considering a network with 8 devices, which yields, on average, 14 DTE links. Results for 30 randomly generated instances are shown in Fig. 4. The results indicate that Sk-Iterative achieves good performance, producing schedules that closely approximate the optimal solution. On average, the difference between the schedule sizes obtained by both approaches is only one time interval.

Next, Sk-Iterative was compared against the Sk-Greedy version. This experiment enables the evaluation of denser networks. Fig. 5 shows results for randomly generated networks with 50, 75, and 100 devices. For each network size, 30 independent instances were generated. The application of the DTE heuristic produced link sets of sizes 98, 148, and 198 for scheduling, respectively. The results in Fig. 5 indicate that, in most cases, the schedules produced

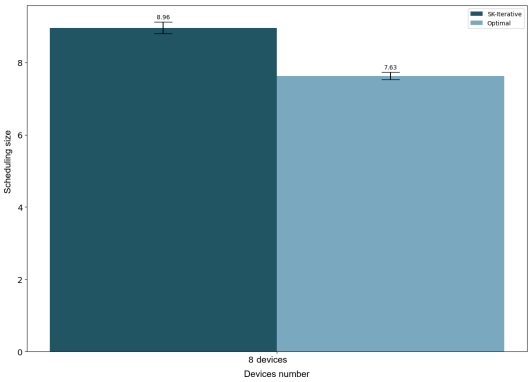

Figure 4: Average size of the schedules produced by the Sk-Iterative and optimal algorithms.

by the Sk-Greedy version are approximately 40% larger than those obtained with Sk-Iterative.

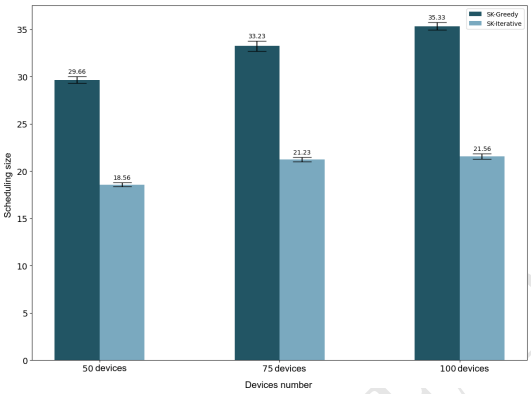

Figure 5: Average size of the schedules obtained by the Sk-Iterative and SK-Greedy algorithms.

6 Conclusion

The Sk-Iterative algorithm is a heuristic strategy proposed to address the scheduling problem in wireless links with spatial reuse under the SINR model. It leverages a set of links generated by the DTE heuristic to create a strongly connected topology. The algorithm employs a greedy, iterative approach to generate candidate assignments, followed by an application of the Set Covering Problem (SCP) algorithm. Simulation results indicate that the schedules produced are very close to those generated by the optimal algorithm. Future research will explore the use of the Sk-Iterative algorithm to support efficient routing strategies and distributed algorithms in wireless networks under the SINR model.

References

[1] Matthew Andrews and Michael Dinitz. 2009. Maximizing capacity in arbitrary wireless networks in the SINR model: Complexity and game theory. In *IEEE INFOCOM 2009*. IEEE, 1332–1340.

[2] Zhijun Cai, Mi Lu, and C.N. Georghiades. 2003. Topology-transparent time division multiple access broadcast scheduling in multihop packet radio networks. *IEEE Transactions on Vehicular Technology* 52, 4 (July 2003), 970–984.

[3] Fabio Engel De Camargo and Elias P. Duarte. 2021. A Down-to-Earth Scheduling Strategy for Dense SINR Wireless Networks. In *10th Latin-American Symp. Dep. Computing (LADC)*.

[4] Chaima Chabira, Ibraheem Shayea, Gulsaya Nurzhaubayeva, Laura Aldasheva, Didar Yedilkhan, and Saule Amanzholova. 2025. AI-Driven Handover Management and Load Balancing Optimization in Ultra-Dense 5G/6G Cellular Networks. *Technologies* 13, 7 (2025), 276.

[5] Elias Procopio Duarte, Andrea Weber, and Keiko VO Fonseca. 2011. Distributed diagnosis of dynamic events in partitionable arbitrary topology networks. *IEEE Transactions on Parallel and Distributed Systems* 23, 8 (2011), 1415–1426.

[6] Elias P Duarte Jr, Luiz A Rodrigues, Edson T Camargo, and Rogerio Turchetti. 2022. A distributed system-level diagnosis model for the implementation of unreliable failure detectors. *arXiv preprint arXiv:2210.02847* (2022).

[7] Elias Procopio Duarte Jr and Andréa Weber. 2003. A distributed network connectivity algorithm. In *The Sixth International Symposium on Autonomous Decentralized Systems, 2003. ISADS 2003.* IEEE, 285–292.

[8] Siti N Fatihah, Gilang RR Dewa, Jindae Kim, and Illsoo Sohn. 2025. Discovery Latency Analysis of Ultra-Dense Internet-of-Things Networks. *IEEE Access* (2025).

[9] Vinicius Fulber-Garcia, Fabio Engel de Camargo, and Elias P. Duarte. 2022. Sk-Greedy: A Heuristic Scheduling Algorithm for Wireless Networks under the SINR Model. In *11th Latin-American Symp. on Dependable Comp. (LADC)*. ACM.

[10] Vinicius Fulber-Garcia, Fábio Engel, and Elias P Duarte. 2024. A genetic scheduling strategy with spatial reuse for dense wireless networks. *International Journal of Hybrid Intelligent Systems* 20, 1 (2024), 41–55. doi:10.3233/HIS-230015

[11] Thiago Garrett, Schahram Dustdar, Luis CE Bona, and Elias P Duarte. 2018. Traffic differentiation on internet of things. In *2018 IEEE Symposium on Service-Oriented System Engineering (SOSE)*. IEEE, 142–151.

[12] Olga Goussevskaia. 2009. Efficiency of Wireless Networks: Approximation Algorithms for the Physical Interference Model. *Foundations and Trends in Networking* 4, 3 (December 2009), 313–420.

[13] Olga Goussevskaia, Yvonne Anne Oswald, and Roger Wattenhofer. 2007. Complexity in geometric SINR. In *MobiHoc '07*. ACM Press.

[14] Tal Grossman and Avishai Wool. 1997. Computational experience with approximation algorithms for the set covering problem. *European Journal of Operational Research* 101, 1 (August 1997), 81–92.

[15] Pryscila Barvik Guttoski, Marcos Sfair Sunye, and Fabiano Silva. 2007. Kruskal's algorithm for query tree optimization. In *11th international database engineering and applications symposium (IDEAS 2007)*. IEEE, 296–302.

[16] Magnus M. Halldorsson and Tigran Tonoyan. 2019. Plain SINR is Enough!. In *ACM Symposium on Principles of Distributed Computing*. 127–136.

[17] Egon Hilgenstieler, Elias P Duarte, Glenn Mansfield-Keeni, and Norio Shiratori. 2007. Improving the precision and efficiency of log-based IP packet traceback. In *IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference*. IEEE, 1823–1827.

[18] Egon Hilgenstieler, Elias P Duarte Jr, Glenn Mansfield-Keeni, and Norio Shiratori. 2010. Extensions to the source path isolation engine for precise and efficient log-based IP traceback. *Computers & Security* 29, 4 (2010), 383–392.

[19] Huawei. 2024. Receiver Sensitivity. https://support.huawei.com/enterprise/br/doc/EDOC1000077015/bc2e25db/receiver-sensitivity. Acessado em 20/03/2024.

[20] Tomasz Jurdzinski and Dariusz R Kowalski. 2012. Distributed backbone structure for algorithms in the SINR model of wireless networks. In *International Symposium on Distributed Computing*. Springer, 106–120.

[21] Mahmoud Kamel, Walaa Hamouda, and Amr Youssef. 2016. Ultra-dense networks: A survey. *IEEE Communications surveys & tutorials* 18, 4 (2016), 2522–2545.

[22] Neeraj Kumar, Al-Sakib Khan Pathan, Elias P Duarte Jr, and Riaz Ahmed Shaikh. 2015. Critical applications in vehicular ad hoc/sensor networks. *Telecommunication Systems* 58, 4 (2015), 275–277.

[23] P Loganathan et al. 2025. Machine Learning-Driven Hybrid Adaptive Beamforming for Efficient Interference Mitigation in Dense Wireless Sensor Networks. In *ICVADV*. IEEE, 984–990.

[24] Neda Mohammadi, Bahram Sadeghi Bigham, and Mehdi Kadivar. 2025. PDSLS: An approximation SINR-based Shortest Link Scheduling algorithm with power control. *Computer Communications* 236 (2025), 108137.

[25] Bogdan T Nassu, Takashi Nanya, et al. 2007. Topology discovery in dynamic and decentralized networks with mobile agents and swarm intelligence. In *ISDA*. 685–690.

[26] Theodore S. Rappaport. 2002. *Wireless communications: principles and practice*. Prent. Hall.

[27] Aggeliki Sgora, Dimitrios J. Vergados, and Dimitrios D. Vergados. 2015. A Survey of TDMA Scheduling Schemes in Wireless Multihop Networks. *Comput. Surveys* 47, 3 (April 2015), 1–39.

[28] Yinglei Teng et al. 2019. Resource Allocation for Ultra-Dense Networks: A Survey, Some Research Issues and Challenges. *IEEE Comm. Surveys & Tutorials* 21, 3 (2019), 2134–2168.

Bridging Blockchains: A Comprehensive Analysis of Interoperability Challenges and Multi-Blockchain Architectures

Pedro M. R. G. Silva
Matheus Lázaro Honório da Silva
Gislainy Velasco
Luciana Berreta
Sergio T. Carvalho
Eduardo S. de Albuquerque
mrgpedrosilva@gmail.com
{matheus.lazaro,gislainycrisostomo}@discente.ufg.br
{Sergio,LucianaBerretta,esalbuquerque}@ufg.br
Instituto de Informática, Universidade Federal de Goiás, Goiânia, Goiás, Brazil

Paulo R. F. Cunha
Centro de Informática - Universidade Federal de Pernambuco, Recife, Pernambuco
Brazil
prfc@cin.ufpe.br

ABSTRACT

The rapid evolution of blockchain technologies has intensified the need for interoperable and multi-chain infrastructures, particularly in domains that rely on secure and auditable data management, such as healthcare. This paper presents a structured literature review on blockchain interoperability and multi-blockchain architectures, with emphasis on their applicability to healthcare data-sharing systems. Through a systematic examination of existing models—including Polkadot, Hyperledger Cacti, and Cosmos—we organize and analyze the main approaches proposed in the field. The study offers three primary contributions: (i) the systematization of the key technical and organizational challenges involved in enabling interoperability across heterogeneous blockchains; (ii) the mapping and classification of current solutions and architectural strategies reported in the literature; and (iii) the identification of multi-blockchain consensus as a promising and underexplored research direction for reducing external trust assumptions in cross-chain operations. The findings aim to support researchers and practitioners in understanding current limitations and guiding future investigations on interoperable blockchain ecosystems.

KEYWORDS

Blockchain Interoperability, multi-blockchain architectures

1 INTRODUCTION

Distributed Ledger Technologies (DLTs), spearheaded by Bitcoin, have revolutionized security, data integrity, and transparency across various industries, from IoT [3, 32] to smart cities [22]. Blockchain's inherent immutability and transparency address traditional cybersecurity vulnerabilities. Its potential is particularly significant in healthcare for secure data sharing and electronic health records [14, 29, 34], and in AI for optimizing machine learning models [29, 36].

The evolution of DLTs beyond initial use cases has driven a critical shift towards seamless communication and interoperability between disparate blockchain networks [6, 29]. This led to "multi-blockchain networks" [23, 34], as the ecosystem moved from monolithic to fragmented, specialized solutions, particularly evident in healthcare studies [22, 31, 34]. The core challenge is enabling effective data exchange across these isolated systems while preserving blockchain's intrinsic security and trust [5, 14, 34]. Interoperability is now a fundamental requirement for robust solutions in critical domains [23].

This paper provides a holistic overview of multi-blockchain research, focusing on critical trends and challenges relevant to global scientific and industrial communities. Our work is structured as:

- **Related Work:** A review of existing literature on blockchain interoperability and multi-blockchain architectures.
- **Fundamentals:** Conceptual foundation of blockchain, consensus mechanisms, and cross-chain communication principles.
- **Multi-Blockchain Architectures:** Analysis of prevalent cross-chain communication approaches, including Relay models, Polkadot, Hyperledger Cacti, and Cosmos.
- **Trends and Challenges:** Identified challenges such as communication across heterogeneous systems, shared consensus mechanisms, regulatory compliance, and decentralized identity verification.
- **Conclusion:** Summarizes key findings and outlines future research directions.

Our comprehensive analysis aims to inform future international research, fostering integrated, secure, and scalable DLT solutions worldwide. This has global relevance for bridging technological divides and promoting responsible blockchain adoption.

2 RELATED WORK

Recent research intensifies efforts for seamless DLT interoperability, driven by rapid technological evolution and diversification. Seminal reviews by [6] and [17] detail blockchain interoperability capabilities, techniques (HTLCs, Relay/Sidechains), and foundational mechanisms. A comprehensive survey by [37] further addresses architectures, solutions, and challenges.

Recent research explores advanced architectures. [23] proposed a Hierarchical Multi-blockchain design using proxy blockchains and Inter-blockchain APIs. [34] introduced a cross-chain communication system with relay and proxy nodes for credential forwarding.

[16] envisions global smart contracts for asset exchange, while [15] surveys cross-chain security, addressing vulnerabilities.

Healthcare is a fertile ground for multi-blockchain applications due to the need for secure, privacy-preserving, and interoperable data-sharing. Private blockchains are explored for sensitive health information management [1, 9, 20, 25, 30, 40]. [23] emphasizes interoperability as a basic prerequisite for robust solutions in healthcare and other critical ecosystems. [2] proposed a secure multi-layered architecture for medical data sharing and access control.

Multi-blockchain architecture remains evolving, particularly in advanced interoperability and security [5, 6, 14, 23, 29, 34]. This paper builds on this by providing a holistic view of current research, identifying key trends and challenges, especially for advanced security and privacy.

3 FUNDAMENTALS

This section provides fundamental concepts for understanding blockchain technology, its consensus mechanisms, and cross-chain communication.

3.1 Blockchain Core Concepts

Blockchain and other Distributed Ledger Technologies (DLTs) fundamentally represent decentralized, distributed ledgers where content must be synchronized and identical across all participating nodes. This ledger is structured as a chain of cryptographically linked blocks, forming an immutable and auditable record. Each block contains a set of validated transactions, and the cryptographic link between blocks is established by embedding the hash of the previous block within the current block's header. This also includes other metadata such as a nonce and the Merkle tree root of its transactions, which effectively summarizes all transactions in the block into a single hash, enabling efficient verification of inclusion [24]. The immutability stems from the fact that altering any transaction would change its hash, propagating up the Merkle tree and altering the block hash, thereby breaking the chain's cryptographic link.

Blockchains can be broadly classified based on two primary aspects: node participation and consensus participation [37].

Node Participation: This determines who can join and operate a node within the network.

- **Public (Permissionless) Blockchains:** Networks like Bitcoin and Ethereum allow anyone to join, validate transactions, and operate nodes without prior authorization. They are fully decentralized, often pseudonymous, and typically rely on economic incentives (e.g., block rewards) for security. Their open nature fosters broad participation but can lead to lower transaction throughput.
- **Private (Permissioned) Blockchains:** In these networks, such as Hyperledger Fabric, participation is restricted. Nodes must be explicitly authorized by network administrators to join, operate, and access the chain state [17, 39]. This often results in higher performance, enhanced privacy (known identities), and more efficient governance suitable for enterprise use cases, albeit with a trade-off in decentralization.

Consensus Participation: This refers to who can participate in the process of validating and adding new blocks.

- **Private Blockchains:** Consensus is solely managed by a single organization or authority that controls all participating nodes.
- **Consortium Blockchains:** Consensus is managed by a pre-selected group of trusted nodes, often representing multiple distinct organizations forming a consortium [39]. This model provides a balance between decentralization and control, making it popular for inter-organizational applications.

These classifications highlight the spectrum from fully decentralized to centrally managed DLTs, each with distinct trust models, performance characteristics, and application suitability.

3.2 Consensus Mechanisms

In any decentralized system, ensuring that all participating nodes agree on a single, consistent state of the ledger is paramount. This is achieved through consensus mechanisms, which are protocols designed to regulate changes to the blockchain state and validate transactions. The choice of consensus mechanism significantly impacts a blockchain's security, scalability, and decentralization properties. A comprehensive overview of these mechanisms is available in [21]. Key mechanisms include:

Proof of Work (PoW): Introduced by [24] with Bitcoin, PoW requires participants (miners) to solve a computationally intensive cryptographic puzzle. The first node to find a valid solution (a nonce that, when combined with block data, produces a hash below a certain target) earns the right to add the next block to the blockchain, receiving a reward. While highly secure against Sybil attacks and expensive for malicious actors to achieve a 51% attack, PoW is known for its substantial energy consumption and limited transaction throughput [19, 29]. The difficulty of the puzzle dynamically adjusts to maintain a consistent block time.

Proof of Stake (PoS): As an energy-efficient alternative to PoW, PoS selects validators based on the amount of cryptocurrency they have "staked" (locked up) as collateral. Nodes with larger stakes have a higher probability of being chosen to create new blocks and validate transactions. This mechanism significantly reduces energy costs, often offers higher transaction throughput, and introduces economic disincentives (slashing) for malicious behavior, as validators risk losing their staked assets [19, 29]. Variants include Delegated PoS (DPoS) and Bonded PoS.

Proof of Authority (PoA): Primarily used in private and consortium blockchains, PoA relies on a limited number of pre-selected and explicitly trusted nodes (authorities) to validate transactions and create new blocks. These authorities are typically known, reputable entities, chosen for their trustworthiness. This mechanism makes the consensus process significantly faster and more efficient due to fewer participants, offering high throughput and immediate finality, albeit at the cost of some decentralization [19, 29]. Other Byzantine Fault Tolerant (BFT) protocols, such as Practical

Byzantine Fault Tolerance (PBFT) or Tendermint, are prevalent in permissioned environments, offering deterministic finality and robust performance.

3.3 Cross-Chain Communication Principles

Cross-chain communication refers to the process by which an operation initiated on one blockchain concludes or impacts another blockchain [6, 8]. This capability is fundamental for realizing the vision of interconnected multi-blockchain networks, moving beyond isolated ledger islands. [3] categorizes cross-chain interoperability into three main modes, each with distinct technical underpinnings:

- **Data Transfer:** This involves one blockchain securely copying or sending data to another, allowing for information sharing without direct asset movement. Technically, this can be achieved via light clients that verify cryptographic proofs (e.g., Merkle proofs) of states or events on the source chain without downloading the entire chain history. Oracles can also facilitate external data transfer by relaying information off-chain.
- **Asset Transfer:** Assets are moved from a source blockchain to a destination blockchain. This typically involves a "lock and mint" or "burn and mint" mechanism. The original asset is "locked" or "burned" on the source chain, and an equivalent "wrapped" or "pegged" representation is minted on the destination chain. Cryptographic proof or multi-signature schemes often secure this locking/burning process on the source chain.
- **Asset Exchange:** This facilitates atomic swaps or exchanges of assets between participants residing on different blockchains, guaranteeing that either both transactions complete or neither does. A common method is the use of Hashed Time-Lock Contracts (HTLCs), which employ cryptographic hash functions and time-based conditions to ensure secure, trust-minimized exchanges without a central intermediary.

A critical challenge in cross-chain communication is the inherent "trust assumption" [7, 38]. A fundamental principle is that one blockchain cannot natively validate a transaction occurring on another blockchain without simulating the other chain's consensus mechanism or processing its entire state, which is often computationally infeasible or time-prohibitive. Consequently, most current cross-chain solutions must assume trust in the consensus algorithm of the participating chains or rely on external entities (e.g., relayers, oracles, trusted third parties) to attest to the validity of cross-chain events. This reliance on external trust rather than robust cryptographic consensus presents a significant hurdle for achieving truly decentralized and secure interoperability.

4 MULTI-BLOCKCHAIN ARCHITECTURES

Multi-blockchain architectures represent systems comprising two or more independent blockchain networks, designed to overcome the inherent isolation of standalone blockchains and facilitate broader collaboration and data exchange. The proliferation of diverse blockchain platforms, each optimized for specific use cases (e.g., privacy, throughput, smart contract capabilities), necessitates robust interoperability solutions. In recent years, a variety of approaches have emerged to enable cross-chain communication, addressing the growing demand for interconnected decentralized ecosystems [6, 37]. This section provides an overview of key multi-blockchain architectural patterns and prominent platforms and their technical specifics for enabling cross-chain interaction modes, which include **Data Transfer**, **Asset Transfer**, and **Asset Exchange**, as defined in Section 3.3.

4.1 Relay-Based Architectures and Sidechains

Relay-based architectures are a foundational approach to cross-chain communication. Initially, this concept emerged with the division between a "mainchain" and various "sidechains." In this model, the mainchain would maintain an asset ledger capable of recognizing and validating state changes on attached sidechains [17]. Over time, the concept of a relay evolved to describe a more generic mechanism where one blockchain can initiate or trigger transactions, smart contracts, or chaincodes on another blockchain [5, 33]. Relay chains act as intermediaries, monitoring and validating events on connected chains (often called parachains, zones, or shards) to enable secure cross-chain communication. They achieve this by processing and verifying proofs of state transitions or transaction finality from the connected chains. These architectures often employ "bridges"—specialized protocols or components—to facilitate the transfer of data or assets between chains, including those with different consensus mechanisms or virtual machines [27]. Bridges can be custodial (relying on a trusted third party for validation) or non-custodial (using smart contracts or light clients for trustless verification). The design of these bridges is crucial for the security and decentralization level of cross-chain operations.

4.2 Polkadot

Polkadot, an initiative co-founded by Dr. Gavin Wood, one of Ethereum's co-founders, is a prominent example of a Layer 0 blockchain designed explicitly for blockchain interoperability [11, 33]. It functions as a relay-based infrastructure where a central "Relay Chain" connects numerous independent "Parachains." Parachains are application-specific blockchains, often built using the Substrate framework, which provides a Runtime Module Library (RML) for rapid development of custom blockchain runtimes. These parachains can have their own governance, consensus mechanisms, and data structures. Polkadot enables these parachains to communicate securely and trustlessly via the Relay Chain through its Cross-Chain Message Passing (XCMP) protocol. XCMP allows for the direct transfer of messages (including asset transfers and data sharing) between parachains without going through the Relay Chain itself for every message, only for notarization and shared security. Crucially, all parachains connected to the Relay Chain benefit from Polkadot's "shared security" model, meaning they inherit the Relay Chain's security guarantees and finalize blocks together, rather than needing to establish their own security infrastructure. This shared security paradigm significantly simplifies the security burden for individual parachains. While Polkadot fosters a heterogeneous environment, allowing diverse parachains to connect, its direct interoperability is primarily confined to chains within its own infrastructure [28].

4.3 Hyperledger Cacti

Hyperledger Cacti is a project within the Hyperledger Foundation, aimed at enabling interoperability specifically among Hyperledger blockchains, such as Hyperledger Besu (an Ethereum client) and Hyperledger Fabric (a permissioned blockchain platform). Described as a "pluggable interoperability framework," Cacti is the result of a merger between Hyperledger Cactus and Hyperledger Weaver. Its modular design allows it to support various ledger types, authentication mechanisms, and smart contract languages through a flexible plugin architecture. It facilitates cross-chain operations between ledgers through either Relay or Node server flows, both executed via a "connector." This blockchain-specific connector layer incorporates contracts and validators, possessing the necessary permissions to execute chaincodes and smart contracts on their respective blockchains. The interoperability process typically involves a request from a source ledger, which is then translated and relayed by the connector to the target ledger for execution. This flexible design allows for various trust models, from tightly coupled enterprise ledgers to more decentralized configurations. Cacti's emphasis on flexibility and extensibility makes it particularly suitable for complex enterprise blockchain environments where diverse, often permissioned, ledgers need to interact securely [5].

4.4 Cosmos

Cosmos positions itself as an "Internet of Blockchains," similar to Polkadot in its ambition to connect disparate chains via a relay-based infrastructure [12]. It organizes independent chains into "Zones," which are sovereign blockchains, often running on the Tendermint BFT (Byzantine Fault Tolerant) consensus algorithm. Tendermint Core provides instant block finality and high throughput, making Zones highly efficient. These Zones connect to "Hubs," which act as relay chains. Hubs maintain records of assets and state changes exchanged between connected Zones. The Inter-Blockchain Communication (IBC) protocol [11, 12] is central to Cosmos, enabling secure and reliable token and data exchange between Hubs and Zones. IBC works by allowing blockchains to send data packets to each other, using light client verification to prove the consensus state of the counterparty chain, without needing to trust an intermediary. This ensures that message passing is secure and trust-minimized, enabling permissionless transfer of tokens (asset transfer) and arbitrary data packets (data transfer) between any IBC-enabled chains. Cosmos's design allows Zones to maintain their sovereignty while still participating in a larger interconnected ecosystem. Cosmos aims to provide a more open interoperability framework, with ambitions to connect with external blockchains like Ethereum and Bitcoin, distinguishing it from more closed ecosystems like Polkadot [26]. The modularity of Cosmos's SDK simplifies the creation of custom Zones compatible with IBC.

5 TRENDS AND CHALLENGES

The blockchain landscape has profoundly transformed from singular public blockchains to fragmented, customizable, and private deployments. This shift is evident in sensitive domains like healthcare [22, 31, 34], supply chain management [18], and financial services [13], where private or permissioned blockchains are increasingly leveraged for data-sharing and collaborative operations. The imperative is to enable seamless data sharing and interaction among isolated blockchain instances, preserving their intrinsic security, immutability, and decentralization [5, 6, 14, 23, 29, 34]. This evolving context presents critical challenges and emerging trends for detailed examination.

5.1 Heterogeneous Blockchain Interoperability and Protocol Standardization

A paramount challenge is robust, trustless communication among heterogeneous multi-blockchain systems. While platform-specific solutions like Polkadot and Hyperledger Cacti advance communication within their ecosystems, a universally applicable inter-blockchain communication protocol remains an active research area [5, 6]. Cosmos's IBC protocol attempts generalization, but faces limitations, particularly for sensitive data where Hubs storing asset data pose privacy concerns [7, 22, 23, 25]. Lack of academic consensus on a definitive multi-blockchain architecture, coupled with diverse consensus mechanisms, data models, and smart contract environments, hinders seamless interoperability [27, 37].

5.2 Shared Consensus and Mitigating Trust Assumptions

Current interoperability solutions frequently rely on an inherent "trust assumption" regarding external systems [7, 38]. This implies one blockchain must trust the validity of transactions reported by another, as efficient native validation is prohibitive. A significant challenge persists in cryptographically validating multi-blockchain transactions without external validators. A compelling, largely unexplored solution lies in *multi-blockchain consensus protocols*. Such protocols would involve collaborative validation of cross-chain transactions by nodes from multiple participating blockchains, substantially reducing external trust dependencies and enhancing overall security [15]. This paradigm shift from assumed trust to cryptographically enforced shared validation represents a critical frontier in interoperability research.

5.3 Sensitive Data Sharing and Regulatory Compliance

Blockchain offers immense potential for data-sharing in healthcare due to transaction immutability and enhanced accountability. There is a strong drive to establish governance frameworks empowering data owners to control access and audit usage [20, 40]. However, blockchain's intrinsic data handling, like permanent ledger storage, challenges adherence to stringent data protection regulations such as GDPR (Europe) and LGPD (Brazil) [31]. Recent research focuses on integrating privacy-enhancing technologies, such as Zero-Knowledge Proof (ZKP) algorithms, to modify data access and enable compliant consensus processes [4, 10]. Further advancements in secure multi-layered architectures for medical data sharing are also explored [2].

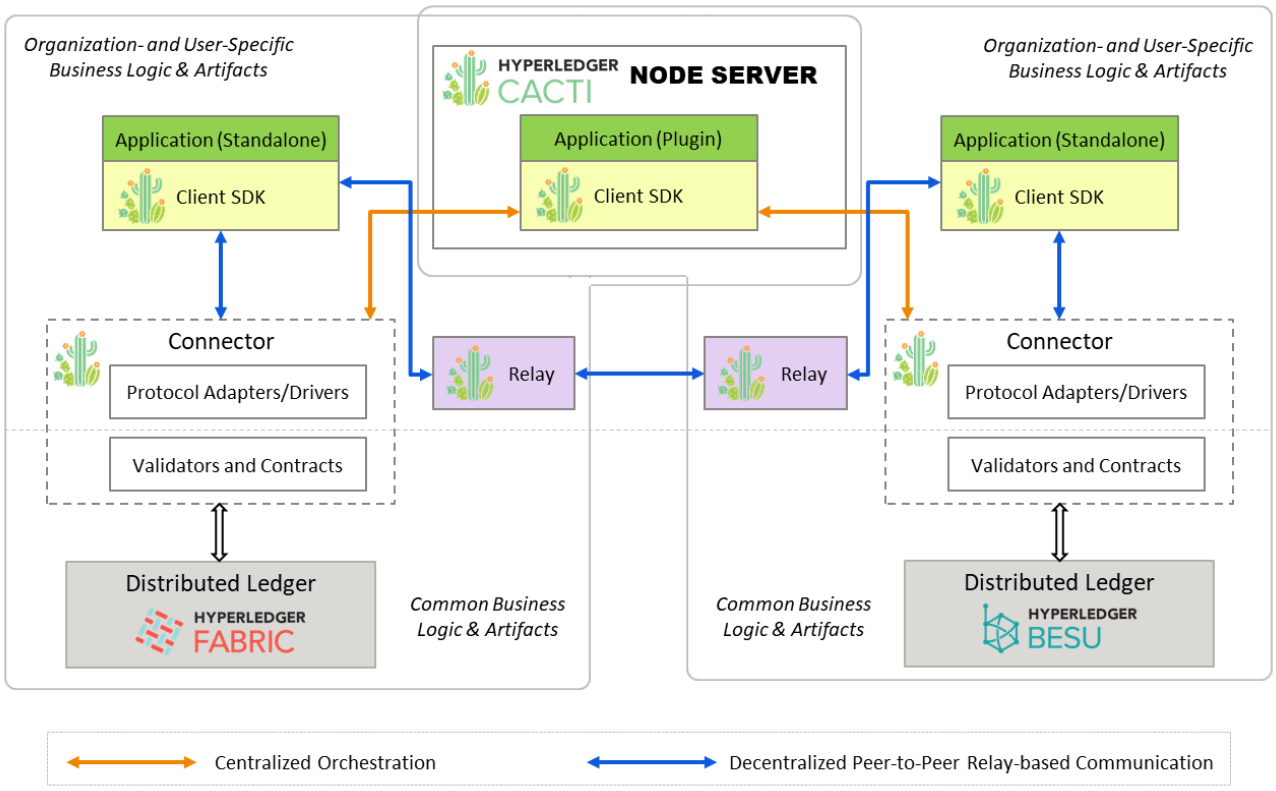

Figure 1: Hyperledger Cacti overview

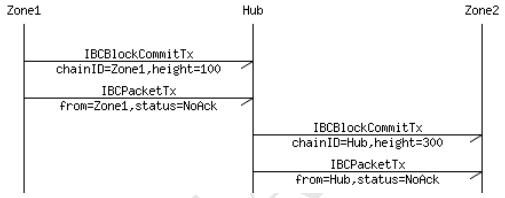

Figure 2: Cosmos Inter-Blockchain protocol

5.4 Decentralized Identity Verification

Reliable identity verification remains a persistent challenge in blockchain systems, especially in healthcare and regulated industries. Traditional blockchain signatures, relying on asymmetric key pairs, do not intrinsically derive verifiable real-world identities from public keys. This necessitates robust mechanisms to link digital identities to real-world entities in a decentralized, privacy-preserving manner. Recent studies focus on Decentralized Identifiers (DIDs) and Self-Sovereign Identity (SSI) models to securely validate identities within blockchain ecosystems [4, 10, 35]. These approaches aim to give individuals greater control over their digital personas and personal data, aligning with decentralization and privacy principles critical for widespread blockchain adoption.

Vou criar uma seção explícita de "Contributions" para a sua apresentação, focando em destacar os pontos importantes.

Claro — segue uma seção explícita “Contributions” em português acadêmico e em formato LaTeX, totalmente alinhada com o conteúdo real do artigo main.tex.txt (que é uma revisão de literatura).

A seção está precisa, formal e compatível com o escopo de um artigo de revisão.

6 CONTRIBUTIONS

This article is structured as a literature review that synthesizes and analyzes recent advances in blockchain interoperability and multi-blockchain architectures, with particular attention to applications in healthcare. The contributions of this review are threefold:

(1) **Systematization of challenges**: We consolidate and organize the main technical, governance-related, and regulatory challenges identified in the literature, including heterogeneous blockchain communication, external trust assumptions, sensitive data handling, and identity verification issues in healthcare-oriented blockchain systems.
(2) **Mapping of existing approaches**: We survey and classify prominent interoperability models and multi-blockchain architectures—such as Relay-based mechanisms, Polkadot, Hyperledger Cacti, and Cosmos—highlighting how each

solution conceptualizes and implements cross-chain communication.

(3) **Identification of future research directions**: We identify multi-blockchain consensus as a promising and underexplored research avenue, emphasizing its potential to reduce cross-chain trust assumptions and strengthen the validation of multi-chain transactions.

These contributions provide a consolidated foundation for researchers and practitioners seeking to understand the current landscape of interoperable blockchain ecosystems and to guide future investigations in the field.

7 LIMITATIONS

While this article provides a structured review of blockchain interoperability and multi-blockchain architectures, several limitations must be acknowledged. First, the study is based exclusively on published literature, which may introduce publication bias and exclude industry implementations or proprietary solutions that are not publicly documented. As a result, some emerging interoperability mechanisms—particularly those used in commercial or consortium-based healthcare systems—may not be captured in this review.

Second, the rapidly evolving nature of blockchain technologies poses an inherent limitation. New interoperability protocols, cross-chain frameworks, and multi-blockchain consensus mechanisms are being proposed continuously, which means that the taxonomy and classifications presented here reflect the state of the field at the time of writing but may not encompass the most recent advancements.

Third, the review focuses primarily on conceptual and architectural aspects described in the literature, without performing empirical validation or comparative benchmarking across solutions. This restricts the ability to assess practical performance, scalability, operational constraints, or real-world security trade-offs associated with each interoperability model.

Finally, although the healthcare sector motivates much of the discussion, the analysis does not perform domain-specific evaluations of regulatory compliance, clinical workflow integration, or healthcare-specific threat models. Future studies should investigate these dimensions empirically and consider how interoperability frameworks operate within real healthcare data ecosystems.

8 CONCLUSION

This article presented a structured literature review of blockchain interoperability models and multi-blockchain architectures, with a particular emphasis on their applicability to healthcare data-sharing systems. Through this review, we (i) systematized the main technical and organizational challenges associated with heterogeneous blockchain communication, identity management, and sensitive data governance; (ii) mapped and classified relevant interoperability approaches—such as Relay-based mechanisms, Polkadot, Hyperledger Cacti, and Cosmos—highlighting how each architecture conceptualizes and operationalizes cross-chain interactions; and (iii) identified multi-blockchain consensus as a promising and underexplored direction for future research, with the potential to reduce trust assumptions in cross-chain transactions.

However, the findings of this review must be interpreted in light of its limitations. The study relies exclusively on published literature, which may exclude industrial implementations and proprietary interoperability mechanisms not publicly documented. Additionally, given the rapid pace of development in blockchain technologies, the landscape of interoperability solutions evolves continuously, potentially introducing new approaches beyond the scope of this review. Moreover, the analysis is conceptual in nature and does not include empirical evaluation or benchmarking of the surveyed architectures. Finally, although the healthcare sector motivates the discussion, domain-specific regulatory, operational, and clinical constraints were not evaluated in depth.

Future work should address these limitations by conducting empirical assessments of interoperability mechanisms, examining real-world multi-blockchain deployments in healthcare ecosystems, and exploring the practical feasibility of shared multi-blockchain consensus protocols in reducing external trust assumptions.

ACKNOWLEDGMENTS

The research for this paper was financially supported by CAPES (Coordination for the Improvement of Higher Education Personnel).

REFERENCES

[1] O. Ajayi, M. Abouali, and T. Saadawi. 2020. Secured inter-healthcare patient health records exchange architecture. In *2020 IEEE International Conference on Blockchain (Blockchain)*. 456–461.

[2] Mamoun Alazab, Ahmad I. Al-Jarrah, Mazin Abutaha, and Rupak Kharel. 2023. A secure blockchain-based multi-layered architecture for medical data sharing and access control. *Future Generation Computer Systems* 148 (2023), 115–126.

[3] A. Alghuried, M. Alkinoon, M. Mohaisen, A. Wang, C. C. Zou, and D. Mohaisen. 2025. Blockchain security and privacy: Threats, challenges, applications, and tools. *Distrib. Ledger Technol. Just Accepted* (2025).

[4] T. Bai, Y. Hu, J. He, H. Fan, and Z. An. 2022. Health-zkidm: A healthcare identity system based on fabric blockchain and zero-knowledge proof. *Sensors* 22, 20 (2022).

[5] R. Belchior, H. Borne-Pons, J. Hamilton, M. Bowman, P. Somogyvari, H. Montgomery, S. Fujimoto, T. Takeuchi, and T. Kuhrt. 2022. Cacti/whitepaper/whitepaper.md at 7bb39576080592919bea0ac89646b32105e1748e ·hyperledger-cacti/cacti. In *Technical Report*.

[6] R. Belchior, L. Riley, T. Hardjono, A. Vasconcelos, and M. Correia. 2023. Do you need a distributed ledger technology interoperability solution? *Distrib. Ledger Technol.* 2, 1 (2023).

[7] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia. 2021. A survey on blockchain interoperability: Past, present, and future trends. *ACM Comput. Surv.* 54, 8 (2021).

[8] B. Bellaj, A. Ouaddah, E. Bertin, N. Crespi, and A. Mezrioui. 2022. Sok: A comprehensive survey on distributed ledger technologies. In *2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 1–16.

[9] S. Chakraborty, S. Aich, and H.-C. Kim. 2019. A secure healthcare system design framework using blockchain technology. In *2019 21st International Conference on Advanced Communication Technology (ICACT)*. 260–264.

[10] Matheus H. da Silva, Gislainy Velasco, N. P. Vaz, M. Martins, P. R. G. Silva, and Sergio Carvalho. 2025. Blockchain and Self-Sovereign Identity: A Healthcare Use Case. In *Anais do VIII Workshop em Blockchain: Teoria, Tecnologias e Aplicações*. SBC, Porto Alegre, RS, Brasil, 154–167.

[11] A. M. Eldin, E. Hossny, K. Wassif, and F. A. Omara. 2022. Survey of blockchain methodologies in the healthcare industry. *2022 5th International Conference on Computing and Informatics (ICCI)* (2022), 209–215.

[12] J. K. Ethan Buchman. 2016. *Cosmos*. Technical Report.

[13] W. Fan, Y. Zhao, and S. Cui. 2021. Blockchain in finance: A systematic literature review. *Journal of Industrial Information Integration* 21 (2021), 100189.

[14] A. N. Gohar, S. A. Abdelmawgoud, and M. S. Farhan. 2022. A patient-centric healthcare framework reference architecture for better semantic interoperability based on blockchain, cloud, and iot. *IEEE Access* 10 (2022), 92137–92157.

[15] Qiang Guo, Mingxu Fu, Yaoxue Zhang, and Fuxiang Li. 2023. A Survey on Cross-Chain Security. *IEEE Transactions on Network Science and Engineering* 10, 1 (2023), 329–346.

[16] F. Hashim, K. Shuaib, E. Baraka, and F. Sallabi. 2024. Enhancing EHR sharing through interconnected blockchains via global smart contracts. *International Journal of Computing and Digital Systems* 16, 1 (2024), 1579–1591.
[17] T. Haugum, B. Hoff, M. Alsadi, and J. Li. 2022. Security and privacy challenges in blockchain interoperability - a multivocal literature review. In *Proceedings of the 26th International Conference on Evaluation and Assessment in Software Engineering, EASE '22*. Association for Computing Machinery, 347–356.
[18] N. Kshetri and P. K. Singh. 2021. Blockchain and supply chain management: A systematic review. *Computers Industrial Engineering* 153 (2021), 107127.
[19] B. Lashkari and P. Musilek. 2021. A comprehensive review of blockchain consensus mechanisms. *IEEE Access* 9 (2021), 43620–43652.
[20] A. R. Lee, M. G. Kim, K. J. Won, I. K. Kim, and E. Lee. 2020. Coded dynamic consent framework using blockchain for healthcare information exchange. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. 1047–1050.
[21] Jin Liang, Dongxiao Wang, Yuhong Tan, Jiangjun Cao, Bin Chen, Yan Han, and Guang Wang. 2023. A Comprehensive Review of Blockchain Consensus Algorithms: State-of-the-Art, Challenges, and Future Directions. *IEEE Transactions on Network and Service Management* 20, 3 (2023), 2533–2553.
[22] J. Liu, G. Zhang, R. Sun, X. Du, and M. Guizani. 2020. A blockchain-based conditional privacy-preserving traffic data sharing in cloud. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. 1–6.
[23] V. Malamas, G. Palaiologos, P. Kotzanikolaou, M. Burmester, and D. Glynos. 2023. Janus: Hierarchical multi-blockchain-based access control (hmbac) for multi-authority and multi-domain environments. *Applied Sciences (Switzerland)* 13, 1 (2023).
[24] S. Nakamoto. 2009. *Bitcoin: A peer-to-peer electronic cash system.* Technical Report.
[25] S. Pandey, A. K. De, S. Choudhary, and M. Asim. 2023. A decentralized blockchain-based architecture for healthcare industry. In *2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI)*, Vol. 1. 1–5.
[26] Jooyoung Park, Yongsuk Lee, and Donghun Lee. 2023. Comparative Analysis of Blockchain Interoperability Solutions: Polkadot, Cosmos, and Chainlink. *Sensors* 23, 12 (2023), 5497.
[27] Partha Pramanik, Sandeep Kumar Garg, and Rajbir Kaur. 2023. A comprehensive survey on blockchain bridges: architectures, security analysis, attacks, and future directions. *Journal of Network and Computer Applications* 220 (2023), 103728.
[28] Prakash Kumar Sahoo and Rabindra Nath Mishra. 2023. Polkadot: A comprehensive survey on blockchain interoperability platform. *J. Parallel and Distrib. Comput.* 174 (2023), 51–69.
[29] R. Song, B. Xiao, Y. Song, S. Guo, and Y. Yang. 2023. A survey of blockchain-based schemes for data sharing and exchange. *IEEE Transactions on Big Data* 9, 6 (2023), 1477–1495.
[30] G. Suciu, M. Balanescu, S. Mitroi, D. Trufin, M. Falahi, C. Serban, and N. Goga. 2022. An overview of blockchain technology in stamina project. In *2022 IEEE International Conference on Blockchain, Smart Healthcare and Emerging Technologies (SmartBlock4Health)*. 1–4.
[31] H. Taherdoost. 2023. The role of blockchain in medical data sharing. *Cryptography* 7, 3 (2023).
[32] T. Wang, Q. Wang, Z. Shen, Z. Jia, and Z. Shao. 2022. Understanding characteristics and system implications of dag-based blockchain in iot environments. *IEEE Internet of Things Journal* 9, 16 (2022), 14478–14489.
[33] G. Wood. 2018. Polkadot: Vision for a heterogeneous multi-chain framework. (2018).
[34] Z. Wu, Y. Wang, and L. Wang. 2025. Gam: A scalable and efficient multi-chain data sharing scheme. *Information Processing and Management* 62, 3 (2025).
[35] J. Xu, K. Xue, S. Li, H. Tian, J. Hong, P. Hong, and N. Yu. 2019. Healthchain: A blockchain-based privacy preserving scheme for large-scale health data. *IEEE Internet of Things Journal* 6, 5 (2019), 8770–8781.
[36] Wei Yao, Wenlu Du, Jingyi Gu, Junyi Ye, Fadi P. Deek, and Guiling Wang. 2024. Establishing a Baseline for Evaluating Blockchain-Based Self-Sovereign Identity Systems: A Systematic Approach to Assess Capability, Compatibility, and Interoperability. *2024 6TH BLOCKCHAIN AND INTERNET OF THINGS CONFERENCE, BIOTC 2024* (2024), 108–119.
[37] Mohamed A. Zaki and Yousif A. Yaseen. 2023. Blockchain interoperability: A survey on architectures, solutions, and challenges. *Journal of Network and Computer Applications* 211 (2023), 103551.
[38] A. Zamyatin, M. Al-Bassam, D. Zindros, E. Kokoris-Kogias, P. Moreno-Sanchez, A. Kiayias, and W. J. Knottenbelt. 2021. Sok: Communication across distributed ledgers. (2021), 3–36.
[39] R. Zhang, R. Xue, and L. Liu. 2019. Security and privacy on blockchain. *Comput. Surveys* 52 (2019).
[40] T.-L. Zhu and T.-H. Chen. 2021. A patient-centric key management protocol for healthcare information system based on blockchain. In *2021 IEEE Conference on Dependable and Secure Computing (DSC)*. 1–5.

Received 19 January 2026

SABIÁ: A Guideline for the installation of AI Data Centers as Critical Infrastructure in Brazil

Caio Leandro Rodrigues Cavalcanti
caio.leandro.rodrigues07@aluno.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Antonio Wendell de Oliveira Rodrigues
wendell@ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Daniel de Menezes Gularte
danielgula@gmail.com
Universidade Christus
Fortaleza, CE, Brazil

Paulo Roberto Freire Cunha
prfc@cin.ufpe.br
UFPE
Recife, PE, Brazil

Antônio Mauro Barbosa de Oliveira
mauro@lar.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Abstract

This work examines the pressing issue of digital sovereignty in Brazil, emphasizing the rapid arrival of Artificial Intelligence data centers as a new form of critical infrastructure. It argues that deploying such facilities requires robust governance, sustainability, and transparency mechanisms, as well as enforceable environmental, social, and technological commitments. It further contends that negotiations with investors must be legally structured, socially fair, and guided by verifiable metrics, ensuring concrete benefits for society—especially for communities directly affected. The paper analyzes the Brazilian Artificial Intelligence Plan (PBIA) and its conceptual conflict with the Special Taxation Regime for Data Center Services (REDATA), showing that tax incentives alone do not ensure technological sovereignty or the capture of strategic value. Two main contributions are highlighted: (i) the book SABIÁ—Brazilian Sovereignty and Autonomy in Artificial Intelligence—as a guideline to accelerate PBIA implementation through an analytical framework based on verifiable metrics (with emphasis on energy, water, and enforceable governance and transparency clauses); and (ii) the proposal *AI Data Centers in Ceará: Strategies for Negotiation, Governance, and Sustainable Development*, which identifies a favorable geopolitical window and positions the state to negotiate high-impact social, technological, environmental, and economic commitments, turning the arrival of data centers into a strategic driver of development and sovereignty. As a SABIÁ case study, this proposal was accepted by the Government of Ceará and converted into a new public AI policy, designed to initiate a new development cycle in the state and strengthen both the state's negotiating capacity and digital sovereignty.

Keywords

Energy-efficient computing and networking, Critical infrastructure management, Green ICT and ICT for green, Environment friendly ICT, AI applied to infrastructures and services, Advanced techniques for networks and services monitoring

1 Introduction

The Artificial Intelligence economy has intensified the centrality of the material infrastructure that sustains computation, storage, and connectivity, shifting the AI debate from a strictly algorithmic plane to a sociotechnical regime grounded in hardware, energy, and territory. Within this arrangement, large-scale data centers emerge both as industrial policy artifacts and as sites of socio-environmental friction, due to their intensive electricity consumption, continuous water demand, land use, and impacts on transmission networks [9].

The energy dimension, therefore, cannot be treated as a mere technological externality: it is the primary material driver of scalability for AI systems and one of the key determinants of their social and political acceptability. From an infrastructure political-economy perspective, the "cloud" is an energy-situated device, territorially negotiated and politically contested.

At the national level, the Brazilian Artificial Intelligence Plan (PBIA) sets guidelines for research, innovation, workforce development, and AI adoption in the country, with direct implications for computational capacity demand and data governance [3]. However, implementing PBIA is not exhausted by the mere physical presence of data centers within national territory. Technological sovereignty requires institutional mechanisms capable of retaining value, qualifying negotiated commitments, regulating access to data, and establishing verifiable standards of transparency and auditing. From the standpoint of infrastructure studies, PBIA's materiality involves normative arrangements, energy and data regimes, supplier ecologies, and governance practices that determine who captures value, who bears costs, and who exercises regulatory power.

In parallel, the federal government instituted the Special Taxation Regime for Data Center Services (REDATA), designed to offer tax incentives and conditionalities aimed at attracting and expanding the sector [2]. The coexistence of PBIA and REDATA has reopened the public debate on how to reconcile objectives of technological sovereignty, sustainability requirements, and fiscal instruments. This discussion has intensified with the global surge in appetite for hyperscale facilities, which has repositioned data centers as strategic assets for industrial policy and as elements of federative competition for investment. From a political-economy standpoint, PBIA and REDATA operate in distinct registers: the former orients national research and innovation capabilities, while the latter emphasizes supply-side expansion via tax incentives. The tension between them highlights the country's longstanding difficulty in aligning industrial policy, sustainability, and digital sovereignty.

Against this backdrop, this paper contributes a systematized presentation of an applied proposal that emerges from the debate structured by SABIÁ—Brazilian Sovereignty and Autonomy in Artificial Intelligence—and is materialized in the document *AI Data Centers in Ceará: Strategy for Negotiation, Governance, and Sustainable Development* [13]. We argue that Ceará, by positioning itself in relation to an international hyperscale-associated problem, can turn external constraints into an opportunity for subnational leadership, provided it adopts policies grounded in verifiable commitments, data and energy governance, empirically grounded sustainability, and legal enforceability.

Through the lens of the political economy of infrastructure, this move translates into a territorial strategy for value capture in a sector marked by global structures of technological dependence. From the perspective of sociotechnical studies (STS), it means inscribing the arrival of data centers into local regimes of energy, water, and innovation, reconfiguring the relationship between territory, digital sovereignty, and industrial policy.

The remainder of this paper is structured as follows. Section 2 outlines the international challenge of hyperscale data centers, detailing their role as critical infrastructure and the material constraints of energy and water, alongside verifiable indicators for governance. Section 3 introduces the SABIÁ framework as a national strategy for digital sovereignty. Section 4 presents the empirical application of these principles in the State of Ceará. Section 5 details the results and ecosystem impacts, and Section 6 concludes the paper.

2 The international challenge of hyperscale data centers

2.1 Data centers as critical infrastructure

The global expansion of AI data centers is taking place in an environment where power grids, water availability, and territorial regulation become strategic bottlenecks. International reports point to a rapid increase in energy demand associated with data centers and networks, with significant impacts on power systems and equipment supply chains [9]. In response, different jurisdictions have adopted formal and operational restrictions, whether due to connection constraints, temporary moratoria, industrial prioritization criteria, or stricter socio-environmental requirements.

This movement shows that data centers are electricity-intensive infrastructures and, in certain architectures, also water-intensive; their social cost and territorial materiality are distributed asymmetrically. The rejection or containment of new projects, therefore, should not be read as merely symbolic, but as an attempt to avoid systemic overload, tariff increases, environmental degradation, and competition for critical resources.

Recognizing data centers as critical infrastructure thus shifts the debate from the abstract sphere of software to sociotechnical regimes that articulate energy, water, equipment, and territory—and, consequently, to regulatory and industrial policy disputes.

2.2 Energy and water as material constraints

Hyperscale exposes the material dependence of AI systems on the simultaneous consumption of electricity and water, turning resources historically treated as supporting inputs into structural constraints on expansion. Metrics such as *Power Usage Effectiveness* (PUE) and *Water Usage Effectiveness* (WUE) become central not only to measure operational efficiency, but also to inform industrial policy decisions, environmental licensing, and energy planning.

On the electricity side, the growing use of *Power Purchase Agreements* (PPAs) by AI companies seeks price predictability and competitiveness. However, such contracts can strain local markets by prioritizing inflexible loads, redistributing risks, and producing territorial asymmetries in development. In parallel, *curtailment* contexts—wasting renewable energy due to grid constraints—reshape the geography of hyperscale by turning systemic constraints into opportunities for energy arbitrage.

On the water side, WUE emerges as a decisive metric in regions subject to water stress, implying *trade-offs* between computational demand, climate security, agricultural use, and urban consumption. The availability of water for cooling, often underestimated in public debate, becomes a determining variable for data center siting and for sociotechnical controversies over resource prioritization, sustainability, and territorial justice.

From the political economy of infrastructure perspective, the interplay among PUE, WUE, PPAs, and *curtailment* reveals that AI expansion operates over material circuits of energy and water, whose governance is not neutral. For sociotechnical studies (STS), these dynamics reiterate that the "cloud" is a multi-scale device that articulates engineering, regulation, territory, and sociotechnical ecologies, redistributing costs, capturing value, and producing controversies.

Governing AI data centers requires distinguishing between operational-efficiency indicators, environmental-impact indicators, and institutional-governance indicators. First, energy efficiency is often expressed through *Power Usage Effectiveness* (PUE), defined as the ratio between the facility's total energy and the energy delivered to IT equipment [10]. Conceptually, the usefulness of PUE follows from the fact that cooling and power-conversion losses can dominate total consumption; thus, reducing the numerator while keeping the denominator constant increases the share of energy converted into computational service.

However, the literature points out limitations of PUE as a standalone criterion, since it does not capture the origin of energy, the efficiency of the compute fleet, or systemic impacts on the territory [6]. Therefore, to avoid oversimplified interpretations, PUE should be combined with granular measurement requirements, independent auditing, and decarbonization targets, as well as governance mechanisms that prevent the indicator from being used merely as a rhetorical instrument.

Second, the water component is formalized through *Water Usage Effectiveness* (WUE), standardized as liters of water per kWh associated with IT energy, with definitions consolidated in an international standard [11]. Here, the argumentative chain is straightforward: if a project consumes water at relevant volumes in regions under water stress, it amplifies social and environmental risk. Thus, in contexts of recurring drought, prioritizing reclaimed water and cooling technologies less dependent on evaporation becomes a requirement for water security, as has been discussed in Brazilian news regarding conditions for data centers in the Northeast [1].

Third, carbon and renewable-energy indicators complement the environmental reading: even a low PUE can coexist with high

emissions intensity if the marginal energy mix is fossil-based. Governance must therefore require periodic reporting on consumption, energy provenance, emissions, and audit standards; otherwise, information asymmetries and regulatory capture may prevail. REDATA, by establishing conditionalities for access to the regime, offers an institutional precedent for linking benefits to requirement verification, although such requirements must be operationalized through monitoring and enforcement instruments [2].

Table 1 consolidates a set of indicators and verifiable requirements, distinguishing internationally standardized metrics from governance indicators that can be contractually enforced by public authorities.

From this set, the logic of public policy becomes stronger: rather than assuming that installing data centers automatically yields development, it becomes possible to require evidence, monitoring, and enforceable commitments. This reduces the likelihood that a territory absorbs diffuse costs without proportional participation in the benefits.

A comparative reading of these cases indicates a recurring pattern: when transparency rules, indicator auditing, and binding commitments are absent, decisions tend to become reactive, culminating in connection restrictions or localized suspensions. In other words, room for subnational leadership emerges precisely when a territory recognizes the problem before a crisis and structures anticipatory governance mechanisms.

In the Brazilian case, investment attraction can be reorganized under a logic of qualified negotiation. Rather than competing only through tax incentives, public authorities can condition authorizations on verifiable obligations regarding energy efficiency, grid resilience, water security, metric transparency, and capacity transfer. This shift opens a strategic window for Ceará, which combines international connectivity and an established digital-infrastructure agenda, while also appearing in recent news about large-scale projects [15].

3 The SABIÁ Book

The SABIÁ book emerges as a Brazilian response to the growing concentration of technological power in large global corporations. It stems from discomfort with exporting our data, talent, and energy to foreign platforms, and from the hope of building a national intelligence agenda guided by science, ethics, and sovereignty.

Its formulation builds on an intellectual and political trajectory consolidated in the book "Digital Sovereignty, Colonization & Literacy" [14], which denounces new forms of algorithmic and infrastructural dependence. Both converge on an urgent repositioning for Brazil: moving from a passive consumer to a producer, regulator, and guardian of its own digital transformation. Today, sovereignty is not defended only with borders, but with code, servers, and collective awareness.

In this context, SABIÁ presents itself as a bridge between critical thinking and public action. Its purpose is to turn reflection into sovereignty—and sovereignty into the future. Conceived to support and accelerate PBIA, SABIÁ is structured as a national program of collaborative execution, involving universities, research centers, companies, public institutions, and civil society, with the goal of enabling the country to develop, apply, and regulate AI technologies with autonomy and in the public interest.

The approach is inspired by a successful Brazilian experience in technological governance: the Brazilian Digital Television System (SBTVD). Created during Lula's first administration (2003), SBTVD demonstrated the effectiveness of coordinated articulation among science, industry, and the state, mobilizing more than twenty R&D institutions, 1,500 researchers, and 60 laboratories, resulting in the Ginga middleware, recognized by the International Telecommunication Union as an international standard. This legacy of technological autonomy now inspires the conception of SABIÁ.

The book systematizes the distinction between technological sovereignty and technological autonomy: sovereignty refers to the political and institutional capacity to set rules; autonomy concerns the technical capacity to develop and maintain solutions without structural dependence [12]. This distinction is fundamental for understanding the arrival of data centers: their physical presence can coexist with dependence if higher-value processing, model governance, and economic appropriation remain externalized.

Beyond diagnosis, SABIÁ offers a propositional design: a framework that links PBIA acceleration to the creation of material conditions for execution—computational, institutional, and regulatory—preventing the plan from being reduced to guidelines without instruments. AI data centers are treated as critical infrastructure not only because of their scale, but because they underpin research, public services, and computation-intensive productive chains.

A central axis is the conversion of commitments into verifiable obligations. Instead of assuming that installing data centers automatically produces development, the framework proposes: (i) auditable impact metrics (with emphasis on energy and water); (ii) transparency and independent auditing; and (iii) social and technological commitments aimed at talent formation, R&D, and access to computing capacity for scientific and public institutions. In this way, sovereignty ceases to be an abstraction and becomes operational through clauses, indicators, and governance.

SABIÁ also positions PBIA and REDATA as potentially complementary instruments, as long as tax incentives do not replace binding commitments. In practical terms, capturing strategic value and reducing technological dependence require bargaining rules and monitoring; without them, the country risks hosting resource-intensive infrastructure with limited internalization of capabilities, knowledge, and income.

More than a technical project, SABIÁ is a state strategy and a gesture of national reconstruction, affirming that Brazil can create, innovate, and steer its technological future with intelligence, ethics, and sovereignty.

Aligned with PBIA, SABIÁ structures the program around five interdependent strategic goals:

- Consolidate a public, cooperative, and federated AI infrastructure (regional supercomputers, DATA-SABIÁ, and open platforms);
- Develop sovereign AI models, trained on Brazilian data and aligned with the country's cultural and linguistic diversity;
- Train talent distributed territorially, with emphasis on youth from peripheries and underserved regions;

Table 1: Indicators and verifiable requirements for governance and sustainability of AI data centers

Indicator	Operational definition	Basis	Suggested enforceability
PUE	Ratio between the data center's total energy and the energy delivered to IT.	ISO/IEC standard [10].	Instrumentation compatible with auditing; periodic disclosure; targets consistent with climate and cooling technology [6].
WUE	Liters of water per kWh associated with IT, as normatively delimited.	ISO/IEC standard [11].	Water inventory with water source, percentage of reuse, and contingency plan; integration with the territory's water-security policy.
Renewable energy	Percentage of effectively consumed energy sourced from verified renewables.	Contractual and regulatory evidence.	Documented proof; alignment with grid expansion and the regional energy mix; obligations linked to licensing.
Transparency and auditing	Publication of data series and independent auditing of indicators.	Regulatory and compliance best practices.	Third-party auditing; penalties for underreporting; reporting standards and reproducibility requirements.
Capacity-related commitments	Reserved capacity and investment in training, R&D, and the local ecosystem.	REDATA conditionalities [2].	Binding targets for R&D, training, and access to compute resources for STI institutions; access governance and accountability.
Jurisdiction and data access	Rules for data processing and access, with attention to extraterritorial legislation.	International legal debate [17].	Contractual clauses, key governance, access auditing, and localization requirements depending on data nature.

Table 2: International cases and controversy vectors associated with large-scale data centers

Location (example)	Critical resource	Externality and debate	Institutional response (evidence)
United States (high-concentration hubs)	Power grid and transmission	Rapid load growth associated with the expansion of large facilities, raising debates about planning and systemic cost.	Studies and sector discussions have increasingly treated data centers as a relevant load for grid planning and supply [9].
Ireland (Greater Dublin)	Energy and connection capacity	Data centers reached a significant share of national electricity consumption, amplifying concerns about security of supply.	Connection guidelines and measures with restrictions in grid-stressed regions and additional conditions for new projects [4, 5].
Singapore	Energy, emissions, and efficiency	Restrictions were adopted in view of energy limitations and carbon targets, with a restart conditioned on sustainability criteria.	Restart via competitive calls, with efficiency and decarbonization criteria as eligibility requirements [7, 8].
Netherlands (national guidance)	Land use and energy	Public debate on territorial compatibility of hyperscale facilities and their infrastructure costs.	Adoption of planning guidance and instruments to restrict/condition hyperscale data centers [16].

- Integrate universities, Federal Institutes, research centers, and local companies into a distributed network for innovation and technology transfer;
- Promote citizen-centered, transparent governance, ensuring social control over data, algorithms, and decisions.

4 A SABIÁ application: the "AI Data Centers in Ceará" proposal

4.1 The "AI Data Centers in Ceará" proposal

The document "AI Data Centers in Ceará: Strategy for Negotiation, Governance, and Sustainable Development" [13] is a proposal built from principles and guidelines of the SABIÁ book [12].

It is a concrete application of the framework, aimed at formulating public policy to attract hyperscale data centers to the State of Ceará. Its central purpose is to convert the arrival of major investors into territorial, scientific, and technological development, aligning incentives and commitments through anticipatory governance and verifiable metrics.

While SABIÁ operates at the national level as a strategy for digital sovereignty and PBIA acceleration, the Ceará document reorganizes that vision at the subnational scale, translating it into operational guidelines for negotiation, licensing, energy planning, social commitments, and institutional transparency. In doing so, it positions Ceará as a strategic actor in the geopolitics of data centers, mobilizing distinctive assets such as a renewable energy matrix, international connectivity, an innovation ecosystem, and world-class universities.

The document is structured as an instrument of state bargaining and regulatory capacity, organized around ten guidelines that treat data centers as development vectors rather than neutral infrastructure. Central elements include industrial and technological induction through commitments, talent formation, university–industry–state integration, public transparency, federated territorial governance, and the need for long-term energy and environmental planning.

4.2 Ceará as an empirical case

Beyond framing data centers as geoeconomic assets, the document argues that investment attraction must move from a race for incentives to qualified negotiation, in which fiscal concessions are conditioned on capacity transfer, applied research, open digital infrastructure, and mechanisms of technological sovereignty.

In this sense, PBIA and REDATA cease to be competing instruments and become complementary, provided that commitments are legally enforceable and auditable. This logic materializes in the ten proposed guidelines that structure the state's strategic positioning:

(1) Data centers as vectors for the future and sustainable development;
(2) The global opportunity and Ceará's strategic role;
(3) Tax incentives as leverage, not an end in themselves;
(4) Industrial and technological induction: the value of negotiated commitments;
(5) Data centers as geoeconomic assets;
(6) Qualified employability and local talent formation;
(7) Transparency, governance, and open innovation;
(8) Long-term energy and environmental planning;
(9) Binding environmental targets and compensation measures;
(10) Digital sovereignty and regional leadership.

Taken together, PBIA, REDATA, and the Ceará guidelines operate at different yet complementary levels: PBIA formulates the ambition; REDATA creates attractiveness; and Ceará develops mechanisms for territorial value capture and sovereignty. In the technology-governance literature, such an arrangement qualifies Ceará as a *subnational strategic node* for AI, where federative policies, critical infrastructure, and institutional capabilities translate into bargaining power and the production of national value.

Ceará is therefore a relevant empirical case for digital sovereignty studies because it combines rare material and institutional conditions: strategic infrastructure (renewable energy, submarine cables, and ports), international connectivity, positive territorial asymmetries, and subnational bargaining capacity. By shifting public policy from attraction to value capture, the state transforms incentives into enforceable commitments. And by operating within technological federalism and institutionally adopting the proposal, it turns a civil-society initiative into public policy, qualifying itself as a *subnational strategic node* in the AI geoeconomy.

5 Results

The results systematized by this article are organized into three complementary planes. First, the debate on digital sovereignty was previously structured in a work dedicated to problematizing coloniality and technological literacy, providing a conceptual basis to treat infrastructure as a political, economic, and institutional issue, through the book *Digital Sovereignty* [14]. This foundation matters because it shifts the analysis from technological promises to the material and distributive conditions of implementation.

Second, conceptual accumulation and a critical reading of tensions between public policy instruments converged into SABIÁ, which presents recommendations aimed at accelerating PBIA based on governance, auditable metrics, and enforceable commitments [12]. The public debate on compatibility between PBIA and REDATA reinforces the timeliness of the problem and the relevance of analytical instruments that avoid vague inferences and normative generalizations.

Third, the applied work materializes in the "AI Data Centers in Ceará" proposal, whose objective is to qualify the social, environmental, and economic negotiation of major investments by aligning commitments, governance, and transparency [13]. The proposal was delivered to the Governor and accepted by the state, resulting in the creation of an AI study group to address these issues. This institutional development becomes even more relevant given recent news about hyperscale projects in the territory, with potential to stress energy and water systems and, consequently, to demand anticipatory institutional design.

Finally, the chain of actions resulting from the initiative is entering the Ceará innovation ecosystem, impacting multiple strategic axes (Figure 1).

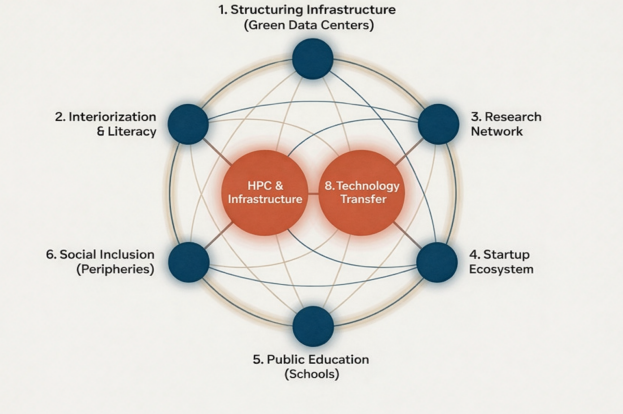

Figure 1: Strategic Axes (AI Data Centers in Ceará).

This means that young people with a computer at home can gain access to high-performance cloud computing networks, enabling innovative production. Research units in universities and innovation hubs at different scales can leverage the infrastructure to share data and support technology transfer.

The plan calls for a strong commitment-oriented effort in awareness, literacy, and digital transformation, enabling Ceará's citizens to access future technologies and think creatively about data-network uses, thereby building a culture of digital sovereignty.

In the areas of education and interiorization, Digital Culture programs develop content production that supports sovereignty-building processes. Resources are planned so that the development of intellectual property—such as the *games* industry—can benefit from negotiated commitments, connecting to strategic plans and current legal certainty in a structured way, as shown in the model in Figure 2.

Nonetheless, it is essential to recall that SABIÁ is a gateway for building an innovation identity and culture, supported by an asset arising from an opportunity generated by data centers in Ceará. The study group formed by organized civil society and government members will prioritize more mature environments for the practical application of negotiated social commitments.

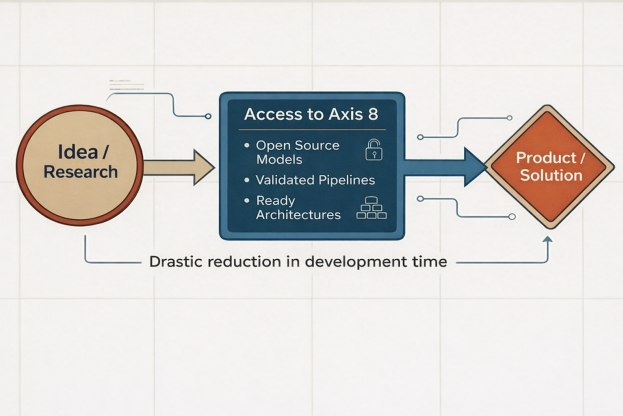

Figure 2: Detailed view of Axis 8 (Technology Transfer).

6 Conclusion

The international cases analyzed show that hyperscale data centers have become a global distributive problem: by exerting pressure on energy, water, and territory, they induce institutional responses that range from connection restrictions to new sustainability standards [4]. In this context, the opportunity for Ceará does not stem from any presumed natural advantage, but from its ability to structure anticipatory governance that converts infrastructure attraction into value retention, training, R&D, and transparency.

The articulation between PBIA and REDATA should therefore be treated as an institutional-design problem: incentives must be subordinated to verifiable requirements, and sovereignty must be operationalized through clauses, auditing, and indicators, with special attention to water security and grid resilience [2, 10, 11]. At this point, SABIÁ and the "AI Data Centers in Ceará" proposal offer a methodological path: turning metrics into obligations, commitments into public policy, and social participation into method [12, 13].

Applying this approach in deployment territories such as Ceará should be understood as a stage of empirical validation. The goal is not to claim results that do not yet exist, but to indicate how requirements and indicators can guide negotiations and licensing before impacts consolidate.

Finally, the initiative's social assets are being debated, with a focus on benefits for Ceará's citizens, institutions, and businesses. The models start from a social commitment with support orchestrated by the current ecosystem, impacting innovation sectors. In this way, Ceará can take the lead by offering a replicable governance model for AI critical infrastructure in Brazil—opening a subnational space for digital sovereignty and industrial policy applied to the hyperscale economy.

References

[1] Agência Brasil. 2025. Governo do ceará promete água de reuso para instalar data center. (Sept. 2025). Retrieved Jan. 16, 2026 from https://agenciabrasil.ebc.com.br/geral/noticia/2025-09/governo-do-ceara-promete-agua-de-reuso-para-instalar-data-center.

[2] Brasil. 2025. Medida provisória n. 1.318/2025 (redata): institui regime especial de tributação para serviços de data center. Retrieved Jan. 16, 2026 from https://www.planalto.gov.br/ccivil_03/_Ato2023-2026/2025/Mpv/mpv1318.htm.

[3] Brasil. Ministério da Ciência, Tecnologia e Inovação. 2025. Plano brasileiro de inteligência artificial (pbia) 2024–2028: ia para o bem de todos. (June 2025). Retrieved Jan. 16, 2026 from https://www.gov.br/mcti/pt-br/acompanhe-o-mcti/noticias/2025/06/mcti-lanca-plano-brasileiro-de-inteligencia-artificial-ia-para-o-bem-de-todos/pbia-ia-para-o-bem-de-todos.pdf.

[4] Central Statistics Office (Ireland). 2025. Data centres metered electricity consumption 2024: key findings. Retrieved Jan. 16, 2026 from https://www.cso.ie/en/releasesandpublications/ep/p-dcmec/datacentresmeteredelectricityconsumption2024/keyfindings/.

[5] Commission for Regulation of Utilities (CRU). 2021. Data centre connection policy: Decision paper. Tech. rep. CRU/21/124. CRU. Retrieved Jan. 16, 2026 from https://cruie-live.storage.googleapis.com/cru-media/documents/CRU212124.pdf.

[6] Miyuru Dayarathna, Yonggang Wen, and Rui Fan. 2016. Data center energy consumption modeling: a survey. *IEEE Communications Surveys & Tutorials*, 18, 1, 732–794. doi:10.1109/COMST.2015.2481183.

[7] Infocomm Media Development Authority (IMDA). 2022. Annex a: policy response to datacentre growth: background to the temporary pause and subsequent competitive allocation. (July 2022). Retrieved Jan. 16, 2026 from https://www.imda.gov.sg/-/media/imda/files/news-and-events/media-room/media-releases/2022/07/annex-a---summary-of-pilot-dc-cfa-key-parameters-and-criteria.pdf.

[8] Infocomm Media Development Authority (IMDA) and Singapore Economic Development Board (EDB). 2025. Imda and edb launch second call for application for data centre capacity (dc-cfa2). Retrieved Jan. 16, 2026 from https://www.imda.gov.sg/resources/press-releases-factsheets-and-speeches/press-releases/2025/imda-and-edb-launch-second-call-for-application-for-data-centre-capacity.

[9] International Energy Agency. 2024. Electricity 2024: Analysis and forecast to 2026. Tech. rep. IEA. Retrieved Jan. 16, 2026 from https://www.iea.org/reports/electricity-2024.

[10] ISO/IEC. 2016. Iso/iec 30134-2: data centres — key performance indicators — part 2: power usage effectiveness (pue). (2016). Retrieved Jan. 16, 2026 from https://www.iso.org/standard/30134-2.

[11] ISO/IEC. 2022. Iso/iec 30134-9:2022: data centres — key performance indicators — part 9: water usage effectiveness (wue). (2022). Retrieved Jan. 16, 2026 from https://www.iso.org/contents/data/standard/07/76/77692.html.

[12] M. Oliveira and G. Lemos. 2025. Sabiá: soberania e autonomia brasileira em inteligência artificial. Retrieved Jan. 16, 2026 from https://amauroboliveira.wordpress.com/wp-content/uploads/2026/01/2025_jan12_-_sabia_-_draft_25.pdf.

[13] Mauro Oliveira. 2025. Datacenters de ia no ceará: estratégia para negociação, governança e desenvolvimento sustentável. Public policy proposal document. Retrieved Jan. 16, 2026 from https://maurooliveira.blog/0-datacenters/.

[14] Mauro Oliveira. 2025. *Soberania Digital*. Omni Editora. ISBN: 9786501684239.

[15] Reuters. 2025. Omnia joins $9 billion tiktok data center project in brazil, expected to have 300 mw capacity. (Nov. 2025). Retrieved Jan. 16, 2026 from https://www.reuters.com/business/omnia-joins-9-billion-tiktok-data-center-project-brazil-expected-have-300-mw-capacity-2025-11-04/.

[16] Rijksoverheid (Netherlands). 2022. Voorbereidingsbesluit hyperscale datacenters. (Feb. 2022). Retrieved Jan. 16, 2026 from https://www.tweedekamer.nl/downloads/document?id=2022D06110.

[17] U.S. Congress. 2018. Clarifying lawful overseas use of data (cloud) act. Retrieved Jan. 16, 2026 from https://www.congress.gov/bill/115th-congress/house-bill/4943.

Voice of the Streets: a platform for urban violence detection based on social sensing

Eliel R. Silva
elielsilva@ufrj.br
PPGI – Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro, RJ, Brasil

Tiago C. de França
tcruzfanca@ufrrj.br
Departamento de Computação
Universidade Federal Rudal do Rio de Janeiro (UFRRJ)
Seropédica, RJ, Brasil

Jonice Oliveira
jonice@dcc.ufrj.br
PPGI – Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro, RJ, Brasil

Abstract

Urban violence poses a significant challenge for citizens and municipalities, necessitating strategies to effectively disseminate official information while residents seek timely access to critical information. This study presents 'Voice of the Streets (VOS)', an Information and Communications Technology (ICT) service designed to detect and classify violence in social media messages and understand its effects in physical environments. Using Design Science Research (DSR) as the methodology, this study developed a platform that integrates social sensing with a machine learning model to categorize the severity of violence. Results demonstrate the platform's feasibility in integrating external data sources, processing multiple data instances, and supporting the visualization of violence patterns.

CCS Concepts

• **Information systems** → **Spatial-temporal systems**; • **Human-centered computing** → *Computer supported cooperative work*; • **Applied computing** → Law, social and behavioral sciences.

Keywords

Social Sensing, Violence Detection, Social Data Management

1 Introduction

An urban environment is an agglomeration of continuous built-up areas, with a high frequency of activities that involve and affect the geographical space [15]. These activities are related to complex dynamics, such as active commuting, physical spaces maintenance, music concerts, and urban violence. Understanding these dynamics is becoming important as the number of people living in cities grows. To obtain a complete picture, about 4.9 million people live in areas affected by organized crime in Rio de Janeiro [21].

Urban Violence is a type of spatiotemporal event that occurs in physical environments like cities, towns, and metropolitan areas. Examples include violent crimes, robbery, vandalism, gang wars [2, 10]. This kind of event brings adverse impacts on society. For example, in 2024, gunfire clashes caused the suspension of classes in 368 public schools in Rio de Janeiro [12]; in 2023, for example, the Brazilian government spent on public security investments about R$ 18.785 billion, corresponding to a 13% increase compared to 2022, when R$16.629 billion were spent. It is essential to detect such events in near real-time and with high accuracy to respond to these challenges and reduce the cost to society.

Meanwhile, the presence of inhabitants in urban dynamics, such as urban violence, represents an opportunity to build collective knowledge about spatiotemporal dynamics. People are an excellent source of urban data compared with traditional physical sensor-based architectures [19]. Today, thanks to tools such as social media, mobile devices, and wearables, each person can be a broadcast source [19]. These combinations offer an opportunity to build systems that leverage the ubiquitous capabilities of social sensing.

In countries like Brazil there are official channels for reporting crime, these mechanisms often involve unidirectional communication that does not facilitate immediate interaction between citizens. On the other hand, the use of social media to report data is an already recognized tool to report crimes in Brazil, with successful examples like "Onde Tem Tiroteio (OTT) [1]" and "Fogo Cruzado" [2]. In this scenario, the present study seeks to answer the following research question: How can a method be developed to systematically detect and classify levels of violent events in social media text posts?

This paper proposes a software-based approach, Voice of the Streets (VOS), to detect and classify violence reports from human sensors. The proposal, which is a work in progress, uses sensors, machine learning, and notification methods to monitor the city and provides an interface for consuming processed social sensing data. Our goal is to define an approach that could serve as a detection agent for an urban violent event, based on observations of citizens already embedded in the daily routine of urban space.

The present work is organized as follows: section 2 presents the research methodology; Section 3 introduces the fundamental concepts to understand our proposal; section 4 presents the related literature and the difference to this work;section 5 presents VOS and provides a conceptual system description; section 6 describes the prototype implemented to validate the feasibility of the VOS; and Section 7 presents the discussion followed by section 8 final considerations and future directions.

2 Methodology

The motivation for VOS development is a lack of appropriate tools for using social sensing data and Information and Communication Technologies (ICTs) to understand patterns of violence severity in urban environments. To address this issue, the present work adopts the implementation of Design Science Research (DSR) proposed by [4].

[1] https://ondetemtiroteio.com/website/ott/index.html
[2] https://fogocruzado.org.br/

DSR is a research method whose objectives involve proposing, creating, and using artifacts that intervene in real-world situations to improve or solve problems [4]. Our adaptation has eleven stages: problem identification; problem Awareness and literature Review; Identification of Artifacts and Configuration of Problem Classes; Proposal of Artifacts to solve the Specific Problem; Design of the Selected Artifact; Artifact Development; Artifact Evaluation; Explication of Learning; Conclusions; Generalization for a Class of Problems; Communication of Results.

The problem identification began with a previous study that proposed a model to detect patterns in general urban named CidadeSocial [15]. While using the proposed architecture for a specific dynamic — urban violence — it was recognized that there was a lack of a sensing platform dedicated to detecting and classifying the level of violence. Even though the project had already explored communication in urban settings, it lacked an artifact that enables automated analysis and knowledge generation based on human-sensing observations.

The following stages, problem awareness and literature review, were conducted interactively. Initially, the CidadeSocial was chosen as a base model for supporting communication in urban environments [15]. This work confirmed the wide range of urban dynamics, ranging from topics such as urban mobility to public services. In a second iteration, during the development of a master's degree dissertation, the model was specialized to urban violence [16]. The understanding from the 'problem awareness' stage worked as an input for the literature review. A targeted review was conducted across Google Scholar, the IEEE Digital Library, the Brazilian Symposium on Multimedia and the Web (WebMedia) 2025 Proceedings, and HAL's open archive, with a specific focus on publications from the ADVANCE workshop.

Based on the results of the targeted review, we analyzed the characteristics of each related work and evaluated the extent to which each solution addresses our research gap. Afterward, in the artifact building, development, and evaluation, we proposed a system that combines techniques such as social sensing, machine learning, and software development to address the requirements. The 'explication of learnings' and 'communication of results' are the corresponding phases that explain the results achieved and that have led to this research paper.

3 Fundamental Concepts

3.1 CidadeSocial: humans into social sensors

CidadeSocial is an information system focused on sharing information via mobile devices through crowdsourcing. The project was initially named UFRJ Social [7] to facilitate information exchange among the academic community across different university campuses. After observing similarities between the dynamics in universities and other urban environments, the application was named CidadeSocial [3]. CidadeSocial was also chosen as part of the framework to demonstrate the integration of social sensing data into emergency preparedness strategies [5]. As a last step in maturity, the project evolved to an architecture that supports unplanned communication [15] and also enables understanding urban dynamics, such as violence [16].

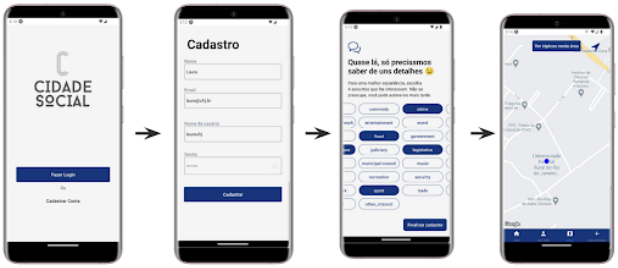

Figure 1: The registration and initial setup flow. From left to right: the landing screen, the user data entry form, the preference selection interface, and the home screen displaying the geolocation dashboard. View full-size image

In the CidadeSocial, users can exchange contextualized information about events and infrastructure (e.g., events, problems, news, and security incidents in the city) without needing to belong to any group or interact with acquaintances on social media. In addition, the application is always concerned with maintaining a summary of the user's location on their device. This way, it remains operational even if the person has no signal (which is essential, considering that in many places there is no Wi-Fi or the mobile connection is weak).

In this mobile application, users can retrieve information using tags, or "interests", and their geolocation. There is an active recommendation, meaning that when entering a location, the user is notified of topics of interest.

CidadeSocial was chosen as the platform to facilitate the pilot studies at the "Voice of the Streets" platform for the following reasons: VOS is part of the CidadeSocial project's ecosystem. However, it is essential to note that VOS is source-agnostic. In other words, it can be integrated with other text-based sources.

3.2 Detecting and Classifying Urban Violence with VISAGE

The concept of violence has a substantial role in the functioning of VOS, as their goal is to detect violence from social media text messages and understand their level of severity and what story it tells about a physical environment. To quantify violence, the present work uses the proposal from the previous work of [17], named VISAGE.

At VISAGE, Souza et al. [17] adapted the QOVS — Quantification of Violence Scale — a proposal made by Tyrer et al.[18], which analyses three dimensions from a violent report: planning, intention, and consequence. Each dimension can receive a numerical score, ranging from 1 to 6 for the planning dimension, and 0 to 5 for intention and consequence [18]. The sum of these three components yields a total score that quantifies the overall magnitude of the violent episode.

VISAGE is a computational approach that turns QOVS into an automated model to classify levels of violence. To simplify the classification process, VISAGE organises violent reports into classes. The main idea is that these classes function as levels on a scale, with the first position encompassing situations with no violence and the

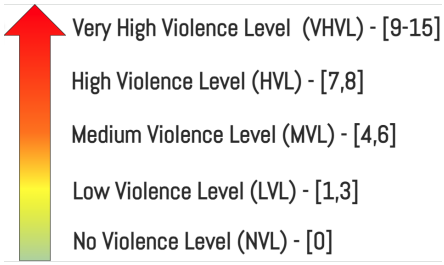

Figure 2: Adaptation of QOVS made by [17]. View full-size image.

last encompassing extremely high violence. To this end, five classes were defined based on specific score ranges as specified by Tyrer et al. [18] and represented in Figure 2. The violence is calculated based on the score of a violent report on a topic. As mentioned earlier, violent episodes can motivate citizens to report them on social media. The purpose of these classes is to assess the degree of violence in this report. A Multinomial Naive Bayes classifier was implemented within VISAGE to automate the classification of messages based on their violence levels.

4 Related works

Spatial-temporal event detection is a collection of processes that includes (i) sensing (e.g., social sensing), (ii) extraction of events of interest, and (iii) the application of actionable events detected [20]. Interest in this type of research grew due to the importance of dynamics such as disasters, health, the environment, crises, and crime. This section presents some current research that uses a type of sensing to detect or address urban violence.

Rocha et al. proposed two unsupervised and complementary heuristics, based on temporal and entity recognition approaches, to cluster videos related to violent events in cities [13]. Compared with state-of-the-art models like GPT-4, their approach performed better in sparse-event scenarios. The difference between our work is in the sensing and application of the event detected. While Rocha limited their work to data from YouTube, the present research is platform-agnostic, and the aim is to understand which patterns posts about a violent event reveal about the city.

Pongpaichet et al. proposed an intelligent system designed for extracting and visualizing crime metadata from vast corpora of online news articles [9]. Unlike general approaches based on social media content, they built their sensing process on news from reliable Thai sources. By applying a multi-class classifier and Named Entity Recognition, they categorized each news article into seven crime types, built a deep learning cross-lingual model to extract this information, and built visualizations. The main difference with this work is the application of the detected event, whereas in [9] the system was designed for monitoring violence; the present work aims to help return this information to human sensors. In this context, human sensors are the individuals who act as observers and providers of information about physical environments.

Heredia et al. presented an interactive system to explore and explain spatial-temporal anomalies in urban data, using spatial-temporal visualizations and Large Language Models to generate explainable contexts [6]. The proposed system addresses the lack of explainability in solutions for spatial-temporal event detection in urban environments. The main difference in this work is that it focuses on statistical methods rather than AI-based explainability.

In the context of urban dynamics, [15] has proposed a general model for detecting patterns. The paper proposed an architecture based on geolocation and interests that enables humans to act as sophisticated sensors, even without a prior connection. As a result, the authors implemented a platform called CidadeSocial, with utility for general scenarios such as urban mobility and community engagement. Although this work is a continuation, the focus here is on developing a platform that can sense violence and help understand how it operates in urban environments like the city of Rio de Janeiro.

5 VOS plataform

The VOS platform is a social sensing software designed to identify and classify violence in social media messages systematically. Also, the VOS platform allows sending notifications for the social sensors, which represents a way to implement non-planned communication — communication where people do not necessarily know each other [15, 16].

According to [19], a social sensing application can be classified as a model-centric application. This type of system uses generalizable models built from sensory data, and it can support human decision-making beyond the immediate location or context of collection. VOS fits into this classification because it uses social media data to detect and classify the level of violence in a post. Also, the information about the level of violence is directly related to the geolocation. Thus, reports of violence not only identify a specific occurrence but also reveal spatial patterns, allowing the understanding of how patterns of violence across different regions of the city correspond to their severity.

In the following subsections, there are descriptions of: VOS features; system architecture; a sample implementation; and the machine learning building process.

5.1 System Entities

As previously mentioned, violence is the primary entity of VOS. However, because it is a system that performs analyses and inferences using social media data, integration with other social media platforms requires additional entities.

The first entity, `User`, represents humans who produce external data or interact with the system. Each user instance includes basic information such as username, email, and password for local authentication. Additionally, geolocation data is stored. There are two user types: local and external. Local users are created directly on the platform to manage and visualize data. External users are instances created to represent authors of posts from CidadeSocial.

The `Topic` entity represents messages posted on other social media platforms. This entity has a struct containing fields such as `title`, `description`, `latitude`, and `longitude`. The-re are additional attributes that represent more complex geometric representations, facilitating spatial analysis. It is worth noting that VOS does not create posts, but they originate on another social media

platform. This entity also stores attributes that record the creation and update dates, enabling analysis of temporal trends.

The entity `Violence` is directly associated with a single topic and captures information related to episodes of violence predicted in the context of the system. Among its attributes are the unique identifier (`id`), the class indicated by the machine learning system (`predicted_class`)—the classes being modeled in the QOVS extension [18] — and the creation (`created_at`) and update (`updated_at`) dates. This entity is used for analyzing sensitive topics, enabling automatic classification and event logging.

The `Interest` entity represents categories that can be associated with multiple users and topics. The interest might work as a tag of a post or even the subject of a text. Combined with violence and location, this entity helps understand the main ts attributes, including `name`, the name of the interest; `description`, details; and the `created_at` and `updated_at` dates.

The `MLModel` entity stores information about the machine learning models used in the system. Attributes include `name` and `description` to identify and describe the model, `file _path` for the path of the stored file, `is_active` to indicate whether the model is in use, as well as `type_of_model` and `metric` that characterize the type and evaluation metric of the model, respectively. There are fields that stores the creation and update dates, as well as the identifier of the user responsible for the upload.

The `Alert` entity was designed to manage alert issuance from the VOS. Its attributes include `urgency` for the alert's urgency level, `expiration_date` for the expiration date, and the fields `headings` and `contents`, which contain the alert's title and content, respectively. The creation (`created_at`) and update (`updated_at`) dates are also recorded. Essentially, an alert can only be created by a single local user.

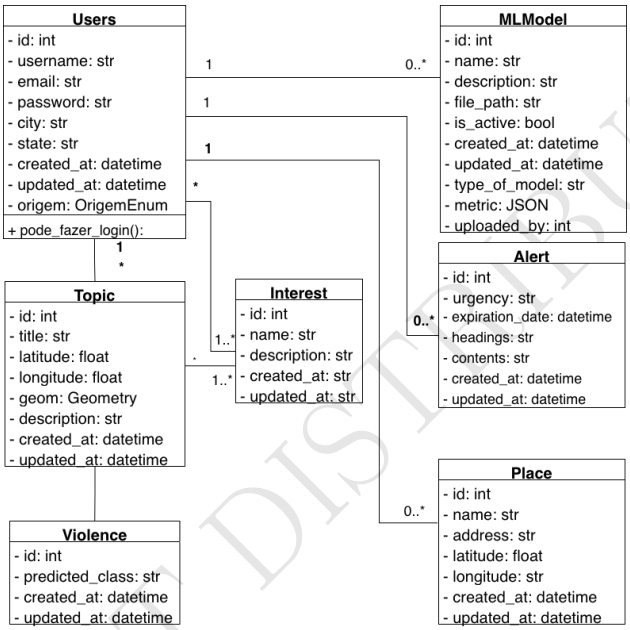

Figure 3: Domain model of the server-side application.View full-size image.

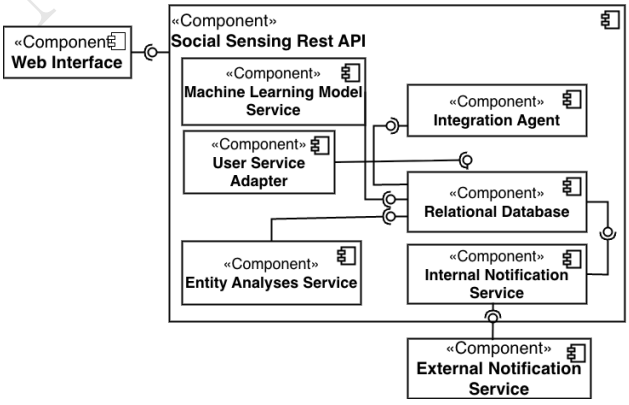

Figure 4: High-level component diagram of the server-side architecture of the 'Social Sensing Rest API'.

5.2 Architecture

In terms of architecture, VOS comprises three main components: the web interface, the social sensing REST API, and the external notification service. The core component is the REST API, which manages all requests and redirects them to the appropriate internal components. Also, the API is responsible for coordinating the logic of sending alerts to the end user via an external notification system. It consists of six subcomponents: `Machine Learning Model Service`, `Integration Agent`, `User Service Adapter`, `Relational Database`, `Entity Analysis Service`, and `Internal Notification Service`.

The `Machine Learning Service` is responsible for classifying violence present in social media text posts based on the method and models generated at [17]. It is worth noting that this service is out of scope to cover the entire Machine Learning Lifecycle [3]. This architecture is designed to receive a trained model, excluding steps such as data collection, pre-processing, and model training. The results of this component are stored in the `Relational Database`.

The `Integration Agent` performs scheduled tasks to reconcile entities from social media and other configured social sensing data sources, including users, topics, interests, and places. The frequency of automated tasks is configurable, and each entity has its reconciliation task performed separately. The data originates from the `Relational Database` component of Application Server 1.

The `Internal Notification Service` is responsible for sending and recording alerts. When a user uses the `Web Interface` component to send an alert, the Internal Notification Service will send the data to the External Notification Service and record the information from this event in the `Relational Database`. This component is only used by the `Web Interface` component.

[3] data collection, feasibility study, documentation, model monitoring, and model risk assessment

5.3 Implementation

Accordingly, within the methodology, a solution to a real-world problem must be implemented and validated. To accomplish this, all the described components and entities were implemented to test the architecture's feasibility. In the following subsections, we describe the implementation of the Social Sensing Rest API, the VOS Web Interface, and the External Notification Service.

5.3.1 Social Sensing Rest API. The Social Sensing Rest API was implemented using the Python programming language and the FastAPI framework [4] — a tool for building APIs in Python. This tool was chosen because it enables the integration of machine learning models built in the same programming language. This component has six other subcomponents.

The implementation was organized using a layered architectural approach, separating code for handling HTTP requests (routers), business logic (services), and data models (models). There is also code responsible for orchestrating data integration and analysis tasks called Task Scheduling. The table brings an example of sample routes. The Table 1 shows sample REST endpoints.

The `Relational Database` was implemented using PostgreSQL [5], an open-source relational database system (RDS). The choice of this tool over other RDS tools was due to support for Geometric Types. This data type provides a built-in geometric data type that represents 2D spatial objects on a planar surface. As a result, the geometric data type allows the social sensing API to perform the requested geo queries.

As defined before, the `Integration Agent` searches for data in an external source and persists it in the Relational Database. At the implementation level, a class called Integration has methods like `get_topics` and `get_places` that query an API and persist the related entities in the database; for example, `save_topics` persists the data from `get_topics`. Also, in the implementation, the Scheduler class defines the periodic execution of the integration logic.

The `User Service Adapter` manages the operations related to Users. At the router layer, some methods expose endpoints for CRUD [6] operations on the User Entity. In the business logic layer, there are methods that save user data from the external API to the local User model. All the produced data is persisted in the Relational Database.

The `Machine Learning Model Service` was implemented to perform three operations: manage pickle files [7]; performing predictions; save predictions result. At the router layer, there is only one endpoint to handle requests to create and update these models via HTTP requests. In the Business logic layer, there is the `MLService` class, which deserializes the model and predicts the violence level from a text. This classification engine uses a Bayesian model (Multimodel Naive Bayes) to categorize messages according to their severity. Based on prior evaluations conducted within the VISAGE framework, the chosen model achieved an accuracy of 84.42% and a weighted F1-Score of 0.81.

[4] https://fastapi.tiangolo.com/
[5] https://www.postgresql.org/
[6] Create, Read, Update, and Delete
[7] The pickle module implements binary protocols for serializing and deserializing a Python object structure. "Pickling" is the process whereby a Python object hierarchy is converted into a byte stream, and "unpickling" is the inverse operation [11].

The `Entity Analysis Service` analyses records of topics stored in the database to generate statistics and identify patterns related to violence. In terms of routing, there are endpoints for defined queries such as: a group of topics within a time range; the number of topics per violence level; topics posted within a square area. For business logic, schedulers define the frequency at which analyses are executed.

The last subcomponent is the `Internal Notification` Service, which handles creating and sending notifications to the External Notification Service. In the routing, there is an endpoint to trigger or send a JSON object with the notification details. In the business logic layer, there is a class named `OneSignalService`. This class contains all the logic to authenticate with the external API and send data to the Social Media Platform.

Table 1: Sample Social Sensing Rest API Endpoints

Method	Endpoint	**Description**
`POST`	`/auth/token`	Login for access token
`GET`	`/topics/`	List topics with filters
`POST`	`/topics/`	Create a new topic
`GET`	`/violence/class`	Get violence count by class
`POST`	`/mlmodels/{id}/predict`	Make a prediction using a specific model instance

5.3.2 Web Interface. The Web Interface component is the front-end application that allows the user interaction with the data produced and handled by `Social Sensing Rest API`. The Web Interface was implemented as a Single Page Application (SPA) built with React [8] version 18. The user interface is strongly based on the Material UI [9] (MUI v6). The application is organized in 8 different React components: (i) Dashboard, (ii) ClassesViolencia, (iii) Home, (iv) MapPage, (v)Reports, (vi)Settings, (vii)UserProfile. The general geographical data visualization was built using the Chart.js [10]. The Table 2 has a description of each implemented table, and the Figure 5 shows a screenshot of MapPage.

5.3.3 Notification system. Sending Notifications is an essential feature in VOS. In this implementation, it was considered sending notifications to the Android application of CidadeSocial [15] to validate the concept of creating communication from social sensing data. To achieve this, the `Internal Notification Service` was integrated with an `External Notification Service`, a third-party service for sending notifications.

Considering that the notification was implemented only for Android devices, the tools to implement external notifications are: `OneSignal` to send text to phones; and `Firebase Cloud Messaging (FCM)` [11] to handle phone infrastructure. The integration between

[8] https://react.dev/
[9] https://mui.com/material-ui/
[10] https://www.chartjs.org/
[11] https://firebase.google.com/

these two is based on the device token, a primary identifier generated by FCM for each Android device.

This token will serve as the primary address, allowing the notification server to route notifications to devices with the CidadeSocial installed. It is important to note that, without this identifier, it would not be possible to define the correct target.

Table 2: Main components of the social sensing platform.

Component – Description
(i)`Dashboard.jsx` – Main component containing all other pages related to the social sensing application.
(ii)`ClassesViolencia.jsx` – Displays statistical information regarding topic violence classification, such as the number of events per topic.
(iii)`Home.jsx` – Renders the dashboard home page with a list of topics published by application users.
(iv)`MapPage.jsx` – Contains all maps of the social sensing platform, such as the topic map and the heatmap.
(v)`Reports.jsx` – Generates and displays analytical reports on data collected in the platform, including violence statistics, user interactions, and trends.
(vi)`Settings.jsx` – Displays and allows modification of platform settings, such as user preferences, permissions, and notifications.
(vii)`UserProfile.jsx` – Displays and edits user profile information, including personal data, interaction history, and display preferences.

6 Experimental Evaluation of the Artifact

This section corresponds to the artifact evaluation stage proposed by [4]. The evaluation aims to determine whether the VOS can systematically detect and classify levels of violence in social media text posts. To verify this section, it uses three indicators: (i) the capacity to integrate with data generated in a social media platform; (ii) the capacity to produce insights based on social media data; (iii) the capacity to process multiple topics based on their level of severity.

It is important to note that all user interface screenshots are originally in Portuguese, considering that the case study was conducted in a Brazilian context. To ensure clarity for international readers, English translations are provided in the figure captions and corresponding descriptions.

6.1 Pilot Study 1 - Integration with External Sources

This test validates the capability to connect VOS to external sources. In this scenario, a violent event is simulated in the city of Rio de Janeiro. For the pilot study, six non-human test users in the CidadeSocial Application. The login information is described in Table 3. The creation process was repeated for all users after the user administrator was created at VOS.

The next step was to generate data that simulated user interactions. The main goal of this stage was to create an alert in VOS. `Ana Souza` started a thread by posting and sharing a question about a traffic jam on a highway. Her post draws attention to the local situation, generating a series of responses that accumulate additional information through comments from other users, such as Bruno Lima, Carla Mendes, and Felipe Santos, as shown in Figure 6.

Table 3: List of users with emails and usernames

Name	Email	Username
Ana Souza	ana.souza@example	ana.souza
Bruno Lima	bruno.lima@example	bruno.lima
Carla Mendes	carla.mendes@example	carla.mendes
Daniel Oliveira	daniel.oliveira@example	daniel.oliveira
Eduarda Castro	eduarda.castro@example	eduarda.castro
Felipe Santos	felipe.santos@example	felipe.santos
Gabriela Ferreira	gabriela.ferreira@example	gabriela.ferreira

In our pilot test, the administrator user logged into VOS could see all post interactions after the scheduled execution of the integration components. Thus, the admin user can decide to send an alert reporting the violent episode to all users with the mobile app installed via push notification. The Figure 7 shows screenshots of VOS to visualize the same posts made on CidadeSocial and the alert on an Android mobile.

6.2 Pilot Study 2 - Processing multiple data instances

The multi-topic sensing test aims to evaluate the social sensing platform's ability to process many topics. The test process is described in Figure 8. The dataset used for this step is the same as that used at [17]. The data is a stratified sample of 1745 posts from the X [12] social media platform. All posts are from the Brazilian context and cover: Rocinha police clashes in 2017, messages about the 2013 Brazilian protests, posts about the military police strike in Espírito Santo in 2017, and posts from profiles that talk about security (like @fogocruzadoapp).

The dataset label process was manual. Two master's degree students, native Portuguese speakers, conducted the classification of messages into five levels of severity (NVI, lVL, mVL, HVL, and VHVL) based on the adaptation of QOVS (planning, intent, and consequence). As a result, the dataset has 1.093 NVI, 491 VHVL, 109 HVL, 29 MVL, and 29 LVL.

Each row of the dataset used for machine learning training and testing is associated with a Cidade Social user (according to Table 16), an interest, and a fictitious location. It is worth noting that the locations are in the "Ilha do Fundão" region and near the central area of the city of Rio de Janeiro. This choice aims to facilitate the visualization of topics on the map. Regarding interests, if the text has any level of violence associated with it, it receives the label "crime"; otherwise, it will receive the label "others." Once combined, the topics are persisted in the VOS relational database.

[12] x.com

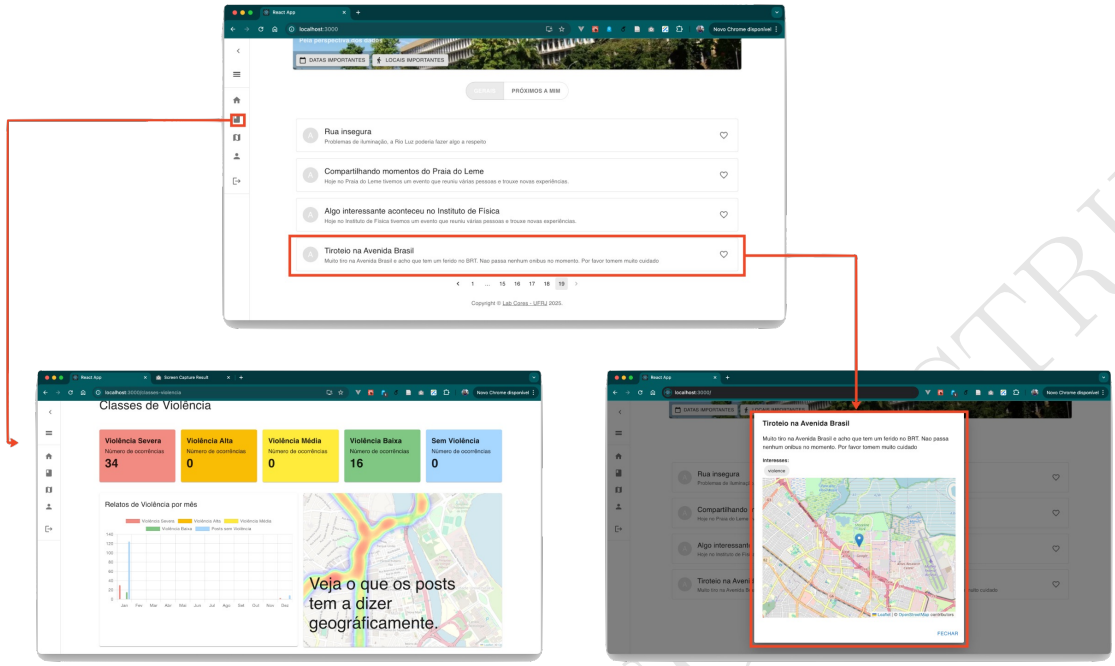

Figure 5: Navigation Flow at VOS. The central screen shows the feed of collected posts. The red arrows display the interaction paths: accessing the data summary with statistical charts and heatmaps (bottom left), and inspecting the details and geolocation of a post (bottom right). View full-size image.

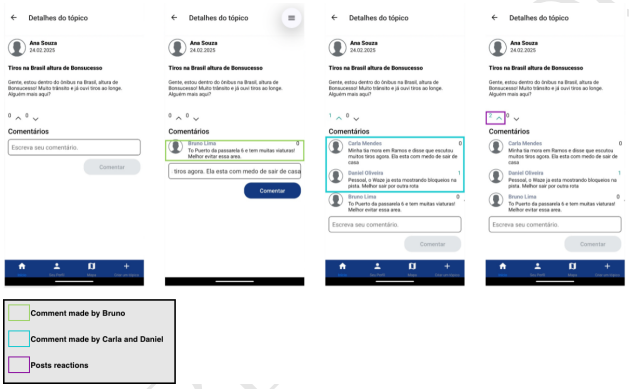

Figure 6: Topic details and interaction flow. From left to right, the screens depict the expansion of a conversation thread. The highlights indicate distinct user contributions: individual comments (green and cyan boxes) and the engagement counter (purple box) representing the topic's reach. View full-size image.

After the topics are generated and persisted, the automated routines of the social sensing platform are executed: (1) topic synchronization and (2) violence classification. The first routine copies the published topics to the platform's relational database. Next, the violence classification routine queries the new topics and applies the implemented machine learning model. The classification results are then available for viewing on the platform. The complete flow of this test is illustrated in Figure 8.

The topics in VOS are initially displayed in a list, where the original tweets are converted into formatted text posts. By selecting a specific topic, you can access details such as title, full text, associated interest, and location. Figure 5 shows a concrete example of this display.

6.3 Pilot Study 3 - Producing insights based on social media data

Pilot Test 3 aims to validate the platform's ability to generate insights into the physical environment. As a setup for this stage, the same setup as generated for the Pilot Study 2 was used. After persisting the data in the VOS relational database, it was possible to view aggregated details of the imported posts, such as the temporal distribution of publications and the number of posts per class of violence. After automatic classification, the numbers obtained were: 1,093 publications without violence, 491 with severe violence, 109 with high violence, 29 with medium violence, and 29 with low violence. These values correspond exactly to the distribution of samples per class of violence in the labeled dataset, as shown in Figure 5.

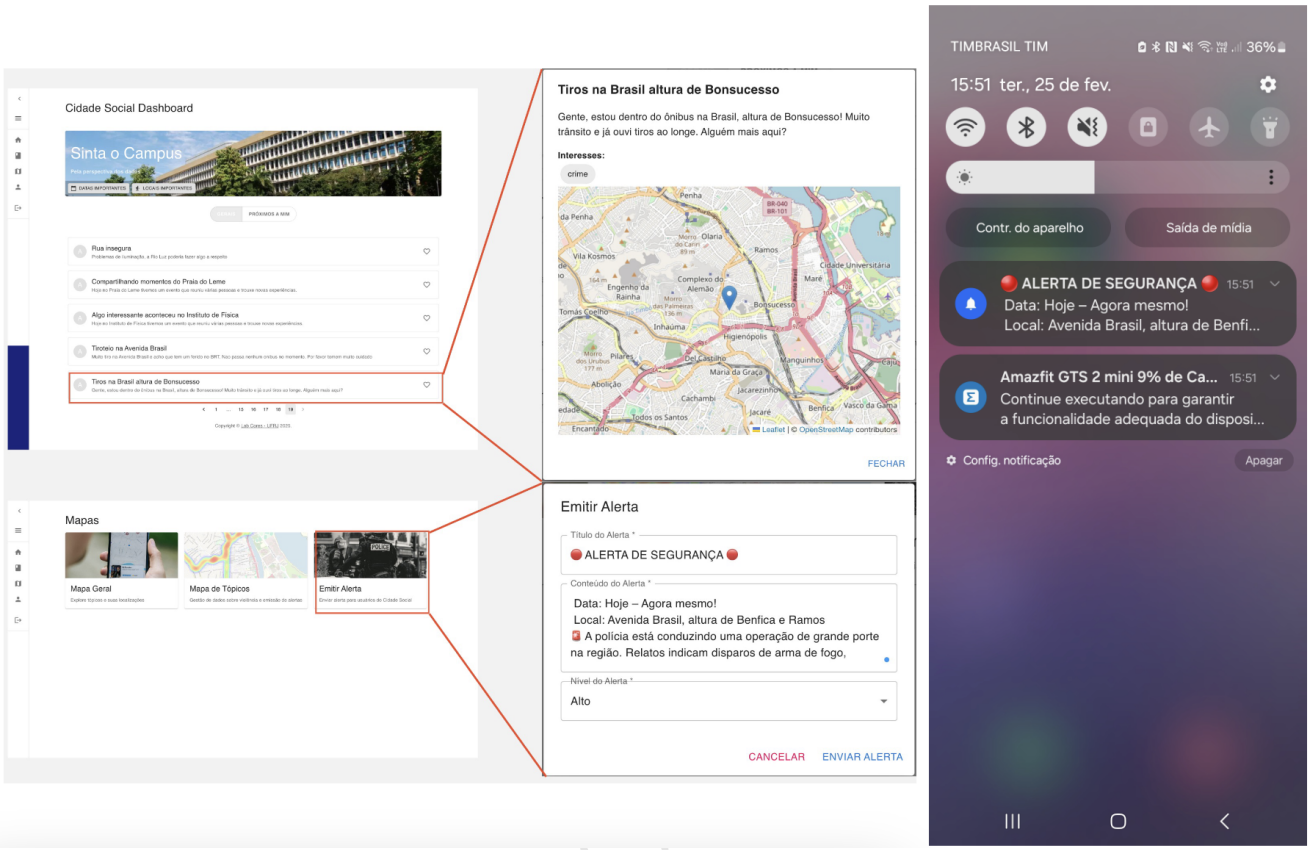

Figure 7: Cross-platform communication mechanism. The figure depicts the transition from the VOS (left), where safety threats are identified and composed into alerts (center), to final dissemination via the mobile notification system (right).View full-size image.

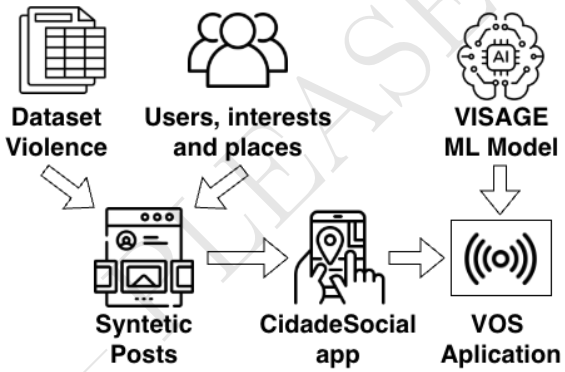

Figure 8: Illustration of multi-topic sensing test.

The other type of visualization produced at VOS is understanding the density of posts in a region. Since the posts contain latitude and longitude information, the map allows you to view the exact location of each publication and identify areas of higher concentration based on publication density. Two types of visualization were used: publication point map and density map. Figure 9 shows the comparison between an area with high concentration and another with low concentration of posts.

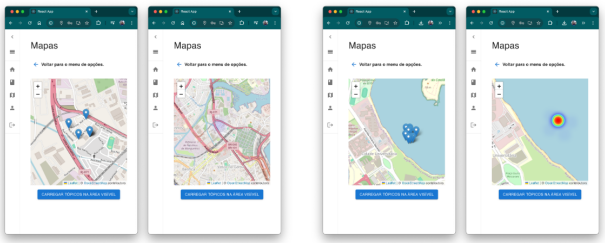

Figure 9: Comparison between maps with low and high topic density in VOS.View full-size image.

7 Discussion

The tests carried out to validate the architecture demonstrate that it fulfilled its primary objective: enabling unplanned communication

in scenarios characterized by so-called urban violence. The ability to integrate information technology components, such as mobile devices, content recommendation algorithms, machine learning models, and notification services, proved functional in practice.

However, some limitations have been identified and warrant highlighting. Dependence on third-party alerting services introduces variable communication latencies. In addition, the need for human moderation to send alerts, although necessary to avoid false positives, is a bottleneck that can slow notification speed.

Another test performed was an experiment that emulates user interaction on the platform. The test sought to track the flow of information between people. Although there are significant differences in a real scenario—such as latency, differences in device hardware, and asynchronous communication—the test created a context in which people exchanged information without needing to know each other beforehand, acting as sensors that allowed the moderating agent to issue the alert. The alert was successfully issued, allowing users who did not interact directly to communicate promptly.

8 Final Considerations

This paper presented a social sensing platform for detecting and classifying urban violence using textual posts from social media. This platform is named Voice of the Streets. Data produced by humans in urban settings can provide information about physical environments. VOS supports the integration with social sensing data sources, the generation of statistics-based knowledge from this data, the use of a machine learning model to detect and classify violence, and the capability to produce alerts for people involved in the urban dynamic.

8.1 Challenges to the validity of the proposal

In addition to demonstrating the feasibility of the work, it is worth noting that the implementation and experiments conducted with the software architecture served to understand the limitations of the proposal presented in this study.

From an ethical perspective, this work presents an artifact that directly addresses data and metadata related to human beings. The misuse of such data, from a moral standpoint, could put the lives of several users at risk. For example, false negatives produced by the social sensing platform could lead to content recommendations that are not aligned with the actual situation at the time. In the case of extremely violent events, people could be placed in situations that put their lives at risk.

Another concern is that recommendations and information generated on the platform could be misused. For example, when it is understood that a point in the city is safe, or when it is made clear that it is a point of interest, malicious users can use this information to be in these locations and commit crimes. Finally, the spread of fake news and other dangerous, biased information is a challenge not exclusive to large social media platforms. The lack of a fact-checking system currently limits the VOS.

8.2 Contributions

The contributions of this work are part of a set of scientific and technological efforts aligned with two sets of challenges in computer science research: Major challenges in computer science in Brazil proposed by the Brazilian Computer Society (SBC) at its 3rd seminar to define the themes [14]; Major Challenges in Information Systems Research in Brazil proposed by the Special Commission on Information Systems (CESI)[1].

One challenge addressed is the discovery of patterns in the context of Smart Cities. The article presents the construction and evaluation of a tool based on social sensing and machine learning that classifies reports of urban violence by severity. This contribution also enables the construction of statistical data, which aligns with Smart City data statistics [14].

When we talk about the challenges posed by CESI, this dissertation also aligns with challenge 2—Information Systems and the Open World Challenges —proposed in GranDSI-BR [1]. The social sensing architecture directly encourages citizens to participate in the urban context by sharing information without prior preparation. The work therefore contributes to the discussion on how information systems can foster new forms of interaction between government and society, with the potential to build more inclusive and democratic digital ecosystems.

Moreover, this work contributes to the context of SDG 16 (Peace, Justice, and Strong Institutions), proposing and validating a systematic approach for the automatic classification of violence severity based on the QOVS [18], which provides a mechanism for mapping patterns of urban violence. Although not the focus of this work, this capability could support public policies aimed at target 16.1 — reducing all forms of violence [8].

Other important contribution is the relation of this work with the core themes of the ADVANCE workshop by proposing a software-based solution that helps to build an evolution in ICT Service for urban dynamics. While in the previous work [15] and it was established a general architectural model ofr social sensing (CidadeSocial) focusing on general urban dynamics, this paper advances that infrastructure by proposing and implementing a specialized service layer for urban violence.

8.3 Future Works

The current work on its social sensing module used a Bayesian classifier as a base model. A natural next step is to deepen the violence detection capability by exploring more advanced techniques to improve the model's accuracy and generalization.

The next step is to use a Multimodal Model. This would allow VOS to understand not only text but also visual (images) and auditory (short audio clips) evidence. The research would also focus on developing a deep learning model for fusing textual, visual, and audio features. This would allow us to achieve what has already been proposed by the related works.

Also, it is important to recognize that the literature review in this work focused only on traditional academic research. To gain a holistic understanding of this domain, future work will include a multivocal literature review. This methodological approach will allow the expansion of the search to the grey literature, enabling

the identification and understanding of solutions from industry, such as mobile applications and other commercial approaches.

Another future path is to understand how bias mitigation and fact-checking could increase the reliability of spatial-temporal platforms, as one limitation identified was the potential for the spread of false or biased information. A future line of research would be the development of a fact-checking module. This could involve the use of knowledge graphs to cross-reference information from different reports on the same event, or the implementation of models to detect anomalies and inconsistencies in the data.

Acknowledgments

This paper used AI, in the process of English grammar correction. The Grammarly [13] platform was used to correct the text.

References

[1] Clodis Boscarioli, Renata Mendes de Araujo, Rita Suzana Maciel, Valdemar Vicente Graciano Neto, Flavio Oquendo, Elisa Yumi Nakagawa, Flavia Cristina Berrnardini, José Viterbo, Dalessandro Vianna, Carlos Bazilio Martins, Adriana Pereira Medeiros, Edwin Meza, Patrick Moratori, Carlos Alberto Malcher Bastos, Renata Mendes de Araujo, Sean Siqueira, Ig Bittencourt, Seiji Isotani, Bernardo Pereira Nunes, Cristiano Maciel, Claudia Cappelli, Vanessa Nunes, Celia G. Ralha, Luiz Sérgio P. Silva, Suzana C. B. Sampaio, Renata T. Moreira, Alexandre M. L. Vasconcelos, Rita Suzana P. Maciel, José Maria N. David, Daniela Claro, Regina Braga, Antonio Carlos Marcelino de Paula, Glauco de Figueiredo Carneiro, Isabel Cafezeiro, Leonardo Cruz da Costa, Luciana Salgado, Marcelo da Costa Rocha, Rodrigo Salvador Monteiro, Roberto Pereira, Maria Cecilia C. Baranauskas, Vinicius Carvalho Pereira, Fabio Silva Lopes, Leandro Augusto da Silva, Vivaldo José Breternitz, and Cleyton Slaviero. 2017. *I GranDSI-BR: Grand Research Challenges in Information Systems in Brazil 2016-2026.* Sociedade Brasileira de Computação. doi:10.5753/sbc.2884.0 Publication Title: Sociedade Brasileira de Computação.

[2] Obed Campos, Pablo Pancardo Garcia, and Jose Adan Hernandez Nolasco. 2022. Dynamic Fuzzy Model to Detect Verbal Violence in Real Time. *Computer Science* 23, 4 (Nov. 2022). doi:10.7494/csci.2022.23.4.4616

[3] Ana Correa, Eliel Roger, Tiago França, José Gomes, and Jonice Oliveira. 2019. CidadeSocial: An Application Software for Opportunistic and Collaborative Engagement of Urban Populations: First Workshop, BiDU 2018, Rio de Janeiro, Brazil, August 31, 2018, Revised Selected Papers. 141–155. doi:10.1007/978-3-030-11238-7_9

[4] Aline Dresch, Daniel Pacheco Lacerda, and José Antonio Valle Antunes Júnior (Junico Antunes). 2015. *Design Science Research: Método de Pesquisa para Avanço da Ciência e Tecnologia.* Bookman Editora.

[5] Tiago Cruz de França. 2019. *ANDARE: um framework para inclusão da análise de dados de mídias sociais no contexto da preparação e resposta à emergência em situações de manifestações de massa.* Tese (Doutorado). Universidade Federal do Rio de Janeiro, Rio de Janeiro. https://tinyurl.com/tmaydae4

[6] Juanpablo Heredia, Leighton Estrada-Rayme, Jeremy Matos-Cangalaya, and Jorge Poco. 2025. Interactive Exploration and Explanation of Spatio-Temporal Anomalies with Graph-LLM Integration. In *2025 38th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI).* 1–6. doi:10.1109/SIBGRAPI67909.2025.11223398

[7] Jonice Oliveira, Andressa Silva, and Raphael Franckini. 2012. *UFRJ Social - Propagação Colaborativa e Recomendação De Informações Utilizando Computação Móvel e Dados Georreferenciados.*

[8] ORGANIZAÇÃO DAS NAÇÕES UNIDAS (ONU). [n. d.]. Objetivos de Desenvolvimento Sustentável | As Nações Unidas no Brasil. https://brasil.un.org/pt-br/sdgs

[9] Siripen Pongpaichet, Boonyapat Sukosit, Chitchaya Duangtanawat, Jiramed Jamjongdamrongkit, Chancheep Mahacharoensuk, Kantapong Matangkarat, Pattadon Singhajan, Thanapon Noraset, and Suppawong Tuarob. 2024. CAMELON: A System for Crime Metadata Extraction and Spatiotemporal Visualization From Online News Articles. *IEEE Access* 12 (2024), 22778–22802. doi:10.1109/ACCESS.2024.3363879

[10] Francisco A. Pujol, Higinio Mora, and Maria Luisa Pertegal. 2020. A soft computing approach to violence detection in social media for smart cities. *Soft Computing* 24, 15 (Aug. 2020), 11007–11017. doi:10.1007/s00500-019-04310-x

[11] Python Software Foundation. [n. d.]. pickle — Python object serialization. https://docs.python.org/3/library/pickle.html

[12] Roberta de Souza. 2024. Confrontos armados causaram suspensão de aulas em 368 escolas públicas no Rio em 2024. https://oglobo.globo.com/rio/noticia/2024/06/17/confrontos-armados-causaram-suspensao-de-aulas-em-368-escolas-publicas-no-rio-em-2024.ghtml Section: Rio.

[13] Saul Sousa da Rocha, Carlos Henrique Vale e Silva, Mateus José da Silva, Jose Rodrigues Torres Neto, Carlos Henrique G. Ferreira, and Glauber Dias Gonçalves. 2025. Uso de Características Temporais e Semânticas para Detectar Eventos em Vídeos de Violência Urbana. In *Brazilian Symposium on Multimedia and the Web (WebMedia).* SBC, 464–472. doi:10.5753/webmedia.2025.16150 ISSN: 0000-0000.

[14] Ana Carolina Salgado, Claudia Lage Rebello da Motta, and Flavia Maria Santoro. 2015. *Grandes Desafios da Computação no Brasil - Relatos do 3º seminário.* Sociedade Brasileira de Computação. https://books-sol.sbc.org.br/index.php/sbc/catalog/book/27 Publication Title: Sociedade Brasileira de Computação.

[15] Eliel R. Silva, Jonice Oliveira, and Tiago Cruz de França. 2023. CidadeSocial: A social sensing model for urban dynamics. In *10th International Workshop on ADVANCEs in ICT Infrastructures and Services (ADVANCE 2023).* 12p.

[16] Eliel Roger da Silva, Tiago Cruz de França, and Jonice Oliveira. 2025. Uma arquitetura de sensoriamento social para suporte à comunicação não planejada no contexto de violência urbana no projeto CidadeSocial. In *Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia).* SBC, 13–14. doi:10.5753/webmedia_estendido.2025.16296 ISSN: 2596-1683.

[17] Matheus Henrique C. T. de Souza, Eliel Roger da Silva, Tiago Cruz de França, and Jonice Oliveira. 2025. VISAGE: Detection and automatic classification of urban violence through social media data. In *Brazilian Workshop on Social Network Analysis and Mining (BraSNAM).* SBC, 26–39. doi:10.5753/brasnam.2025.8111 ISSN: 2595-6094.

[18] Peter Tyrer, Sylvia Cooper, Elizabeth Herbert, Conor Duggan, Mike Crawford, Eileen Joyce, Deborah Rutter, Helen Seivewright, Sandra O'Sullivan, Bharti Rao, Domenic Cicchetti, and Tony Maden. 2007. The Quantification of Violence Scale: a Simple Method of Recording Significant Violence. *International Journal of Social Psychiatry* 53, 6 (Nov. 2007), 485–497. doi:10.1177/0020764007083870

[19] Dong Wang, Tarek Abdelzaher, and Lance Kaplan. 2015. Social sensing trends and applications. 13–20. doi:10.1016/B978-0-12-800867-6.00002-9

[20] Manzhu Yu, Myra Bambacus, Guido Cervone, Keith Clarke, Daniel Duffy, Qunying Huang, Jing Li, Wenwen Li, Zhenlong Li, Qian Liu, Bernd Resch, Jingchao Yang, and Chaowei Yang. 2020. Spatiotemporal event detection: a review. *International Journal of Digital Earth* 13, 12 (Dec. 2020), 1339–1365. doi:10.1080/17538947.2020.1738569 Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/17538947.2020.1738569.

[21] Mariana Zylberkan. 2025. Cerca de 4,9 milhões vivem em áreas com presença do crime organizado no Rio, diz Datafolha. https://www1.folha.uol.com.br/cotidiano/2025/11/cerca-de-49-milhoes-vivem-em-areas-com-presenca-do-crime-organizado-no-rio-diz-datafolha.shtml Section: Cotidiano.

[13] https://app.grammarly.com/

phantomsection

GISSA GPT: An Agent-Oriented Architecture for Intelligent Governance in Digital Health

Caio Leandro Rodrigues Cavalcanti
caio.leandro.rodrigues07@aluno.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Fabio José Gomes de Sousa
prof.fabiojose@gmail.com
FIOCRUZ/BA
Eusébio, CE, Brazil

Rodrigo Matos Aguiar
rodrigo.matos9@hotmail.com
IFCE
Fortaleza, CE, Brazil

César Olavo de Moura Filho
cesar.olavo2011@gmail.com
IFCE
Fortaleza, CE, Brazil

Luiz Odorico Monteiro de Andrade
odorico.monteiro@fiocruz.br
FIOCRUZ/CE
Eusébio, CE, Brazil

Antônio Mauro Barbosa de Oliveira
mauro@lar.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Abstract

The GISSA system (Intelligent Governance in Health Systems) is a computing platform, implemented in 2020, that automatically collects data from Ministry of Health information systems in Brazil: e-SUS, CNES, SIM, SINASC, SI-PNI, and SINAN. It does so through data-extraction bots, analyzes these data using multiple intelligent technologies, and transforms them into integrated information that supports decision-making at different levels of health management. In 2021, the Networks and Systems Laboratory team at IFCE (LAR) implemented "Smart GISSA", a health-governance system based on Machine Learning, defended as a doctoral dissertation at the Federal University of Ceará [9]. In 2024, the same LAR team developed "Giselle Saúde", a sentiment-detection system using Generative AI in Digital Health [16]. This paper presents GISSA GPT, an evolution of Smart GISSA that incorporates Large Language Models (LLMs), leveraging the expertise acquired by the LAR team in building Giselle Saúde. It is an agent-oriented prototype for digital health governance with the following characteristics: (i) a generative-AI environment inspired by Smart GISSA; (ii) an event-driven design for integration among modules; and (iii) Retrieval-Augmented Generation (RAG) mechanisms to anchor responses in verifiable sources, implemented with modern technologies (n8n, LangChain, ChromaDB, etc.) and able to incorporate new protocols and guidelines selected by a multidisciplinary team [13, 18]. Considering that Giselle Saúde focuses on mental health, future work proposes integrating GISSA GPT with Giselle Saúde.

Keywords

e-Health, Workflow Management, Information and Data Management, Decision Analysis and Methods, Big Data Management, Privacy protection and Privacy-by-design, Security and Trust Management, AI for services infrastructure optimization

1 Introduction

The expansion of digital health initiatives has been driven by the increasing volume of clinical and administrative data, the need to broaden access, and advances in information and communication technologies. However, in sensitive domains such as healthcare, expanding access and automating workflows is not enough. Information provided by digital systems must be traceable, auditable, and aligned with ethical and regulatory practices; otherwise, it may lead to inadequate decisions and deepen care asymmetries [19, 3, 18].

Large Language Models (LLMs), based on Transformer architectures, have shown the ability to generate coherent text, synthesize information, and interact through natural language [17, 7]. Yet, such models are not verifiable knowledge bases. Because they operate probabilistically, they can produce plausible statements that are incorrect or unsupported by evidence—a phenomenon often described as hallucination. In healthcare, responses without documentary support can lead to severe clinical and social consequences, especially in surveillance, mental health, and vulnerable populations [2, 18].

Retrieval-Augmented Generation (RAG) has been used to mitigate these limitations by combining document retrieval with controlled text generation. By grounding responses in a validated corpus, the approach promotes transparency, auditability, and governance, making it possible to trace the origin of the information used by the model [13, 20, 18].

This paper presents GISSA GPT, an agent-oriented architecture for governance and decision support in digital health, conceived as an evolution of Smart-GISSA and based on recent experience with Giselle Saúde, a platform for older adults' mental health that uses Generative AI for sentiment detection and analysis [9, 16]. It comprises a Sentiment Detector (SD) and a Generative Virtual Assistant (GVA) to analyze sentiments expressed in interactions between users and health professionals, identifying older adults who may require professional evaluation for specialized care based on urgency, resulting in in-person or remote consultations [16].

The proposal seeks to address pain points observed in surveillance and management environments, such as: (i) low transparency regarding the origin of information; (ii) difficulty integrating services and institutional documents; and (iii) lack of natural-language explanations across multiple heterogeneous sources. GISSA GPT is presented both as an architectural reference and as a reproducible design protocol, including limitations and mitigation mechanisms.

The scope includes decision support and governance based on indicators, alerts, and institutional documents, with traceability

and an audit trail; it includes conversational interaction with explicit sources and controlled retrieval over repositories validated by technical and health teams. It excludes automated diagnosis, therapeutic prescription, and individualized guidance in urgent situations, which must be handled by licensed professionals [18].

2 Smart GISSA

The GISSA platform (Intelligent Governance in Health Services) provides the foundation on which Smart-GISSA was built. Its architecture was designed to overcome data fragmentation by acting as a large integrator of health information [15, 9].

The architecture can be seen in Figure 1:

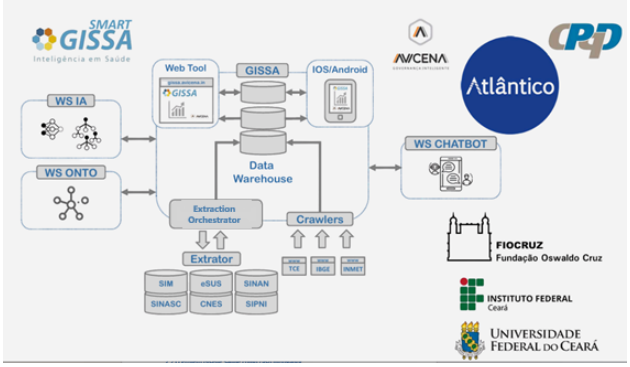

Figure 1: High-level view of the Smart-GISSA.

Smart-GISSA represents the evolution of the GISSA platform by incorporating an intelligence layer that transforms the system from a data repository into a predictive analytics tool [9]. The new architecture expands the capabilities of the original system, preserving the solid data-integration base while adding new specialized web services (WS):

- **WS AI (Artificial Intelligence):** a microservice that encapsulates trained Machine Learning models, returning predictions, classifications, and risk analyses (e.g., probability of maternal/infant death or epidemic trends) [9].
- **WS ONTO (Ontology):** organizes domain knowledge to make sense of heterogeneous data, with semantic relationships and inferences (e.g., diseases, symptoms, risk factors), contributing to semantic interoperability [11, 12].
- **WS CHATBOT:** a service that enables natural-language questions and contextualized answers based on the platform's data and analyses [14].

The Smart-GISSA architecture is conceived as a layered model, following the data value chain: from capture at primary sources, through integration and storage in the Data Warehouse, to the application of AI models and the delivery of results to end users through multiple interfaces. As shown in Figure 2, this modular, microservice-based approach provides greater flexibility, scalability, and ease of maintenance, enabling new functionalities and models to be added independently [9, 4].

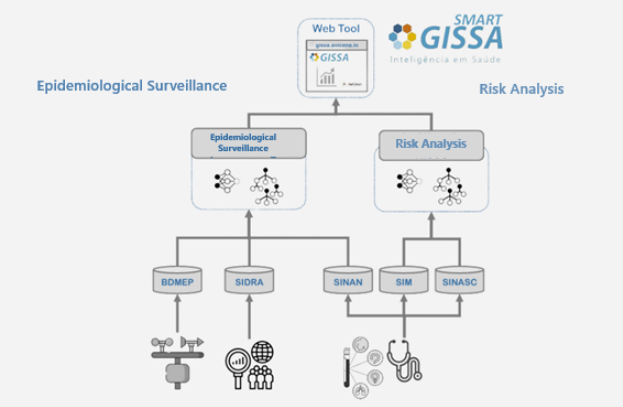

Figure 2: Layered Smart-GISSA architecture with specialized services.

Among the services implemented in Smart-GISSA, the following stand out:

- **Data Mining for Risk of Death (DMRisD):** Maternal and infant deaths are tragedies, many of which could be prevented with timely intervention. The challenge is to identify high-risk pregnant women and newborns as early as possible, within a critical time window. The DMRisD approach was designed to support risk stratification and prioritization in surveillance and care pathways [9].
- **Data Mining for Epidemics (DMEpi):** Arboviruses such as Dengue, Zika, and Chikungunya represent an ongoing threat to public health. Predictive models can help anticipate spatial and temporal risk, complementing traditional surveillance [6, 8].

This technological and institutional trajectory forms the basis on which GISSA GPT is positioned.

3 Theoretical background

3.1 Digital health and virtual assistants

Recent literature characterizes digital health as an umbrella field that integrates *eHealth*, *mHealth*, telemedicine, connected devices, and intelligent systems aimed at organizing care and enabling communication between professionals and users [19, 3]. In this context, virtual assistants have been used for triage, health education, and decision support across different domains, potentially improving access, standardization, and continuity of care [10, 14].

However, the same attributes that enable personalization and real-time interaction also intensify concerns about privacy, bias, algorithmic opacity, and technological dependence—especially when Generative AI is used in clinical domains [4, 2, 3]. In such cases, traceability and explainability become requirements for governance and safety, allowing reconstruction of the informational path that supports a given answer [13, 19].

The challenge becomes even more pronounced in mental health, where communication involves emotional, cultural, and linguistic nuances. In the Giselle Project, the generative assistant is described as capable of producing responses that combine contextual adequacy with the user's emotional and cultural state. Even so, the

project itself emphasizes that such innovations require clinical studies to assess efficacy, effectiveness, and safety, as well as explicit responsible-use policies [16].

In this scenario, GISSA GPT aims to advance the governance axis. Its distinguishing feature is not merely the use of LLMs, but the integration of specialized agents, RAG, and mechanisms for observability, auditability, and documentary traceability [13, 20]. By grounding answers in controlled and verifiable repositories validated by technical and health teams, GISSA GPT treats virtual assistants as infrastructure components for decision support, rather than generic conversational systems [19]. Thus, the proposal shifts the focus from access and interaction to transparency, verifiability, and care governance—core aspects in regulated, high-risk environments [19, 3].

3.2 The Giselle Saúde Project as a motivating case

The Giselle Saúde Project is presented as a platform focused on older adults' mental health, composed of a Sentiment Detector and a Generative Virtual Assistant [16]. Its operational goal is to identify, through conversational interactions, signals indicating the need for professional evaluation, referring users to in-person or remote care according to urgency and priority criteria, thereby integrating technology and human care [16].

The paper describing the Giselle architecture details a model in which the *prompt* is co-designed by health professionals and implemented by a technical team, guiding the dialog flow and handling sensitive cases. The design explicitly highlights the need for safety mechanisms, human oversight, and informed consent, and shows that mental health recommendations require traceability, justification, and accountability [19].

As a motivating case, Giselle exposes a gap that goes beyond mental health: generative conversational assistants are not, by themselves, auditable decision-support systems. Coupling an LLM to a chatbot enables natural interaction, but it does not satisfy requirements such as audit trails, information provenance, integration across heterogeneous sources, governance, and explainability—which are essential in regulated domains [2, 4]. In scientific and technological terms, this translates into open challenges such as:

- the absence of verifiable sources in generated answers;
- the difficulty of reconstructing reasoning and data provenance;
- the lack of integration with epidemiological indicators and institutional documents;
- the absence of governance and policy layers for safe use;
- the need for human escalation mechanisms;
- and the demand for reproducible protocols to evaluate efficacy, effectiveness, and safety.

Such limitations are generalizable and appear in recent literature discussing *clinical copilots* and decision-support systems based on Generative AI, which face barriers related to auditability, institutional integration, and regulatory robustness [2, 4].

In this sense, GISSA GPT is proposed as an architectural advance oriented toward governance, integrating specialized agents, RAG components, and documentary traceability to support decision-making in surveillance and health management environments [13, 20]. The architecture results not only from applying LLMs, but from internalizing the gap observed in Giselle: conversational assistants must operate as institutional infrastructure, not merely as natural-language interfaces [19].

3.3 Capabilities and limitations of LLMs for digital health

Large Language Models (LLMs) based on Transformer architectures, such as ChatGPT, have expanded the frontier of digital health applications by enabling information synthesis, document summarization, guideline translation, and natural-language interaction [17, 7, 4]. Functionally, these models act as mediators between users and systems, converting indicators, clinical records, and administrative workflows into narratives that are easier to understand, potentially improving communication, health education, and decision support in institutional contexts [4, 19].

These capabilities are especially relevant in domains characterized by diverse documents, formats, and ontologies—a typical case in healthcare, where clinical guidelines, epidemiological data, administrative protocols, and care histories coexist as heterogeneous sources [3, 12]. In such scenarios, conversational mediation by LLMs can reduce linguistic and cognitive barriers while enabling integration across previously fragmented systems. For mental-health assistants, as discussed in the previous subsection, this mediation includes emotional, cultural, and linguistic nuance.

Nevertheless, adopting LLMs in digital health involves substantive challenges. Because of their probabilistic nature, these models do not operate as verifiable knowledge bases and can generate plausible statements that are incorrect or unsupported by documentary evidence. In regulated domains, answers without traceable evidence can undermine clinical safety, institutional trust, and continuity of care, amplifying ethical and social risks for vulnerable populations [2].

Recent literature thus identifies a set of limitations that are particularly relevant for digital health:

- (i) lack of traceability—difficulty identifying which sources and reasoning support the answer;
- (ii) low auditability—inability to reconstruct the informational path for oversight and institutional responsibility;
- (iii) limited integration with legacy systems—need for interoperability with protocols, indicators, and official documents;
- (iv) algorithmic opacity—difficulty explaining and interpreting outputs in clinical decisions;
- (v) insufficient governance—lack of explicit policies for responsible use in regulated environments.

In light of these limitations, the literature suggests complementing LLMs with additional verification, human oversight, and documentary-integration mechanisms, such as RAG, specialized agents, audit trails, and informed consent [13, 20, 19]. These mechanisms shift conversational assistants from an access/interaction axis to a transparency/governance axis, which is essential in public systems and surveillance/management environments that require *accountability*, safety, and institutional alignment [19, 3].

Accordingly, the Giselle case discussed earlier highlights a technological gap that motivates this work: LLMs can generate dialog, but they do not guarantee governance, and building digital health

systems based on Generative AI requires architectures that incorporate traceability, explainability, and integration with institutional workflows—requirements addressed by the GISSA GPT proposal [19].

4 Related Work

The advance of digital technologies applied to healthcare has produced a diverse ecosystem of solutions aimed at epidemiological surveillance, decision support, automated triage, health education, and service management. These solutions can be grouped into three main streams in recent literature: (i) health data monitoring and analysis systems; (ii) conversational assistants and intelligent agents for healthcare; and (iii) Generative AI architectures and Retrieval-Augmented Generation (RAG) techniques.

(i) Health data monitoring and analysis systems. Platforms such as NSSP/BioSense illustrate the centrality of institutional infrastructures for data collection and analysis, combining data integration, dashboards, and alerts to support decisions and public health planning [8]. Other contemporary initiatives, such as *Outbreaks Near Me*, have also been described as mechanisms for monitoring and risk communication in public health [5]. In Brazil, discussions on standardization and interoperability indicate that sustainable integration requires well-defined models and processes, especially when consolidating heterogeneous repositories and rationalizing administrative data [1, 3]. In this context, GISSA and Smart-GISSA align with approaches that aim to unify sources and enable distributed, multipurpose analyses while maintaining governance and consistency over data use [9, 15].

(ii) Conversational assistants and intelligent agents in healthcare. Conversational assistants have been investigated as technologies for access, health education, and support across different domains, including mental health [14, 10]. The literature also reports limitations related to traceability, explainability, and risk mitigation, particularly when conversational systems operate without explicit integration to governed repositories and without formal accountability mechanisms [18]. The Giselle Saúde project contributes to this stream by combining Generative AI with sentiment detection/analysis and human supervision, describing requirements tied to consent, safety, and cultural adequacy for vulnerable populations [16].

(iii) Generative AI, RAG, and governance. Large Language Models (LLMs) have been explored for tasks such as information synthesis, result explanation, and mediation of natural-language queries [4]. However, a recurring concern is that LLMs are not verifiable knowledge bases and may produce plausible but incorrect answers, complicating auditing when there is no grounding in evidence [2, 18]. Retrieval-Augmented Generation (RAG) techniques have emerged as an alternative to ground answers in traceable sources by combining retrieval of relevant excerpts from controlled corpora with conditioned text generation [13, 20]. In digital health, this arrangement is often considered relevant for transparency and governance, although there is still limited systematization of RAG and observability practices for institutional decision-making environments [19, 18].

Convergence and gaps. The literature offers relevant contributions but still presents gaps for the problem addressed in this paper:

(1) **Governance and auditing.** Few works treat LLMs as auditable components embedded in formal health-management workflows, with recorded context to support decisions [18].
(2) **Operational integration.** Conversational systems often do not integrate, within a single *pipeline*, epidemiological indicators, analytical/predictive models, and institutional normative documents [19].
(3) **Institutional explainability.** Part of the literature discusses clinical or algorithmic explanations; there is less emphasis on explanations aimed at managers and oversight bodies, with explicit traceability and provenance [18].
(4) **Agent-based orchestration.** The combination of specialized agents, RAG, LLMs, and observability mechanisms is still rarely described as a structured arrangement for digital health [13, 20].

GISSA GPT positions itself in this space by proposing an architecture that integrates governed data (GISSA), analytical services (Smart-GISSA), and institutional documents; offers natural-language conversational interaction mediated by agents; incorporates RAG techniques to anchor answers in traceable sources; and includes observability and auditing mechanisms to support decision-making in digital health [19, 18, 13].

5 GISSA GPT Architecture

The GISSA GPT architecture was designed to reuse the existing infrastructure of GISSA and Smart-GISSA while incorporating Large Language Models (LLMs) as a conversational interface module and Retrieval-Augmented Generation (RAG) mechanisms organized around specialized agents [13, 20]. Its central goal is to support governance and decision-making in digital health while preserving traceability, auditability, and alignment with institutional workflows [19, 3].

Structurally, GISSA GPT adopts a microservice- and event-oriented approach, in which relatively independent modules communicate through queues, *webhooks*, and APIs. The architecture is organized into functional modules: (i) data ingestion and governance; (ii) analytics and risk services; (iii) the sub-symbolic module (LLM and RAG); (iv) agent orchestration; (v) natural-language interaction; and (vi) observability and auditing [4, 13].

5.1 Overview

At a high level, GISSA GPT acts as a conversational governance layer over the GISSA/Smart-GISSA ecosystem:

- it receives natural-language questions and commands from managers and health professionals;
- it identifies intent and task type (indicator query, protocol explanation, summary generation, risk analysis, etc.);
- it triggers specialized agents that query GISSA indicators, Smart-GISSA predictive models, and institutional documents;
- it uses RAG to ground the answer in verifiable sources;
- it returns an explained response with explicit references to the origin of data, an audit trail, and the option to record the interaction for governance purposes.

The architecture is agnostic to the LLM provider (it can consume different models via APIs), but it requires that every generated answer be linked to a set of retrieved evidence and that the interaction be logged with sufficient metadata for later auditing [19, 13].

5.2 Functional modules

5.2.1 Data and governance module (GISSA). The first module reuses GISSA as the primary data source:

- extraction bots collect data from the Ministry of Health systems (e-SUS, CNES, SIM, SINASC, SI-PNI, SINAN, among others);
- the data are integrated into an analytical repository (*data lake*/*data warehouse*), with standardized dictionaries, schema versioning, and access policies;
- data quality, anonymization/pseudonymization, and aggregation rules are applied according to current digital-health norms and data-protection requirements.

This module ensures that any query handled through GISSA GPT is supported by governed, documented, and versioned data [19, 3]. When applicable, standardization may adopt semantic interoperability references and electronic health record modeling approaches (e.g., openEHR/ADL archetypes) to maintain consistency over time and across services [1, 12].

5.2.2 Symbolic module (Smart-GISSA). The second module corresponds to the Machine Learning services developed within Smart-GISSA:

- predictive models and risk classifiers;
- population stratification services;
- anomaly detection or identification of relevant epidemiological patterns.

These services are exposed as independent APIs and can be invoked by GISSA GPT agents to compose natural-language answers [9]. Thus, the LLM does not generate predictions autonomously; it orchestrates and explains results produced by validated models [2, 19].

5.2.3 Sub-symbolic module (LLM and RAG). The third module organizes the documentary knowledge relevant for health governance:

- clinical protocols and national guidelines;
- resolutions, ordinances, and technical notes;
- internal manuals, administrative flows, and GISSA documents;
- institutionally validated reports and *dashboards*.

Documents are processed in an indexing *pipeline* (for example, using LangChain + ChromaDB or equivalent), with chunking, metadata enrichment (source, date, version, document type), and creation of vector and symbolic indices [13, 20]. During interaction, the RAG module:

(1) receives the natural-language query;
(2) generates a vector representation of the question;
(3) retrieves the most relevant passages (documents, protocols, notes);
(4) provides this context to the LLM, which must cite or explicitly indicate the sources used.

5.2.4 Agent orchestration module. The fourth module hosts intelligent agents responsible for specific tasks. Examples:

- **Epidemiology Agent** — queries GISSA indicators, applies filters (time, territory, age range), and produces analytical summaries;
- **Data Quality Agent** — checks integrity, completeness, and consistency of requested indicators;
- **Governance/Compliance Agent** — verifies whether the answer involves protocols, norms, or regulatory risk and injects additional explanations;
- **Explanation Agent** — turns numeric and technical outputs into narratives understandable to non-specialist managers.

Orchestration is performed by an *agent manager* and a workflow orchestrator (e.g., n8n), following an event-driven model: the arrival of a new question, alert, or data update generates events that may trigger one or more agents. Results are consolidated and sent to the natural-language interaction module [4, 13].

5.2.5 Natural-language interaction module. The fifth module is the interface between users and the GISSA GPT ecosystem:

- access channels (web interface, institutional chatbot, integration with existing systems);
- conversation management module (session, contextual history, turn control);
- LLM *gateway* (responsible for sending *prompts* enriched with RAG context + agent outputs and receiving responses).

This module implements policies for:

- scope control (types of questions that can be answered);
- response filtering (blocking attempts at individualized diagnosis or prescription);
- explicit limitations (warnings about decision-support nature and the need for professional validation).

5.2.6 Observability and auditing module. The sixth module concentrates observability and governance mechanisms:

- interaction *logs* (query, retrieved context, triggered agents, used sources, model versions);
- usage *dashboards* (what types of questions are asked, by which user profiles, with which sources);
- audit trails for safety, quality, and impact assessment;
- alerts for anomalous behavior (e.g., misuse outside scope).

This module is essential to treat GISSA GPT as institutional infrastructure, enabling retrospective inspection of decisions, workflow review, and policy adjustment [19, 3].

5.3 Interaction flow

In simplified terms, a typical interaction flow is as follows:

(1) **Input** — a manager or health professional formulates a natural-language question (e.g., "Which neighborhoods show the highest risk of dengue outbreaks next month?").
(2) **Classification** — the system identifies the task type (indicator query + prediction + risk explanation).
(3) **Agent orchestration** — the orchestrator triggers:
   - the Epidemiology Agent, which queries GISSA and Smart-GISSA models;
   - the Data Quality Agent, which assesses indicator reliability;
   - the RAG module, which retrieves relevant protocols and documents.

(4) **LLM synthesis** — the LLM receives agent outputs and retrieved document passages and generates a natural-language response, including:
   - an explanation of risk factors;
   - source and date indications;
   - interpretive cautions when applicable.

(5) **Output and logging** — the response is delivered to the user, accompanied by references and, optionally, a formal record in the auditing module, including the context used to support the decision.

With this design, the LLM does not replace GISSA or Smart-GISSA; it acts as a mediation and explanation layer reinforced by specialized agents and RAG [13, 19].

Figure 3 represents the conceptual GISSA GPT architecture as a functional chain that integrates conversational interaction, orchestration of specialized agents, RAG mechanisms, an LLM-based generative module, and auditing and governance modules [13, 20]. At the entry point, the user (a manager or health professional) interacts with a conversational interface (web/chatbot), whose utterance is forwarded to a conversation manager and task classifier responsible for identifying the interaction goal. This module coordinates queries to GISSA/Smart-GISSA APIs, enabling access to epidemiological indicators, analytical models, and other preexisting resources.

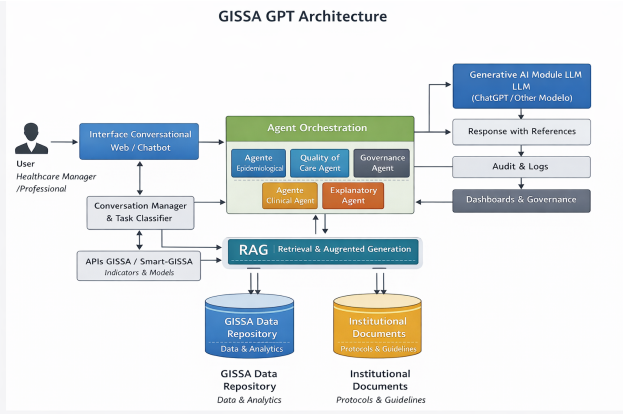

Figure 3: Conceptual GISSA GPT architecture.

Structured information is then routed to the agent orchestration module, which aggregates epidemiology, data quality, governance, and explanation agents. These agents complement processing with checks, repository queries, contextualization, and justifications.

The RAG module queries two categories of sources: (i) the GISSA data repository, containing data and analyses; and (ii) institutional documents, such as guidelines, protocols, and norms. The consolidated result is forwarded to the generative LLM, which produces the final answer associated with references or retrieved documentary excerpts [13, 20].

The output is submitted to auditing, *logging*, and traceability modules, enabling retrospective inspection, and it can feed *dashboards* and governance mechanisms oriented to decision support and digital surveillance [19, 3].

6 Concluding remarks

This work presented *GISSA GPT*, an agent-oriented architecture for governance and decision support in digital health, built on the GISSA/Smart-GISSA ecosystem and inspired by lessons from the Giselle Saúde project [9, 16]. The proposal starts from the diagnosis that LLMs, while useful as a conversational mediation layer, are not sufficient to sustain decisions in regulated domains without additional modules for governance, traceability, and institutional integration [18].

Conceptually, GISSA GPT contributes by treating virtual assistants not as isolated interfaces, but as components of an institutional decision-support infrastructure coupled to governed data, validated predictive models, and normative documents [19, 18]. The combination of a data module (GISSA), analytical services (Smart-GISSA), RAG, specialized agents, and observability mechanisms forms an architectural arrangement that addresses gaps identified in the literature, such as lack of audit trails, difficulty of operational integration, and low explainability for managers and oversight bodies [18, 13].

Technologically, the proposed architecture describes how LLMs can be repositioned from autonomous answer generators to orchestrators of evidence and explanations grounded in verifiable sources. By requiring that each response be anchored in institutionally recognized indicators, models, and documents, GISSA GPT shifts the value axis of natural-language interaction toward the ability to support governance processes in digital health [13, 18].

This work is predominantly architectural and exploratory. It does not yet provide large-scale clinical or operational validation, nor systematic metrics of impact on decision quality, response time, or user trust. These limitations motivate the agenda for future work.

As follow-ups, four main directions stand out:

(1) prototyping and institutional pilots, with controlled deployment in surveillance and management environments, to evaluate performance, usability, acceptability, and effects on decision workflows;
(2) integration with Giselle Saúde, exploring scenarios in which GISSA GPT acts as a governance and explanation module for aggregated cases, while Giselle remains focused on individual mental-health interaction with clinical safeguards [16];
(3) a formal governance evaluation, including metrics of traceability, auditability, adherence to norms, and perceived risk by managers, health professionals, and oversight bodies [18];
(4) generalization of the architecture to other digital health domains and other public administration sectors, where LLMs can be combined with governed data and institutional rules [19].

In summary, GISSA GPT does not aim to replace health professionals or automate clinical decisions, but to provide an architectural reference for incorporating Generative AI responsibly into digital health infrastructure [18]. By articulating data, models, documents, and agents under a governance logic, this work seeks to contribute to anchoring the use of LLMs in public policy in requirements of transparency, accountability, and public interest [18, 19].

References

[1] Tiago Veloso Araujo, Silvio Ricardo Pires, and Paulo Bandiera-Paiva. 2014. Adoção de padrões para registro eletrônico em saúde no brasil. *RECIIS – Revista Eletrônica de Comunicação, Informação & Inovação em Saúde*, 8, 4. Retrieved Jan. 17, 2026 from https://www.reciis.icict.fiocruz.br/index.php/reciis/article/view/440.

[2] John W. Ayers, Adam Poliak, Mark Dredze, et al. 2023. Comparing physician and artificial intelligence chatbot responses to patient questions posted to a public social media forum. *JAMA Internal Medicine*, 183, 6, 589–596. doi:10.1001/jamainternmed.2023.1838.

[3] Ivana Cristina de Holanda Cunha Barreto, Kelen Gomes Ribeiro, and Luiz Odorico Monteiro de Andrade, eds. 2024. *Saúde Digital: Conceitos, Pesquisas e Desenvolvimento Tecnológico*. Editorial Casa, Curitiba, PR, Brazil. ISBN: 978-65-5216-261-8. doi:10.70271/250505.1056.

[4] Rishi Bommasani et al. 2021. On the opportunities and risks of foundation models. *arXiv*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2108.07258 arXiv: 2108.07258.

[5] Boston Children's Hospital (HealthMap). 2026. Outbreaks near me. Crowdsourced participatory surveillance for influenza and COVID-19. Retrieved Jan. 18, 2026 from https://outbreaksnearme.org/.

[6] Brasil. Ministério da Saúde. 2017. Monitoramento dos casos de dengue, febre de chikungunya e febre pelo vírus zika até a semana epidemiológica 35.

[7] Tom B. Brown et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2005.14165.

[8] Centers for Disease Control and Prevention. 2025. National syndromic surveillance program (NSSP) and the BioSense platform. Retrieved Jan. 17, 2026 from https://www.cdc.gov/nssp/php/about/about-nssp-and-the-biosense-platform.html.

[9] Raimundo Valter Costa Filho. 2021. *Smart-GISSA: um Sistema para Governança em Saúde Digital Baseado em Aprendizado de Máquina*. PhD thesis. Universidade Federal do Ceará, Fortaleza, CE, Brazil. Retrieved Jan. 17, 2026 from http://repositorio.ufc.br/handle/riufc/60257. Ph.D. dissertation.

[10] Kathleen K. Fitzpatrick, Alison Darcy, and Molly Vierhile. 2017. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): a randomized controlled trial. *JMIR Mental Health*, 4, 2, e19. doi:10.2196/mental.7785.

[11] Sebastian Garde, Petra Knaup, Evelyn J. S. Hovenga, and Sam Heard. 2007. Towards semantic interoperability for electronic health records: domain knowledge governance for openehr archetypes. *Methods of Information in Medicine*, 46, 3, 332–343.

[12] Dipak Kalra, Thomas Beale, and Sam Heard. 2005. The openehr foundation. *Studies in Health Technology and Informatics*, 115, 153–173.

[13] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Veselin Stoyanov. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2005.11401 arXiv: 2005.11401.

[14] Adam S. Miner et al. 2016. Smartphone-based conversational agents and responses to questions about mental health, interpersonal violence, and physical health. *JAMA Internal Medicine*, 176, 5, 619–625. doi:10.1001/jamainternmed.2016.0400.

[15] A. M. B. Oliveira et al. 2021. Lariisa: soluções digitais inteligentes para apoio à saúde pública. *Ciência & Saúde Coletiva*, 26, 5369–5378. doi:10.1590/1413-81232021265.03382021.

[16] F. J. G. Sousa et al. 2024. Giselle, uma plataforma que analisa sentimentos da pessoa idosa, apoiada por inteligência artificial generativa. In *Saúde Digital: Conceitos, Pesquisas e Desenvolvimento Tecnológico*. (1st ed.). Vol. 1. Ivana Cristina de Holanda Cunha Barreto, Kelen Gomes Ribeiro, and Luiz Odorico Monteiro de Andrade, editors. Editorial Casa, Curitiba, PR, Brazil, 174–194.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/1706.03762.

[18] World Health Organization. 2021. *Ethics and Governance of Artificial Intelligence for Health*. World Health Organization, Geneva. Retrieved Jan. 17, 2026 from https://www.who.int/publications/i/item/9789240029200.

[19] World Health Organization. 2021. *Global Strategy on Digital Health 2020–2025*. World Health Organization, Geneva. Retrieved Jan. 17, 2026 from https://www.who.int/publications/i/item/9789240020924.

[20] F. Ye, S. Li, Y. Zhang, and L. Chen. 2024. R^2AG: incorporating retrieval information into retrieval augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 11584–11596. Retrieved Jan. 17, 2026 from https://aclanthology.org/2024.findings-emnlp.678/.

phantomsection

Analysis of ZKPs-based approaches of Multi-party blockchain-based genomic data sharing.

Huyen-Trang Le
huyen.le@etud.uni-evry.com
Université Paris-Saclay, Univ Evry, IBISC, France
Évry-Courcouronnes, Ile-de-France
France

Adnan Imeri
adnan.imeri@list.lu
Luxembourg Institute of Science and Technology (LIST)/Université Paris-Saclay, Univ Evry, IBISC, France
Évry-Courcouronnes, Ile-de-France
France

Nazim Agoulmine
nazim.agoulmine@univ-evry.fr
Université Paris-Saclay, Univ Evry, IBISC, France
Évry-Courcouronnes, Ile-de-France
France

Abstract

The secure, privacy-preserving sharing of genomic data across multiple institutions is a critical enabler for precision medicine, yet it remains fundamentally constrained by the identifiability and immutability of genomic data. While blockchain technologies have been proposed to provide decentralized governance, auditability, and tamper resistance for genomic data sharing, blockchain-only solutions are insufficient because they expose transaction metadata, access patterns, and smart-contract logic, leaving significant privacy risks unresolved. Zero-Knowledge Proofs (ZKPs) have recently emerged as a key cryptographic primitive for addressing such limitations, enabling verifiable access control, policy compliance, and computation correctness without disclosing sensitive genomic data. Although several surveys examine ZKPs or blockchain in isolation or across heterogeneous application domains, there is currently no dedicated survey that systematically analyzes their combined use in multi-party blockchain-based genomic data sharing systems. This paper addresses this gap by presenting a comprehensive, domain-specific survey of ZKP-enabled blockchain architectures for genomic data sharing. We classify existing approaches by architectural models, ZKP techniques, governance mechanisms, and threat-mitigation capabilities, and then compare their assumptions, performance characteristics, and deployment maturity. Furthermore, we identify open challenges in scalability, interoperability, proof overhead, and regulatory compliance, and outline future research directions for secure, scalable, and ethically compliant genomic data-sharing ecosystems.

CCS Concepts

• **Security and privacy → Information-theoretic techniques**; **Privacy protections**; **Data anonymization and sanitization**; • **Social and professional topics → Genetic information**.

Keywords

Blockchain, Zero-knowledge proof, Privacy-preserving, Genomic data privacy

1 Introduction

1.1 Context and Inspiration

The rapid advancement of genomic sequencing technologies has fundamentally transformed biomedical research and precision medicine by enabling large-scale analysis of genetic variation across individuals and populations [6, 24]. Collaborative genomic data sharing among hospitals, research laboratories, and pharmaceutical stakeholders has become essential for accelerating disease discovery, improving diagnostic accuracy, and supporting personalized therapeutic strategies [9, 22]. However, genomic data are uniquely sensitive: a single genome is inherently identifiable, it remains fundamentally unchanged (but not entirely) over an individual's life, and it encodes hereditary information that may implicate biological relatives. Once disclosed or misused, the associated privacy risks are irreversible, making secure and trustworthy genomic data sharing a critical challenge [15, 23].

Despite the need for multi-party collaboration, most existing genomic data-sharing infrastructures rely on centralized repositories or trusted intermediaries [14]. All of these architectures are subject to external single points of failure threats, amplify insider threats, and limit transparency and accountability across institutional boundaries. To address these limitations, blockchain technologies have been proposed as decentralized infrastructures that provide immutability, auditability, and tamper-evident governance for genomic data sharing [1, 5]. These properties are particularly attractive in cross-organizational biomedical ecosystems where trust assumptions are distributed.

However, blockchain alone is insufficient to meet the stringent privacy requirements of genomic data sharing. The transparency of distributed ledgers exposes transaction metadata, access patterns, and smart-contract execution traces, which may enable inference attacks or indirect donor re-identification even when raw genomic data are stored off-chain [8, 25]. Moreover, smart contracts are executed deterministically and publicly, potentially revealing authorization logic, policy parameters, or usage patterns that can be exploited by adversaries [17, 26]. These limitations have been widely acknowledged, yet many blockchain-based genomic platforms continue to rely on coarse-grained access control or implicit trust assumptions that fall short of providing strong privacy guarantees [20].

On the other hand, Zero-Knowledge Proofs (ZKPs) have emerged as a powerful cryptographic primitive that reconciles blockchain transparency with the confidentiality demands of genomic data. ZKPs allow a prover to demonstrate the validity of a statement—such as authorization, policy compliance, or correctness of computation—without revealing the underlying genomic data [11, 12]. When integrated with blockchain and smart contracts, ZKPs enable verifiable access control, privacy-preserving governance, and auditable compliance, while preventing disclosure of sensitive genomic attributes or analytical behavior [19]. This capability is particularly

relevant for multi-party genomic data sharing environments involving heterogeneous stakeholders, regulatory constraints, and asymmetric trust relationships.

Although blockchain and ZKP technologies have both been studied extensively, existing surveys typically analyze them either in isolation or across broad cross-domain application areas such as finance, supply chains, or digital identity systems [13, 28]. In the genomic domain, prior reviews mainly focus on blockchain-based consent management, access logging, or secure storage, often treating ZKPs as a future enhancement rather than a core architectural component [3, 20]. As a result, there is currently no dedicated survey that systematically examines how ZKPs are integrated into blockchain-based genomic data sharing systems, nor how these integrations address domain-specific threats such as donor re-identification, unauthorized genomic inference, and transitive privacy risks affecting genetic relatives.

This paper addresses this gap by presenting a comprehensive, domain-specific survey of Zero-Knowledge Proof–enabled blockchain-based genomic data sharing (ZBGDS) systems. The survey is structured around architectural models, cryptographic mechanisms, governance strategies, and threat-mitigation capabilities specific to genomic data ecosystems.

Contributions. Our main contributions are summarized as follows:

(1) **Domain-specific synthesis:** that presents the first structured survey dedicated exclusively to ZBGDS, offering a focused analysis of architectural, governance, and operational requirements that distinguish GD-sharing from conventional BC applications.
(2) **Concept-to-model progression:** that provides an integrated account of ZKP techniques in ZBGDS, tracing the progression from foundational principles to system models. This includes detailed coverage of threat mitigation mechanisms, performance–privacy trade-offs, and emerging opportunities for secure genomic computation.
(3) **Real-world and forward-looking insights:** By incorporating recent ZBGDS prototypes and genomic BC platforms, this part identifies practical deployment constraints and extracts application-driven insights. In addition, it outlines key open challenges and proposes future context-sensitive research directions to guide the next generation of privacy-preserving GD infrastructures.

From a regulatory and ethical perspective, the surveyed approaches are examined in the context of data protection and governance requirements, including consent management, auditability, and compliance with regulations such as the General Data Protection Regulation (GDPR). In particular, the tension between blockchain immutability and data subject rights, as well as the role of cryptographic techniques in supporting privacy-by-design principles, is considered throughout the analysis.

The remainder of this paper is organized as follows. Section 2 introduces background concepts and threat models relevant to ZBGDS. Section 3 reviews and compares ZKP-based architectures and mechanisms for blockchain-enabled genomic data sharing. Section 4 discusses open challenges and future research directions, and Section 5 concludes the paper.

1.2 Analysis Scope and Research Methodology

Recent surveys [13, 28] typically provide multi-domain, high-level conceptual discussions on the integration of ZKPs in BC-based technologies, including financial transactions, IoT security, and supply chain assurance. In contrast, other literature primarily emphasizes secure BC implementation logic and SC access-governance policies, without ZKP integration, to prevent data leakage and threats. The discussions focus on isolated cryptographic solutions or suggest ZKPs for future work [3, 20]. This gap underscores the need for a state-of-the-art analysis of the intersection of ZKPs, BC governance, and multi-party GD sharing.

To address these limitations, our study particularly focuses on ZKPs within BC-based GD-sharing (ZBGDS) ecosystems. Figure 1 examines the full spectrum of ZBGDS components covered in this survey, including the challenges of the existing system, mitigation of threats, technological advances, and future research directions. Our review purpose is not merely to summarize existing ZBGDS ecosystems, but more crucially, to develop a coherent analytical perspective on how privacy-preserving ZKP techniques are incorporated into a decentralized genomic governance context. The intersection of blockchain, cryptography, and genomics spans multiple disciplines, and our analysis is guided by three main questions about current research, with a technical and architectural focus.

- In what ways are ZKP mechanisms integrated into BC-based systems designed for genomic data sharing?
- What advantages of privacy, security, and governance threat handling are addressed by ZKP-enabled blockchain systems?
- What technical limitations, performance trade-offs, and deployment challenges are reported in existing ZKP-based genomic data-sharing frameworks?

Researches are included in our study if they proposed genomic-focused blockchain systems with ZKPs techniques, and discussed access control governance aspects relevant to sensitive data sharing. The exclusion criteria for our analysis focused solely on blockchain infrastructure and excluded ZKPs in domains unrelated to genome data governance, and consisted solely of abstracts or non-scientific content. The selection process used a multi-stage screening process: title and abstract assessment to determine thematic relevance, followed by full-text evaluation to ensure alignment with the research questions and eligibility criteria. Although the review aims to provide focused and technically grounded coverage, it remains subject to inherent limitations, including terminological variation across disciplines, the continuous evolution of cryptographic techniques, and the limited public availability of experimental or prototype implementations.

2 Background, Preliminaries and Threat-Resilience Capabilities in ZBGDS

ZBGDS ecosystems combine decentralized governance with advanced cryptographic techniques to mitigate threats in GD. This section introduces the key principles underlying BC with ZKPs (Section 2.1) and explains how their integration contributes to threat resilience in multi-party data-sharing workflows (Section 2.2).

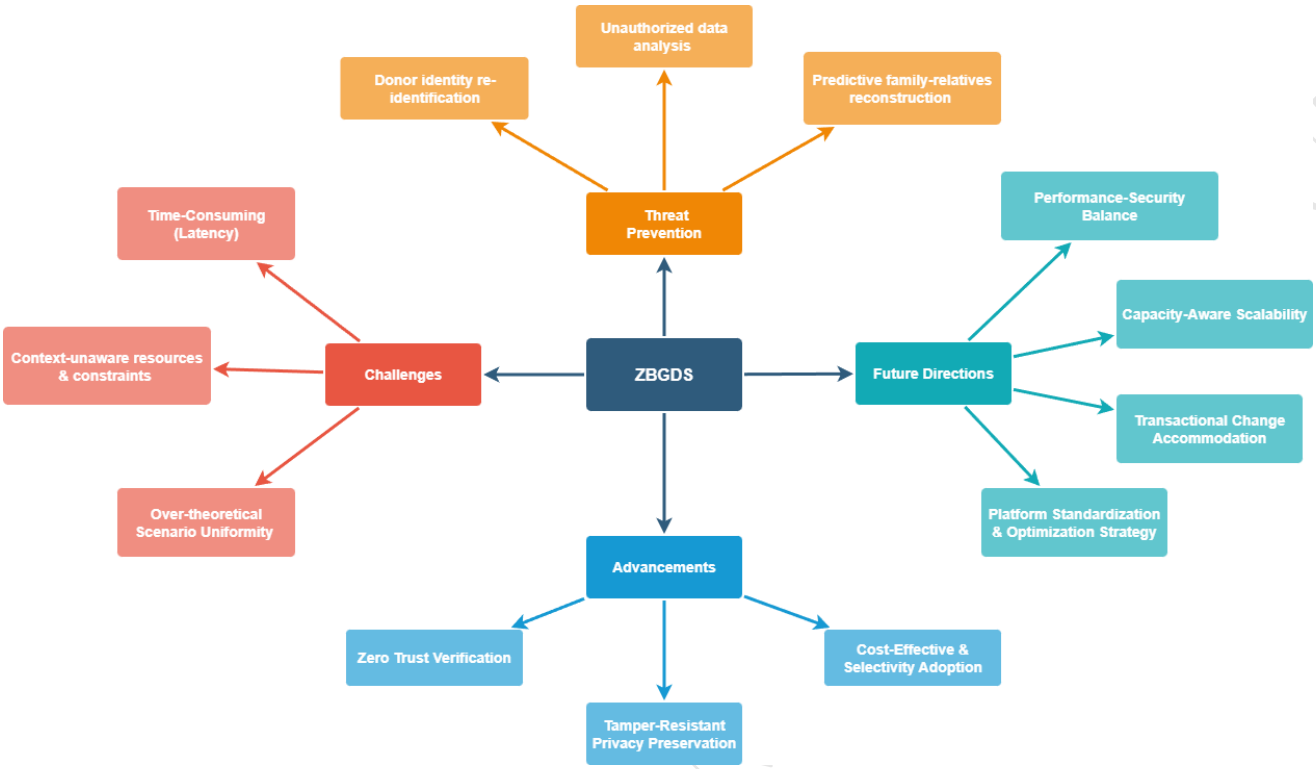

Figure 1: Four core dimensions in ZBGDS: threat prevention, technological advancements, current challenges, and future directions.

2.1 Background concepts

The transformation of BC systems [10] provides immutably decentralized ledgers whose entries are replicated across a network of independently operated nodes. Each node is structured as a linked block sequence encompassing a set of validated transactions and a cryptographic hash pointer to its predecessor. This chain creates an append-only log whose data records are no longer being modified. Modern implementations [9, 17] also include consensus mechanisms, SC agreements, and ZKPs integration.

Consensus mechanisms: determine a node's agreement on the next valid block. Depending on the deployment model, ZBGDS systems may rely on Proof-of-Authority (PoA) [1], Byzantine-fault-tolerant (BFT) protocols [27], or other permissioned-network consensus mechanisms. By eliminating reliance on any single authority, consensus ensures that decisions regarding access, consent updates, or analytic requests remain tamper-evident and auditable across institutions.

SC agreement: extends predefined infrastructure with data access policies. Within ZBGDS, SC [23] can formalize consent management between donors and recipient parties, as well as request regulations and usage constraints. SC transparency for such accountability otherwise exposes operational metadata. Because transactions and contract executions are publicly visible, ZBGDS faces challenges in managing genome information. BC technology, therefore, typically adopts hybrid on-chain/off-chain storage [25]: the chain records policy states and commitments, while genomic datasets and computational payloads reside in external repositories. This separation preserves decentralization and verifiability without disclosing high-dimensional genomic content on a permanent ledger.

ZKPs integration: complements the BC governance between parties to prove the correctness of computational actions without disclosing the underlying data [12]. A ZKP consists of three main stages: proof generation, where a prover constructs a cryptographic witness without gaining access to the hidden inputs. In the second stage, ZKPs are stored with asymmetric encryption [11] in the BC system. When SC is invoked for identity validation, the transaction is executed through proof verification in the third stage.

2.2 Threat Prevention Capabilities

Genomic datasets encode deeply personal information in biological characteristics, making them vulnerable to misuse even when traditional safeguards are in place. As shown in Figure 1, ZBGDS handles several critical threats that directly endanger personal identity and sensitive biological traits.

Unauthorized data analysis: adversaries with non-consenting access to genomic features might infer phenotypic traits, disease predispositions, or ancestry information. The risk is amplified by modern machine-learning pipelines, which can extract clinical or

behavioral insights from even small genomic fragments. Such unauthorized analysis not only violates individual consent but may also facilitate discriminatory practices in insurance, employment, or social contexts.

Donor identity re-identification: genomic sequence functions as a quasi-immutable biometric identifier, linkage attacks combining demographic attributes, public genealogy platforms, or previously leaked genomic datasets. The inherent stability and uniqueness of DNA sequences make these attacks particularly difficult to prevent or remediate. This elevates re-identification risk to one of the most severe privacy threats in biomedical data governance.

Predictive family-relative reconstruction: genomic similarities allow transitive exposure. Due to hereditary structure, predictive family-relative inference places entire genetic lineages at risk, challenging biomedical data governance. This transitive privacy risk challenges traditional consent models, as individuals may expose entire family networks, shifting genomic privacy from an individual to a collective concern.

The presence of high-risk genomic markers is hardly tackled by traditional BC frameworks. ZKPs offer a structural protection form in the accuracy of computations, the validity of access rights, or the compliance with governance policies. Within ZBGDS workflows, ZKPs ensure that only authorized, policy-aligned operations are executed while concealing user-specific attributes, dataset contents, and analytical behavior. This integration significantly offers a privacy-preserving framework aligned with multi-party data-sharing environments.

3 Advancements and Model Comparisons in ZBGDS Schemes

3.1 ZBGDS Referential Architecture

Figure 2 presents a high-level view of a ZBGDS system, organized into three sequential phases that capture the interactions among genomic donors, medical clinics, BC components, and data consumers. This architecture illustrates how ZKPs and BC operate together to enforce policy-compliant GD exchange.

In **Data Collection and Commitment (Phase 1)**, the medical clinic collects and preprocesses the donor's GD, then stores the resulting data commitments in the BC ledger while the GD is stored off-chain in external repositories for alleviating computational cost. During this stage, the clinic performs GD validation, metadata generation, and commitments. These commitments, as defined in the associated SC, are subsequently anchored to the BC storage, forming the basis for subsequent verification steps.

Proof Generation and Registration (Phase 2) is carried out in the ZKP module. The system then generates proofs attesting properties such as data integrity, consent validity, and compliance with access-control policies. These proofs are typically recorded in BC on-chain storage, enabling transparency, tamper-resistance, and auditability across participants.

Finally, GD consumers (hospitals, pharmaceutical companies, research institutions) can request access or perform queries through **Secure Access and Verification (Phase 3)**. When such a request in the transation is made, the SC triggers the ZKP verification process, allowing the consumer to obtain validated outputs without accessing raw genomic content or sensitive donor information. This mechanism preserves privacy by rigorously enforcing access policies.

3.2 ZBGDS Advancements

ZKPs introduce several foundational advances that shape how GD-sharing systems are designed and deployed in BC-based architectures. Each advancement reflects a core architectural principle that enables secure, policy-compliant, and privacy-preserving genomic workflows in multi-party environments.

Zero-Trust Verification: enable an operational model in which neither donors, medical clinics, nor GD consumers need to trust one another. ZKPs allow stakeholders to verify compliance with genomic access policies, data integrity requirements, or query correctness without revealing any raw genomic content or consent information. This zero-trust approach is essential for operations across institutional, geographic, and regulatory boundaries, and for addressing misaligned incentives that can cause serious privacy breaches.

Tamper-Resistant Privacy Preservation: Combining BC immutability with succinct ZKPs strengthens privacy guarantees against tampering and misuse. ZKPs ensure that computations, access rules, and audit trails remain verifiable without exposing genomic sequences or sensitive metadata. ZBGDS model limits attack surfaces arising from ledger transparency and provides a technically enforceable pathway for regulated biomedical deployments.

Selective Disclosure and Cost-Aware Deployment: By enabling selective verification of specific genomic attributes or policy constraints, ZKPs can reduce unnecessary data exposure and verification redundancy. However, proof generation and on-chain verification incur additional computational and gas costs that must be carefully managed to ensure practical deployment in performance-constrained biomedical environments.

Together, these three innovations establish ZKPs as a fundamental supplementary element for building scalable, trusted, and privacy-preserving GD-sharing architectures in distributed settings.

3.3 Evolution and Comparative Analysis of BC-Based Platforms Toward ZKP-Enabled ZBGDS Architectures

Table 1 illustrates that while early blockchain-based systems primarily improve governance auditability, ZKP-enabled architectures increasingly target re-identification and inference threats, albeit at the cost of higher computational and deployment complexity.

This subsection compares the evolution of blockchain-based genomic data-sharing platforms, highlighting how successive generations address trade-offs in governance, privacy, and scalability through different cryptographic and architectural choices.

The transformative era of BC technologies over the past decade provides essential context for understanding how ZBGDS have emerged. From early basic ledgers with transaction integrity to SC-enable platforms and, more recently, privacy-preserving proof systems, each generation has directly shaped the feasibility, scalability, and trust assumptions of genomic governance architectures.

The first generation of BC platforms, initially designed through linked blocks and transparent auditability, the further extended

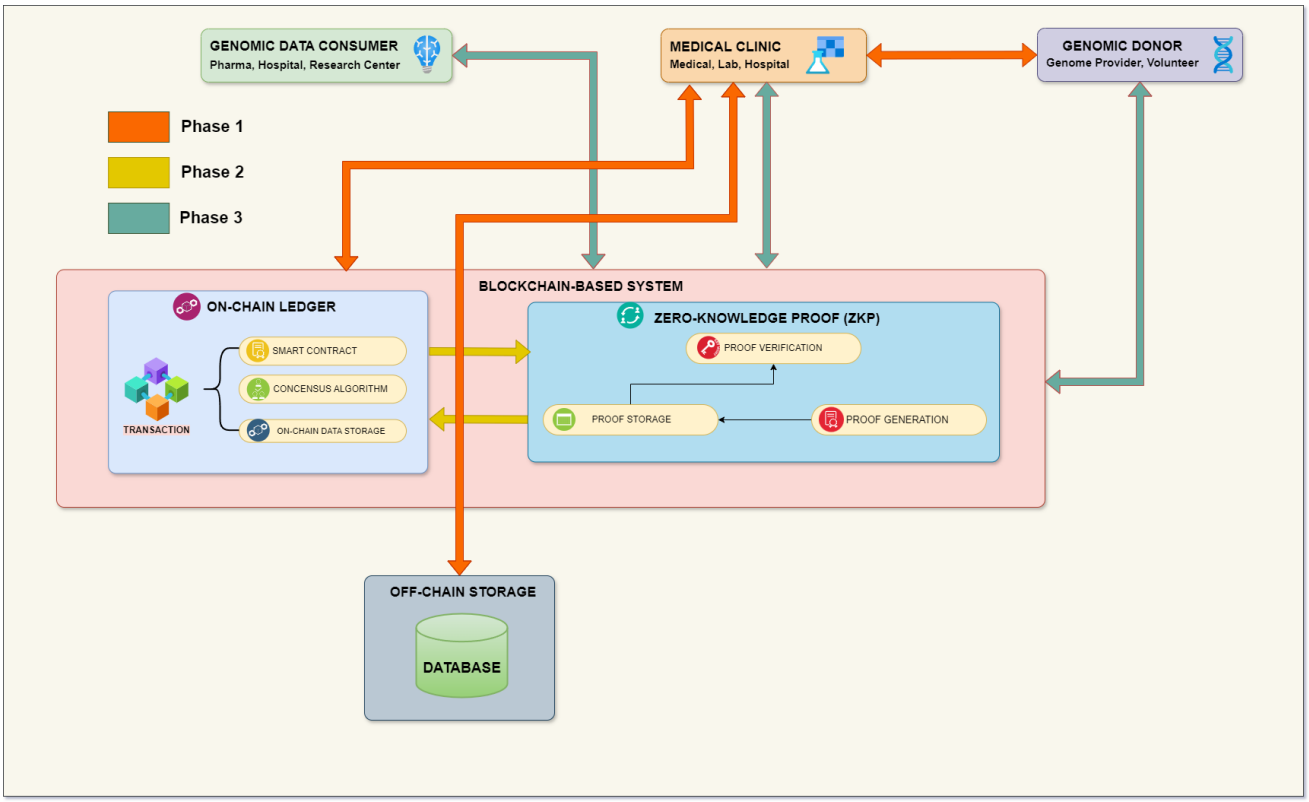

Figure 2: High-level ZBGDS Architecture

to GD provenance. Early genomic prototypes introduced Proof-of-Work (PoW) mechanisms to ensure integrity-preserving timestamping of sequencing data. One such architecture is Atalay M. Ileri et al. work [16] on Coinami frameworks adapted the PoW concept by integrating DNA sequence alignment into the block validation process, thereby converting otherwise computational difficulty into useful high-throughput sequencing (HTS) analysis tasks. Kuo et al. also promoted iDASH Secure Genome Analysis Competition 2018 [18] on BC-based genomic dataset access logging through one in three competition tracks. The contest demonstrated that BC-based ledgers could adopt multiple cross-site access records and support accurate queries within a short period, highlighting the feasibility of decentralization for large genomic repositories. Despite these advances, first-generation systems typically relied on trusted intermediaries and lacked self-automated mechanisms for transparent agreement and stakeholder governance, which motivated the design of later SC–enabled architectures. However, these first-generation systems provide limited cryptographic privacy guarantees and lack automated, fine-grained governance mechanisms suitable for sensitive multi-party genomic data sharing.

SC introduced in the second generation enables self-executing programs for independent authorization and event-driven governance. POPS-G (Privacy and Ownership Protection System for Genomics), proposed by Dakshayini et al. [7], secures sensitive genetic data while emphasizing patient ownership via SCs, and Federico Carlini et al. present Genesy [5] for safeguarding services and authorized transactions among research institutions, hospitals, and pharmaceutical companies. On the other hand, Beyhan et al. introduce KeyGen [8], a dual-BC architecture that combines a permissioned Hyperledger Fabric chain for private data index management with an Ethereum-derived chain for public computation, and integrates cryptographic key management to enable verifiable GD sharing while preserving confidentiality. Despite these advances, SC-based architectures remain inherently transparent, incur high gas costs due to re-executed SC verification, and provoke GD leakage. Therefore, this article highlights future directions involving ZKPs and verifiable computation to achieve a more secure and efficient privacy-preserving mechanism. While smart contracts improve programmability and automate governance, their inherent transparency and execution costs remain fundamental limitations for privacy-sensitive genomic workflows.

Through the third generation of non-interactive ZKPs integration into ZBGDS architecture, though contributions remain limited, it primarily targets three directions: decentralized genomic access control, privacy-preserving governance for federated biomedical AI, and verifiable healthcare data exchange. In decentralized genomic access control, Alghazwi et al. [2] introduced the DARC protocol, which applies Groth16 zk-SNARKs (Zero-Knowledge Succinct Non-Interactive Argument of Knowledge) combined with Merkle Forest structures to verify credentials without revealing identities. This

Table 1: Comparative Overview of Blockchain Generations in GD Sharing Architectures

Article	Genomic Use Case	ZKP / BC Techniques	Primary Threats Addressed	Research Contributions	Considerations & Limitations
Generation 1: Basic BC Architecture					
[16]	DNA sequence analysis	Signed token, PoW	Governance auditability	Converts into HTS read mapping (30 CPU days per human genome)	Early design, dependent on trusted intermediaries, no automated governance
[18]	Genomic dataset access logging	MultiChain (Bitcoin BC fork)	Governance auditability, access traceability	Demonstrated 800,000 cross-site access records queried in 3 minutes	Speed, scalability, and memory cost evaluation limited
Generation 2: SC–Enabled Governance					
[5]	Cross-institution GD exchange	Hyperledger Fabric, GDPR standards	Unauthorized access, consent enforcement	Consent management, tokenized exchange, and auditability	Transition to consortium BC required; difficulty replicating large data; tension with GDPR Right to Erasure
[7]	NFT-based genomic ownership protection	ERC-721 Smart Contracts, IPFS off-chain storage	Unauthorized access, ownership misuse	Real-time consent management and ownership-based sharing	No integration with existing healthcare databases
[8]	GD sharing	Dual-BC (Hyperledger + Ethereum), key management	Unauthorized access, governance auditability	Combines private/public chains enabling verifiable sharing	System complexity, transparency leakage, high cost; full deployment needed
Generation 3: ZKP-Enhanced Platforms					
[4]	DNA STR profile	PoW/PoS ZKPs	Re-identification, forensic misuse	Conceptual workflow with key rotation, secure access control, and auditing	High-level design; limited ZKP integration and implementation details
[2]	Dataset credentials	FL zk-SNARK (Groth16)	Unauthorized access, identity leakage	DARC verification protocol with low proof generation time (2.4 s)	Trusted issuers required; high gas cost (212k gas); no revocation or Sybil resistance
[21]	Clinical diagnostics	Transformer-based ZBGDS	Inference attacks, unauthorized model updates	Secure FL updates with auditability and a fairness gap <3%	High computational cost; ZKP overhead; limited model interpretability

design supports an important step toward decentralized authorization and selective attribute disclosure of such sensitive genomic resources. However, key aspects such as scalability, credential revocation, Sybil resistance, and proof-generation efficiency remain unresolved, leaving DARC far from real-world operational biomedical deployment.

In privacy-preserving governance for federated biomedical AI, Oyebode et al. [21] proposed a ZKP-enhanced federated learning (FL) framework in which institutions prove local compliance updates before global model aggregation. This mitigates poisoning and improves accountability but introduces substantial computational overhead, particularly when applied to Transformer-based architectures. Additionally, institution interoperability and real clinical integration remain unaddressed.

Another innovative ZBGDS approach has been suggested for the management of DNA short tandem repeat (STR) profiles using PoW/Proof of Stake (PoS) in Arunkumar et al. [4]. Their model outlines secure access control, key rotation, and auditability for forensic genomics. However, it remains mostly a high-level conceptual workflow, with no concrete ZKP implementation, limited empirical evaluation, and limited integration. As a result, its contribution lies primarily in motivating the genomic identification of ZKPs rather than in deployable mechanisms. While smart contracts

improve programmability and automate governance, their inherent transparency and execution costs remain fundamental limitations for privacy-sensitive genomic workflows.

Overall, these early studies demonstrate the feasibility of integrating non-interactive ZKPs into biomedical and governance of GD. However, they collectively suffer from similar limitations: high proof-generation costs, lack of standardization, lack of interoperability, trusted setups, incomplete identity and revocation mechanisms, and, most importantly, a complete absence of real-world deployment across heterogeneous healthcare scenarios. Table 1 synthesizes the core ZKP techniques, contributions, and limitations of ZBGDS representatives.

While these ZKP-enabled architectures demonstrate clear privacy and governance advantages over earlier generations, their practical deployment exposes a range of unresolved technical and operational challenges.

4 Open Challenges and Research Directions

In the realm of cryptographic platforms, ZBGDS has demonstrated considerable promise in enhancing privacy while still promoting collaborative research. However, some technical and operational limitations remain, which presents potential solutions to achieve practical, scalable deployment. In this section, we categorize these issues into two main areas: existing systematic challenges (Section 4.1) encompassing current bottlenecks in real-world scenarios, and future development directions (Section 4.2) outlining ongoing research opportunities to improve ZBGDS performance, scalability, and interoperability.

4.1 Existing systematic challenges

Current ZBGDS ecosystems face several open challenges that hinder real-world deployment and scalability. These limitations arise not only from cryptographic overheads but also from operational, environmental, and methodological gaps that emerge when theoretical models confront the complexity of real-world healthcare ecosystems.

Time-Consuming (Latency): Generating ZKPs for large-scale genomic datasets remains computationally intensive. Producing a zk-SNARK or similar non-interactive proof often requires intensive preprocessing, a large memory footprint, and slow execution time. Proof creation and verification can introduce significant delays that hinder real-time multi-institutional research analysis workflows, where even small latencies can accumulate into processing bottlenecks.

Context-unaware resources & constraints: often overlooking the heterogeneity of computational environments and network constraints through participating institutions in current ZBGDS implementations. Cloud-based, on-premises, and hybrid deployments have varying performance characteristics, yet most frameworks assume uniform resource availability, which limits data orchestration and robustness. As a result, less-resourced institutions may experience disproportionately higher delays or failures, inadvertently widening the technological gap among participants and weakening the collaborative nature of GD sharing.

Over-theoretical scenario uniformity: evaluating in controlled, idealized environments that do not reflect the variability of real-world GD-sharing. Many factors, such as incomplete data, heterogeneous consent policies, and fluctuating user participation, are often underexplored, thereby reducing the practical applicability of research prototypes. In practice, genomic workflows encounter incomplete metadata, diverse access-control policies, intermittent connectivity, evolving roles and permissions, and varying ethical constraints. Without evaluating systems under realistic conditions, current ZBGDS frameworks risk underestimating the engineering required for deployment across heterogeneous biomedical infrastructures.

These limitations reveal that current ZBGDS architectures remain far from production maturity. High-proving latency, limited awareness of heterogeneous computational contexts, and reliance on overly simplified evaluation settings collectively constrain their reliability, scalability, and adaptability in real genomic ecosystems. Addressing these systemic gaps requires future research that integrates performance-aware ZKP engineering, context-adaptive orchestration mechanisms, and realistic, compliance-aligned deployment models to ensure that privacy-preserving GD sharing can function effectively across diverse biomedical landscapes.

4.2 Future research directions

With the target of reducing identified vulnerabilities, certain principal directions in ZBGDS research suggest improving efficiency, scalability, and adaptability, without compromising privacy and security.

Performance–Security Balance: achieve an optimal trade-off between complex cryptographic privacy and proof size. Although ZKPs provide strong privacy guarantees, their generation and integration into smart contracts can incur significant computational overhead. Research into collaboration among non-interactive proof, storage optimization, and hybrid blockchain approaches may help reduce latency without compromising security. The main objective is to scale to population-level genomic datasets without imposing prohibitive costs on participating medical institutions.

Capacity-aware Scalability: incorporate scalable storage, proof aggregation, and decentralized indexing in ZBGDS platforms. Efficient batching, recursive proofs, and off-chain computation frameworks remain essential research opportunities. Integrating verifiable computation frameworks into genomic infrastructures may further support dynamically adaptable systems that vary with institutional resources, network conditions, and workload patterns.

Transactional Change Accommodation: leverage flexible and update-friendly smart contract designs. Ensuring auditability in terms of dynamic consent, data revocation, and evolving access privileges while preserving privacy over time remains unresolved, particularly for long-term genomic storage. This direction calls for cryptographic proofs, combined with programmable governance mechanisms that can evolve alongside biomedical and regulatory needs.

Platform Standardization and Optimization Strategy: To facilitate cross-institutional collaboration and the adoption of ZBGDS platforms, interoperability must be established through standardized APIs, semantic metadata encoding, and integrated regulatory

compliance (e.g., GDPR, HIPAA). These standards should be complemented by optimization strategies that span storage, computation, and protocol design.

Advancing ZBGDS requires a holistic approach that leverages cryptographic innovation, system optimization, regulatory alignment, and domain-specific bioinformatics expertise. The convergence of BC, ZKPs, and genome science creates a pathway toward privacy-preserving GD ecosystems.

5 Conclusion

This survey provided a focused ZBGDS examination, highlighting the unique architectural, operational, and privacy requirements. We further advanced a concept-to-model perspective by unifying ZKP principles with emerging system designs, emphasizing their role in threat mitigation and performance–privacy trade-offs. Complementing this, recent prototypes and platforms have advanced practical applications and highlighted unresolved challenges that continue to shape ZBGDS's future directions. In general, our analysis underscores the central role of ZKPs in enabling trustworthy, privacy-preserving GD infrastructures, while highlighting the need for interdisciplinary progress to translate into real-world applications.

Acknowledgments

This work was supported by the TTDGen project, funded under the ATIGE program of Genopole.

References

[1] Faisal Albalwy, Andrew Brass, and Angela Davies. 2021. A blockchain-based dynamic consent architecture to support clinical genomic data sharing (ConsentChain): Proof-of-concept study. *JMIR medical informatics* 9, 11 (2021), e27816.

[2] Mohammed Alghazwi, D Karastoyanova, and F Turkmen. 2023. DARC: Decentralized Anonymous Researcher Credentials for Access to Federated Genomic Data. In *Proceedings of the International Workshop on Trends in Digital Identity (TDI)*.

[3] Mohammed Alghazwi, Fatih Turkmen, Joeri Van Der Velde, and Dimka Karastoyanova. 2022. Blockchain for genomics: a systematic literature review. *Distributed Ledger Technologies: Research and Practice* 1, 2 (2022), 1–28.

[4] P Arunkumar, B Surendiran, S Suresh, and V Sankaranarayanan. 2024. Zero-Knowledge Proof Approach for DNA STR Profile Security using Blockchain: A Framework for Enhancing Genetic Privacy. *NFSU Journal of Cyber Security and Digital Forensics* (2024).

[5] Federico Carlini, Roberto Carlini, Stefano Dalla Palma, Remo Pareschi, and Federico Zappone. 2020. The Genesy model for a blockchain-based fair ecosystem of genomic data. *Frontiers in Blockchain* 3 (2020), 483227.

[6] Mauricio Chalita, Yeong Ouk Kim, Sein Park, Hyun-Seok Oh, Jae Hyoung Cho, Jeongsup Moon, Nuga Baek, Changsik Moon, Kihyun Lee, Junwon Yang, et al. 2024. EzBioCloud: a genome-driven database and platform for microbiome identification and discovery. *International Journal of Systematic and Evolutionary Microbiology* 74, 6 (2024), 006421.

[7] M Dakshayini, Ammaji Kavalleswari, and H S Suhasini. 2024. Blockchain-NFT Enabled Privacy and Ownership Protection System for Genomics[POPS-G]. In *2024 International Conference on Emerging Technologies in Computer Science for Interdisciplinary Applications (ICETCS)*. 1–6. doi:10.1109/ICETCS61022.2024.10544201

[8] Beyhan Adanur Dedeturk, Ahmet Soran, and Burcu Bakir-Gungor. 2025. GenShare: A Blockchain-based Genomic Data Sharing Platform. *Distributed Ledger Technologies: Research and Practice* (2025).

[9] Ahmed Elhussein, Ulugbek Baymuradov, Noémie Elhadad, Karthik Natarajan, and Gamze Gürsoy. 2024. A framework for sharing of clinical and genetic data for precision medicine applications. *Nature medicine* 30, 12 (2024), 3578–3589.

[10] G Fathima et al. 2024. Investigating novel approaches to privacy-aware healthcare data sharing in cloud environment. In *2024 International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 1485–1492.

[11] Tao Feng, Pu Yang, Chunyan Liu, Junli Fang, and Rong Ma. 2022. Blockchain Data Privacy Protection and Sharing Scheme Based on Zero-Knowledge Proof. *Wireless Communications and Mobile Computing* 2022, 1 (2022), 1040662.

[12] Sarthak Gangurde, Ashwini Jadhav, Vijay Gatkal, and Mansi More. 2025. Zk-Gene: A Zero-Knowledge Proof Framework for Secure Genetic Marker Verification. In *2025 Global Conference in Emerging Technology (GINOTECH)*. IEEE, 1–5.

[13] Rodrigo Dutra Garcia, Gowri Ramachandran, Kealan Dunnett, Raja Jurdak, Caetano Ranieri, Bhaskar Krishnamachari, and Jo Ueyama. 2024. A Survey of Blockchain-Based Privacy Applications: An Analysis of Consent Management and Self-Sovereign Identity Approaches. *arXiv preprint arXiv:2411.16404* (2024).

[14] Geoffrey S Ginsburg, Thomas W Burke, and Phillip Febbo. 2008. Centralized biorepositories for genetic and genomic research. *Jama* 299, 11 (2008), 1359–1361.

[15] Seoyeon Hwang, Ercan Ozturk, and Gene Tsudik. 2023. Balancing Security and Privacy in Genomic Range Queries. *ACM Trans. Priv. Secur.* 26, 3, Article 23 (March 2023), 28 pages. doi:10.1145/3575796

[16] Atalay M Ileri, Halil I Ozercan, Alper Gundogdu, Ahmet K Senol, M Yusuf Ozkaya, and Can Alkan. 2016. Coinami: a cryptocurrency with DNA sequence alignment as proof-of-work. *arXiv preprint arXiv:1602.03031* (2016).

[17] Adnan Imeri, Nazim Agoulmine, and Djamel Khadraoui. 2024. Blockchain and Smart Contract for Trusted Decentralized Digital Genomics. In *International Workshop on ADVANCEs in ICT Infrastructures and Services*.

[18] Tsung-Ting Kuo, Xiaoqian Jiang, Haixu Tang, XiaoFeng Wang, Tyler Bath, Diyue Bu, Lei Wang, Arif Harmanci, Shaojie Zhang, Degui Zhi, et al. 2020. iDASH secure genome analysis competition 2018: blockchain genomic data access logging, homomorphic encryption on GWAS, and DNA segment searching. *BMC medical genomics* 13, Suppl 7 (2020), 98.

[19] Go Eun Myeong and Kim Sa Ram. 2025. Blockchain Based Zero Knowledge Proof Protocol For Privacy Preserving Healthcare Data Sharing. *Journal of Technology Informatics and Engineering* 4, 1 (2025), 171–189.

[20] Adrien Oliva, Anubhav Kaphle, Roc Reguant, Letitia MF Sng, Natalie A Twine, Yuwan Malakar, Anuradha Wickramarachchi, Marcel Keller, Thilina Ranbaduge, Eva KF Chan, et al. 2024. Future-proofing genomic data and consent management: a comprehensive review of technology innovations. *GigaScience* 13 (2024), giae021.

[21] O Oyegoke. 2024. Transformers on encrypted federated datasets anchored by blockchain zero-knowledge proofs for privacy-preserving multilingual healthcare diagnostics and equity. *Int J Res Publ Rev* 5, 12 (2024), 6112–28.

[22] Midhun Punukollu. 2022. AI-Driven Genomic Sequencing: Revolutionizing Personalized Medicine Through Predictive Analytics. *Los Angeles Journal of Intelligent Systems and Pattern Recognition* 2 (2022), 293–329.

[23] Mahsa Shabani. 2019. Blockchain-based platforms for genomic data sharing: a de-centralized approach in response to the governance problems? *Journal of the American Medical Informatics Association* 26, 1 (2019), 76–80.

[24] Emil Uffelmann, Qin Qin Huang, Nchangwi Syntia Munung, Jantina De Vries, Yukinori Okada, Alicia R Martin, Hilary C Martin, Tuuli Lappalainen, and Danielle Posthuma. 2021. Genome-wide association studies. *Nature Reviews Methods Primers* 1, 1 (2021), 59.

[25] Leon Visscher, Mohammed Alghazwi, Dimka Karastoyanova, and Fatih Turkmen. 2022. Poster: Privacy-preserving genome analysis using verifiable off-chain computation. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. 3475–3477.

[26] Jayneel Vora, Anand Nayyar, Sudeep Tanwar, Sudhanshu Tyagi, Neeraj Kumar, Mohammad S Obaidat, and Joel JPC Rodrigues. 2018. BHEEM: A blockchain-based framework for securing electronic health records. In *2018 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 1–6.

[27] Gang Xu, Tengkai Yao, Kejia Zhang, Xiangfei Meng, Xin Liu, Ke Xiao, and Xiubo Chen. 2023. An optimized Byzantine fault tolerance algorithm for medical data security. *Electronics* 12, 24 (2023), 5045.

[28] Lu Zhou, Abebe Diro, Akanksha Saini, Shahriar Kaisar, and Pham Cong Hiep. 2024. Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. *Journal of Information Security and Applications* 80 (2024), 103678. doi:10.1016/j.jisa.2023.103678

Received 22 December 2025; revised 12 March 2025; accepted 5 June 2009

GISSA GPT: An Agent-Oriented Architecture for Intelligent Governance in Digital Health

Caio Leandro Rodrigues Cavalcanti
caio.leandro.rodrigues07@aluno.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Fabio José Gomes de Sousa
prof.fabiojose@gmail.com
FIOCRUZ/BA
Eusébio, CE, Brazil

Rodrigo Matos Aguiar
rodrigo.matos9@hotmail.com
IFCE
Fortaleza, CE, Brazil

César Olavo de Moura Filho
cesar.olavo2011@gmail.com
IFCE
Fortaleza, CE, Brazil

Luiz Odorico Monteiro de Andrade
odorico.monteiro@fiocruz.br
FIOCRUZ/CE
Eusébio, CE, Brazil

Antônio Mauro Barbosa de Oliveira
mauro@lar.ifce.edu.br
PPGCC-IFCE
Fortaleza, CE, Brazil

Abstract

The GISSA system (Intelligent Governance in Health Systems) is a computing platform, implemented in 2020, that automatically collects data from Ministry of Health information systems in Brazil: e-SUS, CNES, SIM, SINASC, SI-PNI, and SINAN. It does so through data-extraction bots, analyzes these data using multiple intelligent technologies, and transforms them into integrated information that supports decision-making at different levels of health management. In 2021, the Networks and Systems Laboratory team at IFCE (LAR) implemented "Smart GISSA", a health-governance system based on Machine Learning, defended as a doctoral dissertation at the Federal University of Ceará [9]. In 2024, the same LAR team developed "Giselle Saúde", a sentiment-detection system using Generative AI in Digital Health [16]. This paper presents GISSA GPT, an evolution of Smart GISSA that incorporates Large Language Models (LLMs), leveraging the expertise acquired by the LAR team in building Giselle Saúde. It is an agent-oriented prototype for digital health governance with the following characteristics: (i) a generative-AI environment inspired by Smart GISSA; (ii) an event-driven design for integration among modules; and (iii) Retrieval-Augmented Generation (RAG) mechanisms to anchor responses in verifiable sources, implemented with modern technologies (n8n, LangChain, ChromaDB, etc.) and able to incorporate new protocols and guidelines selected by a multidisciplinary team [13, 18]. Considering that Giselle Saúde focuses on mental health, future work proposes integrating GISSA GPT with Giselle Saúde.

Keywords

e-Health, Workflow Management, Information and Data Management, Decision Analysis and Methods, Big Data Management, Privacy protection and Privacy-by-design, Security and Trust Management, AI for services infrastructure optimization

1 Introduction

The expansion of digital health initiatives has been driven by the increasing volume of clinical and administrative data, the need to broaden access, and advances in information and communication technologies. However, in sensitive domains such as healthcare, expanding access and automating workflows is not enough. Information provided by digital systems must be traceable, auditable, and aligned with ethical and regulatory practices; otherwise, it may lead to inadequate decisions and deepen care asymmetries [19, 3, 18].

Large Language Models (LLMs), based on Transformer architectures, have shown the ability to generate coherent text, synthesize information, and interact through natural language [17, 7]. Yet, such models are not verifiable knowledge bases. Because they operate probabilistically, they can produce plausible statements that are incorrect or unsupported by evidence—a phenomenon often described as hallucination. In healthcare, responses without documentary support can lead to severe clinical and social consequences, especially in surveillance, mental health, and vulnerable populations [2, 18].

Retrieval-Augmented Generation (RAG) has been used to mitigate these limitations by combining document retrieval with controlled text generation. By grounding responses in a validated corpus, the approach promotes transparency, auditability, and governance, making it possible to trace the origin of the information used by the model [13, 20, 18].

This paper presents GISSA GPT, an agent-oriented architecture for governance and decision support in digital health, conceived as an evolution of Smart-GISSA and based on recent experience with Giselle Saúde, a platform for older adults' mental health that uses Generative AI for sentiment detection and analysis [9, 16]. It comprises a Sentiment Detector (SD) and a Generative Virtual Assistant (GVA) to analyze sentiments expressed in interactions between users and health professionals, identifying older adults who may require professional evaluation for specialized care based on urgency, resulting in in-person or remote consultations [16].

The proposal seeks to address pain points observed in surveillance and management environments, such as: (i) low transparency regarding the origin of information; (ii) difficulty integrating services and institutional documents; and (iii) lack of natural-language explanations across multiple heterogeneous sources. GISSA GPT is presented both as an architectural reference and as a reproducible design protocol, including limitations and mitigation mechanisms.

The scope includes decision support and governance based on indicators, alerts, and institutional documents, with traceability

and an audit trail; it includes conversational interaction with explicit sources and controlled retrieval over repositories validated by technical and health teams. It excludes automated diagnosis, therapeutic prescription, and individualized guidance in urgent situations, which must be handled by licensed professionals [18].

2 Smart GISSA

The GISSA platform (Intelligent Governance in Health Services) provides the foundation on which Smart-GISSA was built. Its architecture was designed to overcome data fragmentation by acting as a large integrator of health information [15, 9].

The architecture can be seen in Figure 1:

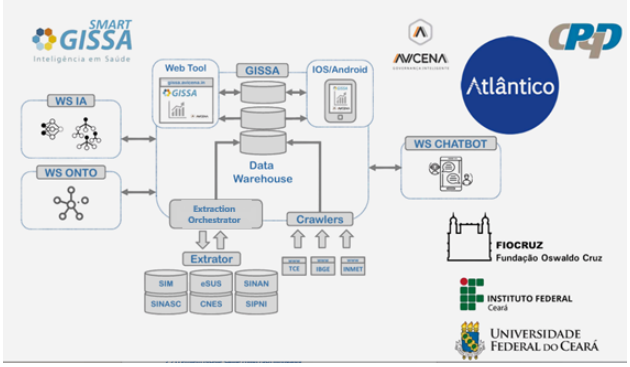

Figure 1: High-level view of the Smart-GISSA.

Smart-GISSA represents the evolution of the GISSA platform by incorporating an intelligence layer that transforms the system from a data repository into a predictive analytics tool [9]. The new architecture expands the capabilities of the original system, preserving the solid data-integration base while adding new specialized web services (WS):

- **WS AI (Artificial Intelligence):** a microservice that encapsulates trained Machine Learning models, returning predictions, classifications, and risk analyses (e.g., probability of maternal/infant death or epidemic trends) [9].
- **WS ONTO (Ontology):** organizes domain knowledge to make sense of heterogeneous data, with semantic relationships and inferences (e.g., diseases, symptoms, risk factors), contributing to semantic interoperability [11, 12].
- **WS CHATBOT:** a service that enables natural-language questions and contextualized answers based on the platform's data and analyses [14].

The Smart-GISSA architecture is conceived as a layered model, following the data value chain: from capture at primary sources, through integration and storage in the Data Warehouse, to the application of AI models and the delivery of results to end users through multiple interfaces. As shown in Figure 2, this modular, microservice-based approach provides greater flexibility, scalability, and ease of maintenance, enabling new functionalities and models to be added independently [9, 4].

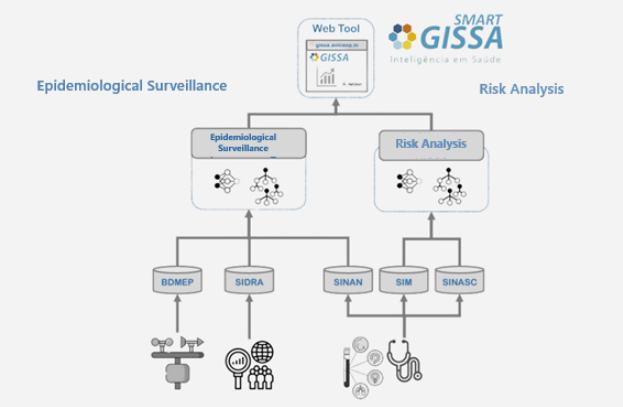

Figure 2: Layered Smart-GISSA architecture with specialized services.

Among the services implemented in Smart-GISSA, the following stand out:

- **Data Mining for Risk of Death (DMRisD):** Maternal and infant deaths are tragedies, many of which could be prevented with timely intervention. The challenge is to identify high-risk pregnant women and newborns as early as possible, within a critical time window. The DMRisD approach was designed to support risk stratification and prioritization in surveillance and care pathways [9].
- **Data Mining for Epidemics (DMEpi):** Arboviruses such as Dengue, Zika, and Chikungunya represent an ongoing threat to public health. Predictive models can help anticipate spatial and temporal risk, complementing traditional surveillance [6, 8].

This technological and institutional trajectory forms the basis on which GISSA GPT is positioned.

3 Theoretical background

3.1 Digital health and virtual assistants

Recent literature characterizes digital health as an umbrella field that integrates *eHealth*, *mHealth*, telemedicine, connected devices, and intelligent systems aimed at organizing care and enabling communication between professionals and users [19, 3]. In this context, virtual assistants have been used for triage, health education, and decision support across different domains, potentially improving access, standardization, and continuity of care [10, 14].

However, the same attributes that enable personalization and real-time interaction also intensify concerns about privacy, bias, algorithmic opacity, and technological dependence—especially when Generative AI is used in clinical domains [4, 2, 3]. In such cases, traceability and explainability become requirements for governance and safety, allowing reconstruction of the informational path that supports a given answer [13, 19].

The challenge becomes even more pronounced in mental health, where communication involves emotional, cultural, and linguistic nuances. In the Giselle Project, the generative assistant is described as capable of producing responses that combine contextual adequacy with the user's emotional and cultural state. Even so, the

project itself emphasizes that such innovations require clinical studies to assess efficacy, effectiveness, and safety, as well as explicit responsible-use policies [16].

In this scenario, GISSA GPT aims to advance the governance axis. Its distinguishing feature is not merely the use of LLMs, but the integration of specialized agents, RAG, and mechanisms for observability, auditability, and documentary traceability [13, 20]. By grounding answers in controlled and verifiable repositories validated by technical and health teams, GISSA GPT treats virtual assistants as infrastructure components for decision support, rather than generic conversational systems [19]. Thus, the proposal shifts the focus from access and interaction to transparency, verifiability, and care governance—core aspects in regulated, high-risk environments [19, 3].

3.2 The Giselle Saúde Project as a motivating case

The Giselle Saúde Project is presented as a platform focused on older adults' mental health, composed of a Sentiment Detector and a Generative Virtual Assistant [16]. Its operational goal is to identify, through conversational interactions, signals indicating the need for professional evaluation, referring users to in-person or remote care according to urgency and priority criteria, thereby integrating technology and human care [16].

The paper describing the Giselle architecture details a model in which the *prompt* is co-designed by health professionals and implemented by a technical team, guiding the dialog flow and handling sensitive cases. The design explicitly highlights the need for safety mechanisms, human oversight, and informed consent, and shows that mental health recommendations require traceability, justification, and accountability [19].

As a motivating case, Giselle exposes a gap that goes beyond mental health: generative conversational assistants are not, by themselves, auditable decision-support systems. Coupling an LLM to a chatbot enables natural interaction, but it does not satisfy requirements such as audit trails, information provenance, integration across heterogeneous sources, governance, and explainability—which are essential in regulated domains [2, 4]. In scientific and technological terms, this translates into open challenges such as:

- the absence of verifiable sources in generated answers;
- the difficulty of reconstructing reasoning and data provenance;
- the lack of integration with epidemiological indicators and institutional documents;
- the absence of governance and policy layers for safe use;
- the need for human escalation mechanisms;
- and the demand for reproducible protocols to evaluate efficacy, effectiveness, and safety.

Such limitations are generalizable and appear in recent literature discussing *clinical copilots* and decision-support systems based on Generative AI, which face barriers related to auditability, institutional integration, and regulatory robustness [2, 4].

In this sense, GISSA GPT is proposed as an architectural advance oriented toward governance, integrating specialized agents, RAG components, and documentary traceability to support decision-making in surveillance and health management environments [13, 20]. The architecture results not only from applying LLMs, but from internalizing the gap observed in Giselle: conversational assistants must operate as institutional infrastructure, not merely as natural-language interfaces [19].

3.3 Capabilities and limitations of LLMs for digital health

Large Language Models (LLMs) based on Transformer architectures, such as ChatGPT, have expanded the frontier of digital health applications by enabling information synthesis, document summarization, guideline translation, and natural-language interaction [17, 7, 4]. Functionally, these models act as mediators between users and systems, converting indicators, clinical records, and administrative workflows into narratives that are easier to understand, potentially improving communication, health education, and decision support in institutional contexts [4, 19].

These capabilities are especially relevant in domains characterized by diverse documents, formats, and ontologies—a typical case in healthcare, where clinical guidelines, epidemiological data, administrative protocols, and care histories coexist as heterogeneous sources [3, 12]. In such scenarios, conversational mediation by LLMs can reduce linguistic and cognitive barriers while enabling integration across previously fragmented systems. For mental-health assistants, as discussed in the previous subsection, this mediation includes emotional, cultural, and linguistic nuance.

Nevertheless, adopting LLMs in digital health involves substantive challenges. Because of their probabilistic nature, these models do not operate as verifiable knowledge bases and can generate plausible statements that are incorrect or unsupported by documentary evidence. In regulated domains, answers without traceable evidence can undermine clinical safety, institutional trust, and continuity of care, amplifying ethical and social risks for vulnerable populations [2].

Recent literature thus identifies a set of limitations that are particularly relevant for digital health:

- (i) lack of traceability—difficulty identifying which sources and reasoning support the answer;
- (ii) low auditability—inability to reconstruct the informational path for oversight and institutional responsibility;
- (iii) limited integration with legacy systems—need for interoperability with protocols, indicators, and official documents;
- (iv) algorithmic opacity—difficulty explaining and interpreting outputs in clinical decisions;
- (v) insufficient governance—lack of explicit policies for responsible use in regulated environments.

In light of these limitations, the literature suggests complementing LLMs with additional verification, human oversight, and documentary-integration mechanisms, such as RAG, specialized agents, audit trails, and informed consent [13, 20, 19]. These mechanisms shift conversational assistants from an access/interaction axis to a transparency/governance axis, which is essential in public systems and surveillance/management environments that require *accountability*, safety, and institutional alignment [19, 3].

Accordingly, the Giselle case discussed earlier highlights a technological gap that motivates this work: LLMs can generate dialog, but they do not guarantee governance, and building digital health

systems based on Generative AI requires architectures that incorporate traceability, explainability, and integration with institutional workflows—requirements addressed by the GISSA GPT proposal [19].

4 Related Work

The advance of digital technologies applied to healthcare has produced a diverse ecosystem of solutions aimed at epidemiological surveillance, decision support, automated triage, health education, and service management. These solutions can be grouped into three main streams in recent literature: (i) health data monitoring and analysis systems; (ii) conversational assistants and intelligent agents for healthcare; and (iii) Generative AI architectures and Retrieval-Augmented Generation (RAG) techniques.

(i) Health data monitoring and analysis systems. Platforms such as NSSP/BioSense illustrate the centrality of institutional infrastructures for data collection and analysis, combining data integration, dashboards, and alerts to support decisions and public health planning [8]. Other contemporary initiatives, such as *Outbreaks Near Me*, have also been described as mechanisms for monitoring and risk communication in public health [5]. In Brazil, discussions on standardization and interoperability indicate that sustainable integration requires well-defined models and processes, especially when consolidating heterogeneous repositories and rationalizing administrative data [1, 3]. In this context, GISSA and Smart-GISSA align with approaches that aim to unify sources and enable distributed, multipurpose analyses while maintaining governance and consistency over data use [9, 15].

(ii) Conversational assistants and intelligent agents in healthcare. Conversational assistants have been investigated as technologies for access, health education, and support across different domains, including mental health [14, 10]. The literature also reports limitations related to traceability, explainability, and risk mitigation, particularly when conversational systems operate without explicit integration to governed repositories and without formal accountability mechanisms [18]. The Giselle Saúde project contributes to this stream by combining Generative AI with sentiment detection/analysis and human supervision, describing requirements tied to consent, safety, and cultural adequacy for vulnerable populations [16].

(iii) Generative AI, RAG, and governance. Large Language Models (LLMs) have been explored for tasks such as information synthesis, result explanation, and mediation of natural-language queries [4]. However, a recurring concern is that LLMs are not verifiable knowledge bases and may produce plausible but incorrect answers, complicating auditing when there is no grounding in evidence [2, 18]. Retrieval-Augmented Generation (RAG) techniques have emerged as an alternative to ground answers in traceable sources by combining retrieval of relevant excerpts from controlled corpora with conditioned text generation [13, 20]. In digital health, this arrangement is often considered relevant for transparency and governance, although there is still limited systematization of RAG and observability practices for institutional decision-making environments [19, 18].

Convergence and gaps. The literature offers relevant contributions but still presents gaps for the problem addressed in this paper:

(1) **Governance and auditing.** Few works treat LLMs as auditable components embedded in formal health-management workflows, with recorded context to support decisions [18].
(2) **Operational integration.** Conversational systems often do not integrate, within a single *pipeline*, epidemiological indicators, analytical/predictive models, and institutional normative documents [19].
(3) **Institutional explainability.** Part of the literature discusses clinical or algorithmic explanations; there is less emphasis on explanations aimed at managers and oversight bodies, with explicit traceability and provenance [18].
(4) **Agent-based orchestration.** The combination of specialized agents, RAG, LLMs, and observability mechanisms is still rarely described as a structured arrangement for digital health [13, 20].

GISSA GPT positions itself in this space by proposing an architecture that integrates governed data (GISSA), analytical services (Smart-GISSA), and institutional documents; offers natural-language conversational interaction mediated by agents; incorporates RAG techniques to anchor answers in traceable sources; and includes observability and auditing mechanisms to support decision-making in digital health [19, 18, 13].

5 GISSA GPT Architecture

The GISSA GPT architecture was designed to reuse the existing infrastructure of GISSA and Smart-GISSA while incorporating Large Language Models (LLMs) as a conversational interface module and Retrieval-Augmented Generation (RAG) mechanisms organized around specialized agents [13, 20]. Its central goal is to support governance and decision-making in digital health while preserving traceability, auditability, and alignment with institutional workflows [19, 3].

Structurally, GISSA GPT adopts a microservice- and event-oriented approach, in which relatively independent modules communicate through queues, *webhooks*, and APIs. The architecture is organized into functional modules: (i) data ingestion and governance; (ii) analytics and risk services; (iii) the sub-symbolic module (LLM and RAG); (iv) agent orchestration; (v) natural-language interaction; and (vi) observability and auditing [4, 13].

5.1 Overview

At a high level, GISSA GPT acts as a conversational governance layer over the GISSA/Smart-GISSA ecosystem:

- it receives natural-language questions and commands from managers and health professionals;
- it identifies intent and task type (indicator query, protocol explanation, summary generation, risk analysis, etc.);
- it triggers specialized agents that query GISSA indicators, Smart-GISSA predictive models, and institutional documents;
- it uses RAG to ground the answer in verifiable sources;
- it returns an explained response with explicit references to the origin of data, an audit trail, and the option to record the interaction for governance purposes.

The architecture is agnostic to the LLM provider (it can consume different models via APIs), but it requires that every generated answer be linked to a set of retrieved evidence and that the interaction be logged with sufficient metadata for later auditing [19, 13].

5.2 Functional modules

5.2.1 Data and governance module (GISSA). The first module reuses GISSA as the primary data source:

- extraction bots collect data from the Ministry of Health systems (e-SUS, CNES, SIM, SINASC, SI-PNI, SINAN, among others);
- the data are integrated into an analytical repository (*data lake*/*data warehouse*), with standardized dictionaries, schema versioning, and access policies;
- data quality, anonymization/pseudonymization, and aggregation rules are applied according to current digital-health norms and data-protection requirements.

This module ensures that any query handled through GISSA GPT is supported by governed, documented, and versioned data [19, 3]. When applicable, standardization may adopt semantic interoperability references and electronic health record modeling approaches (e.g., openEHR/ADL archetypes) to maintain consistency over time and across services [1, 12].

5.2.2 Symbolic module (Smart-GISSA). The second module corresponds to the Machine Learning services developed within Smart-GISSA:

- predictive models and risk classifiers;
- population stratification services;
- anomaly detection or identification of relevant epidemiological patterns.

These services are exposed as independent APIs and can be invoked by GISSA GPT agents to compose natural-language answers [9]. Thus, the LLM does not generate predictions autonomously; it orchestrates and explains results produced by validated models [2, 19].

5.2.3 Sub-symbolic module (LLM and RAG). The third module organizes the documentary knowledge relevant for health governance:

- clinical protocols and national guidelines;
- resolutions, ordinances, and technical notes;
- internal manuals, administrative flows, and GISSA documents;
- institutionally validated reports and *dashboards*.

Documents are processed in an indexing *pipeline* (for example, using LangChain + ChromaDB or equivalent), with chunking, metadata enrichment (source, date, version, document type), and creation of vector and symbolic indices [13, 20]. During interaction, the RAG module:

(1) receives the natural-language query;
(2) generates a vector representation of the question;
(3) retrieves the most relevant passages (documents, protocols, notes);
(4) provides this context to the LLM, which must cite or explicitly indicate the sources used.

5.2.4 Agent orchestration module. The fourth module hosts intelligent agents responsible for specific tasks. Examples:

- **Epidemiology Agent** — queries GISSA indicators, applies filters (time, territory, age range), and produces analytical summaries;
- **Data Quality Agent** — checks integrity, completeness, and consistency of requested indicators;
- **Governance/Compliance Agent** — verifies whether the answer involves protocols, norms, or regulatory risk and injects additional explanations;
- **Explanation Agent** — turns numeric and technical outputs into narratives understandable to non-specialist managers.

Orchestration is performed by an *agent manager* and a workflow orchestrator (e.g., n8n), following an event-driven model: the arrival of a new question, alert, or data update generates events that may trigger one or more agents. Results are consolidated and sent to the natural-language interaction module [4, 13].

5.2.5 Natural-language interaction module. The fifth module is the interface between users and the GISSA GPT ecosystem:

- access channels (web interface, institutional chatbot, integration with existing systems);
- conversation management module (session, contextual history, turn control);
- LLM *gateway* (responsible for sending *prompts* enriched with RAG context + agent outputs and receiving responses).

This module implements policies for:

- scope control (types of questions that can be answered);
- response filtering (blocking attempts at individualized diagnosis or prescription);
- explicit limitations (warnings about decision-support nature and the need for professional validation).

5.2.6 Observability and auditing module. The sixth module concentrates observability and governance mechanisms:

- interaction *logs* (query, retrieved context, triggered agents, used sources, model versions);
- usage *dashboards* (what types of questions are asked, by which user profiles, with which sources);
- audit trails for safety, quality, and impact assessment;
- alerts for anomalous behavior (e.g., misuse outside scope).

This module is essential to treat GISSA GPT as institutional infrastructure, enabling retrospective inspection of decisions, workflow review, and policy adjustment [19, 3].

5.3 Interaction flow

In simplified terms, a typical interaction flow is as follows:

(1) **Input** — a manager or health professional formulates a natural-language question (e.g., "Which neighborhoods show the highest risk of dengue outbreaks next month?").
(2) **Classification** — the system identifies the task type (indicator query + prediction + risk explanation).
(3) **Agent orchestration** — the orchestrator triggers:
   - the Epidemiology Agent, which queries GISSA and Smart-GISSA models;
   - the Data Quality Agent, which assesses indicator reliability;
   - the RAG module, which retrieves relevant protocols and documents.

(4) **LLM synthesis** — the LLM receives agent outputs and retrieved document passages and generates a natural-language response, including:
- an explanation of risk factors;
- source and date indications;
- interpretive cautions when applicable.

(5) **Output and logging** — the response is delivered to the user, accompanied by references and, optionally, a formal record in the auditing module, including the context used to support the decision.

With this design, the LLM does not replace GISSA or Smart-GISSA; it acts as a mediation and explanation layer reinforced by specialized agents and RAG [13, 19].

Figure 3 represents the conceptual GISSA GPT architecture as a functional chain that integrates conversational interaction, orchestration of specialized agents, RAG mechanisms, an LLM-based generative module, and auditing and governance modules [13, 20]. At the entry point, the user (a manager or health professional) interacts with a conversational interface (web/chatbot), whose utterance is forwarded to a conversation manager and task classifier responsible for identifying the interaction goal. This module coordinates queries to GISSA/Smart-GISSA APIs, enabling access to epidemiological indicators, analytical models, and other preexisting resources.

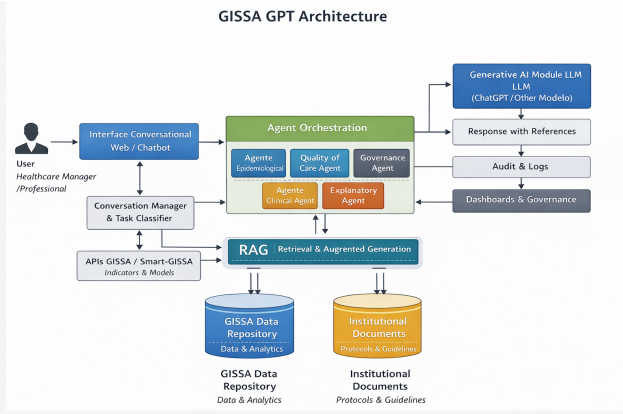

Figure 3: Conceptual GISSA GPT architecture.

Structured information is then routed to the agent orchestration module, which aggregates epidemiology, data quality, governance, and explanation agents. These agents complement processing with checks, repository queries, contextualization, and justifications.

The RAG module queries two categories of sources: (i) the GISSA data repository, containing data and analyses; and (ii) institutional documents, such as guidelines, protocols, and norms. The consolidated result is forwarded to the generative LLM, which produces the final answer associated with references or retrieved documentary excerpts [13, 20].

The output is submitted to auditing, *logging*, and traceability modules, enabling retrospective inspection, and it can feed *dashboards* and governance mechanisms oriented to decision support and digital surveillance [19, 3].

6 Concluding remarks

This work presented *GISSA GPT*, an agent-oriented architecture for governance and decision support in digital health, built on the GISSA/Smart-GISSA ecosystem and inspired by lessons from the Giselle Saúde project [9, 16]. The proposal starts from the diagnosis that LLMs, while useful as a conversational mediation layer, are not sufficient to sustain decisions in regulated domains without additional modules for governance, traceability, and institutional integration [18].

Conceptually, GISSA GPT contributes by treating virtual assistants not as isolated interfaces, but as components of an institutional decision-support infrastructure coupled to governed data, validated predictive models, and normative documents [19, 18]. The combination of a data module (GISSA), analytical services (Smart-GISSA), RAG, specialized agents, and observability mechanisms forms an architectural arrangement that addresses gaps identified in the literature, such as lack of audit trails, difficulty of operational integration, and low explainability for managers and oversight bodies [18, 13].

Technologically, the proposed architecture describes how LLMs can be repositioned from autonomous answer generators to orchestrators of evidence and explanations grounded in verifiable sources. By requiring that each response be anchored in institutionally recognized indicators, models, and documents, GISSA GPT shifts the value axis of natural-language interaction toward the ability to support governance processes in digital health [13, 18].

This work is predominantly architectural and exploratory. It does not yet provide large-scale clinical or operational validation, nor systematic metrics of impact on decision quality, response time, or user trust. These limitations motivate the agenda for future work.

As follow-ups, four main directions stand out:

(1) prototyping and institutional pilots, with controlled deployment in surveillance and management environments, to evaluate performance, usability, acceptability, and effects on decision workflows;
(2) integration with Giselle Saúde, exploring scenarios in which GISSA GPT acts as a governance and explanation module for aggregated cases, while Giselle remains focused on individual mental-health interaction with clinical safeguards [16];
(3) a formal governance evaluation, including metrics of traceability, auditability, adherence to norms, and perceived risk by managers, health professionals, and oversight bodies [18];
(4) generalization of the architecture to other digital health domains and other public administration sectors, where LLMs can be combined with governed data and institutional rules [19].

In summary, GISSA GPT does not aim to replace health professionals or automate clinical decisions, but to provide an architectural reference for incorporating Generative AI responsibly into digital health infrastructure [18]. By articulating data, models, documents, and agents under a governance logic, this work seeks to contribute to anchoring the use of LLMs in public policy in requirements of transparency, accountability, and public interest [18, 19].

References

[1] Tiago Veloso Araujo, Silvio Ricardo Pires, and Paulo Bandiera-Paiva. 2014. Adoção de padrões para registro eletrônico em saúde no brasil. *RECIIS – Revista Eletrônica de Comunicação, Informação & Inovação em Saúde*, 8, 4. Retrieved Jan. 17, 2026 from https://www.reciis.icict.fiocruz.br/index.php/reciis/article/view/440.

[2] John W. Ayers, Adam Poliak, Mark Dredze, et al. 2023. Comparing physician and artificial intelligence chatbot responses to patient questions posted to a public social media forum. *JAMA Internal Medicine*, 183, 6, 589–596. doi:10.1001/jamainternmed.2023.1838.

[3] Ivana Cristina de Holanda Cunha Barreto, Kelen Gomes Ribeiro, and Luiz Odorico Monteiro de Andrade, eds. 2024. *Saúde Digital: Conceitos, Pesquisas e Desenvolvimento Tecnológico*. Editorial Casa, Curitiba, PR, Brazil. ISBN: 978-65-5216-261-8. doi:10.70271/250505.1056.

[4] Rishi Bommasani et al. 2021. On the opportunities and risks of foundation models. *arXiv*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2108.07258 arXiv: 2108.07258.

[5] Boston Children's Hospital (HealthMap). 2026. Outbreaks near me. Crowdsourced participatory surveillance for influenza and COVID-19. Retrieved Jan. 18, 2026 from https://outbreaksnearme.org/.

[6] Brasil. Ministério da Saúde. 2017. Monitoramento dos casos de dengue, febre de chikungunya e febre pelo vírus zika até a semana epidemiológica 35.

[7] Tom B. Brown et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2005.14165.

[8] Centers for Disease Control and Prevention. 2025. National syndromic surveillance program (NSSP) and the BioSense platform. Retrieved Jan. 17, 2026 from https://www.cdc.gov/nssp/php/about/about-nssp-and-the-biosense-platform.html.

[9] Raimundo Valter Costa Filho. 2021. *Smart-GISSA: um Sistema para Governança em Saúde Digital Baseado em Aprendizado de Máquina*. PhD thesis. Universidade Federal do Ceará, Fortaleza, CE, Brazil. Retrieved Jan. 17, 2026 from http://repositorio.ufc.br/handle/riufc/60257. Ph.D. dissertation.

[10] Kathleen K. Fitzpatrick, Alison Darcy, and Molly Vierhile. 2017. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): a randomized controlled trial. *JMIR Mental Health*, 4, 2, e19. doi:10.2196/mental.7785.

[11] Sebastian Garde, Petra Knaup, Evelyn J. S. Hovenga, and Sam Heard. 2007. Towards semantic interoperability for electronic health records: domain knowledge governance for openehr archetypes. *Methods of Information in Medicine*, 46, 3, 332–343.

[12] Dipak Kalra, Thomas Beale, and Sam Heard. 2005. The openehr foundation. *Studies in Health Technology and Informatics*, 115, 153–173.

[13] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Veselin Stoyanov. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/2005.11401 arXiv: 2005.11401.

[14] Adam S. Miner et al. 2016. Smartphone-based conversational agents and responses to questions about mental health, interpersonal violence, and physical health. *JAMA Internal Medicine*, 176, 5, 619–625. doi:10.1001/jamainternmed.2016.0400.

[15] A. M. B. Oliveira et al. 2021. Lariisa: soluções digitais inteligentes para apoio à saúde pública. *Ciência & Saúde Coletiva*, 26, 5369–5378. doi:10.1590/1413-81232021265.03382021.

[16] F. J. G. Sousa et al. 2024. Giselle, uma plataforma que analisa sentimentos da pessoa idosa, apoiada por inteligência artificial generativa. In *Saúde Digital: Conceitos, Pesquisas e Desenvolvimento Tecnológico*. (1st ed.). Vol. 1. Ivana Cristina de Holanda Cunha Barreto, Kelen Gomes Ribeiro, and Luiz Odorico Monteiro de Andrade, editors. Editorial Casa, Curitiba, PR, Brazil, 174–194.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*. Retrieved Jan. 17, 2026 from https://arxiv.org/abs/1706.03762.

[18] World Health Organization. 2021. *Ethics and Governance of Artificial Intelligence for Health*. World Health Organization, Geneva. Retrieved Jan. 17, 2026 from https://www.who.int/publications/i/item/9789240029200.

[19] World Health Organization. 2021. *Global Strategy on Digital Health 2020–2025*. World Health Organization, Geneva. Retrieved Jan. 17, 2026 from https://www.who.int/publications/i/item/9789240020924.

[20] F. Ye, S. Li, Y. Zhang, and L. Chen. 2024. R^2AG: incorporating retrieval information into retrieval augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 11584–11596. Retrieved Jan. 17, 2026 from https://aclanthology.org/2024.findings-emnlp.678/.

Fine-Grained Personal Data Usage Control using Blockchain, Verifiable Credentials, and IRM Technologies

Louis RAFFIN
I3S / CNRS & Docaposte
Université Côte d'Azur
Sophia Antipolis, FR, EU
louis.raffin@i3s.univ-cotedazur.fr

Karima BOUDAOUD
I3S / CNRS
Université Côte d'Azur
Sophia Antipolis, FR, EU
karima.boudaoud@i3s.univ-cotedazur.fr

Yves ROUDIER
I3S / CNRS
Université Côte d'Azur
Sophia Antipolis, FR, EU
yves.roudier@i3s.univ-cotedazur.fr

Abstract

This article extends our previous publication, which focused on the user experience of our solution for injecting identity-wallet attributes into dynamic digital documents. Based on the feedback collected, we now shift the emphasis toward the challenges of access control and usage management associated with this personal data. To address the identified limitations, our architecture has been redesigned around a decentralized model, leveraging blockchain technology to strengthen transparency, traceability, and trust between parties. This evolution builds on our initial approach and makes the enforcement of user-defined permissions consistent and verifiable across wallets. The framework shows that fine-grained and interoperable access control on identity-enriched documents is feasible.

CCS Concepts

• **Security and privacy** → *Multi-factor authentication*; **Access control**; **Digital rights management**; *Privacy-preserving protocols*; Pseudonymity, anonymity and untraceability; *Authorization*; *Social aspects of security and privacy*; **Privacy protections**; **Usability in security and privacy**.

Keywords

Personal Data, Verifiable Credentials, Identity Wallets, Blockchain, Decentralized Access Control, Usage Control, Information Rights Management (IRM), Open Digital Rights Language (ODRL), Privacy Preserving Access Control, Fine-Grained Policy Enforcement, Consent and Permission Management, Data Transparency and Traceability, GDPR Compliance, Regulatory Compliance, Auditability

ACM Reference Format:
Louis RAFFIN, Karima BOUDAOUD, and Yves ROUDIER. 2026. Fine-Grained Personal Data Usage Control using Blockchain, Verifiable Credentials, and IRM Technologies. In *Proceedings of Advance 2026 (ADVANCE'2026)*. ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

1.1 Context and motivation

Digital identity wallets and verifiable credentials (VCs) are emerging as a promising way to let individuals store and selectively disclose certified personal attributes, such as identity data, professional status, or qualifications. Compared with traditional identity management systems, these technologies aim to strengthen user control over disclosure and reduce dependence on centralized identity providers [2].

However, once attributes are disclosed and integrated into digital workflows or shared documents, users often lose control over how these data are accessed, reused, retained, or redistributed. This limitation highlights the need for fine-grained usage control mechanisms that remain enforceable and auditable beyond the initial disclosure step [14] [15].

1.2 Prior work and identified limitations

In our previous work [12], we proposed a document-centric approach to integrate personal attributes from digital identity wallets into dynamic digital documents while allowing users to attach usage permissions to the disclosed data. In that first solution, a document was created from a template containing content-control markers, and users could connect through their identity wallet to review the available attributes and select which ones they wished to inject into the document. Users were also able to define the usage conditions associated with their data before insertion into the final document.

These permissions were expressed using ODRL, which allowed us to formalize fine-grained usage constraints such as retention limits, redistribution restrictions, purpose limitation, or deletion requirements [6]. An important aspect of the approach was that several users could asynchronously contribute their own attributes to the same shared document. To protect all contributors, the system aggregated the individual policies and applied the most restrictive permissions to the resulting document. This made it possible to govern multi-party documents containing personal data from different sources within a unified usage-control framework.

In the implemented prototype, ODRL policies were translated into effective permissions enforced by the document-management platform, notably through platform-level access-control mechanisms. This demonstrated the feasibility of combining identity wallets, dynamic documents, and IRM-based enforcement to let users define how their personal data may be accessed, retained, or redistributed after disclosure. However, the architecture remained largely centralized: policy generation, translation, and enforcement depended on a single platform, providing neither cryptographic proof of compliance nor immutable audit trails. In addition, the enforcement of the most restrictive policy at the whole-document

This research project is financed by Docaposte with the support of the Association Nationale de la Recherche et de la Technologie (ANRT).
ADVANCE'2026, Florianopolis, SC-Brazil

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/XXXXXXX.XXXXXXX

level could be perceived as too coarse-grained in collaborative scenarios. These limitations motivated the decentralized architecture introduced in this paper.

2 Contributions of this paper

In this paper, we extend our previous document-centric approach by proposing a decentralized architecture for fine-grained personal data usage control. Our contribution is threefold:

(1) we identify the limitations of centralized enforcement for usage policies attached to wallet-derived attributes
(2) we propose a blockchain-based architecture combining smart contracts, encrypted off-chain storage, and ODRL policy anchoring
(3) we discuss how this design improves transparency, auditability, and trust in multi-party document workflows

3 Related work

Existing blockchain-based DRM and consent-management approaches provide useful foundations for transparency, traceability, and decentralized policy enforcement. However, they generally do not address the specific case of wallet-derived personal attributes being injected into shared dynamic documents, nor the combination of ODRL-based usage policies, multi-party contributions, and document-centric enforcement. This gap motivates the architecture proposed in this paper.

3.1 Blockchain Based DRM

Blockchain-based DRM technologies represent a significant advancement in the management of digital rights [7]. Platforms such as Custos Media Technologies and Po.et leverage blockchain to embed unique watermarks in media files and timestamp content, ensuring transparent and immutable proof of ownership. Ethereum-based smart contracts, utilized by SingularDTV and Binded, automate the enforcement of licensing agreements, providing a secure and efficient method for managing digital content.

The advantages of blockchain-based DRM include several key aspects:

(1) Decentralization: Blockchain-based DRM technologies eliminate the need for intermediaries, allowing content creators to manage their rights directly. This reduces costs and enhances transparency in the distribution process.
(2) Security and Immutability: Blockchain offers a secure and immutable way to manage digital rights, protecting content from piracy and unauthorized use. Transactions recorded on the blockchain cannot be altered, ensuring the integrity of the data.
(3) Transparency: Blockchain records are publicly accessible, allowing easy verification of ownership and usage of content. This builds trust between content creators and consumers.
(4) Interoperability: Standards like MPEG-Dash and CMAF enable different blockchain-based DRM platforms to communicate and share data effectively, facilitating integration of various DRM solutions [18].
(5) Automation via Smart Contracts: Smart contracts automate the enforcement of licensing agreements, ensuring that content is used according to the granted rights. This reduces human error and improves the efficiency of rights management.

Additionally, cloud-native DRM solutions are becoming essential for scalability and flexibility as more businesses adopt cloud-based infrastructures [5].

The future of blockchain-based DRM is promising [3]. The integration of blockchain technology into DRM systems leverages its transparency and security for managing and enforcing digital rights. As these works show, for example: [17]. Recent developments in blockchain-based DRM focus on enhancing security and user experience, with AI-enhanced systems and quantum-resistant encryption. These systems analyze usage patterns, detect anomalies, and identify potential security threats in real time. As cloud DRM and AI DRM technologies continue to evolve, they will complement blockchain-based DRM by providing scalable, flexible, and intelligent solutions for protecting digital content in an increasingly interconnected digital landscape [10].

3.2 Blockchain Consent Management

Blockchain consent management offers a decentralized, transparent, and secure way to handle user data permissions [8]. By recording every consent action on an immutable ledger, it ensures users have full visibility and control over their data. This approach enhances privacy and accountability, making it ideal for sectors like healthcare [20] [4] and IoT [13], where precise and customizable permissions are crucial. Smart contracts automate consent enforcement, reducing human error and ensuring compliance with user preferences. Unlike traditional systems, which often operate in silos and offer broad consent options, blockchain consent management provides a unified and interoperable solution with granular control over data sharing [1].

4 Challenges and Requirements for Usage Control

Beyond the construction of dynamic documents enriched with identity-wallet attributes, our work has specifically focused on defining and enforcing access-control and usage-control mechanisms tightly coupled to the personal data injected into the document. To support this, we designed a dedicated configuration interface allowing users to specify the exact permissions they wish to associate with their data. These requirements are captured by ODRL policies provided by each contributor (e.g., no redistribution, delete-before, purpose limitation) that we aggregate and enforce.

A key contribution is that permissions come from data providers, not from the document issuer. Since several users may contribute identity attributes to the same document, our solution aggregates all provided policies and applies the most restrictive permissions to the final document, a behavior that traditional mechanisms cannot achieve (the initial document is shared, and users can inject their respective data asynchronously) [6]. This protects multi-party workflows where personal data from different sources coexist. In the previous prototype, policies were enforced through file-level ACLs because platforms such as Nextcloud do not support per-section permissions. This is a platform constraint rather than a limitation of ODRL itself.

Altogether, these contributions demonstrate the feasibility of attribute-driven, user-centric access control applied directly within dynamic documents. By relying on IRM technologies and ODRL policies, the system ensures machine-interpretable governance across the document lifecycle; however, enforcing these policies via platform ACLs provides neither cryptographic evidence of compliance nor an immutable audit trail.

5 Proposed Decentralized Architecture

5.1 Limitations of the previous solution

Despite its technical strengths and end-to-end demonstration, our initial solution exhibited several limitations identified through feedback from partners and users involved in the trials. First, the architecture remained highly centralized: a single platform collected the attributes, generated the policies, and enforced the permissions, creating both a single point of failure and an operator-centric trust assumption. Several reviewers highlighted a lack of operational transparency, as users lacked verifiable proof of consistent ODRL enforcement and immutable traces of data access and use [8] [11]. Policy governance also depended entirely on the central infrastructure, particularly the translation of ODRL rules into native permissions, which made cross-system auditing difficult and hindered strong interoperability across heterogeneous environments. Finally, applying the most restrictive permission at the level of the entire document was perceived as overly coarse-grained; some users expressed the desire for enforcement to be limited to the specific sections containing their personal data, rather than affecting the entire shared document. For instance, if only a single personal attribute (e.g., date of birth) appears in a multi-party document, applying the "most restrictive policy" at the whole-document level can unnecessarily block download or sharing for non-sensitive sections contributed by others.

These observations led us to explore a new approach that decentralizes policy evaluation and auditing using distributed ledger technologies. Under this model, consent and obligations can be anchored on-chain, while smart contracts orchestrate access verification and provide execution proofs—such as timestamped deletions or recorded refusals to distribute—resulting in immutable traceability, operator-independent verifiability, and ultimately increased transparency and trust.

We distinguish between perceived transparency and cryptographic transparency. Perceived transparency refers to operator-controlled signals—UI feedback, application logs, and compliance reports—that help users understand how their data is handled. While useful for usability and trust, such signals are not tamper-evident and cannot be independently verified. Cryptographic transparency, by contrast, relies on verifiable artifacts—policy hashes, immutable on-chain events, timestamped proofs, and digital signatures—that any party can audit without trusting the platform. Our previous, centralized prototype largely offered perceived transparency through platform logs. The decentralized design adds cryptographic transparency by anchoring policy digests and access events on-chain.

5.2 Decentralized Model and Smart Contract Design

The new solution is built on an Ethereum-based blockchain and smart contracts written in Solidity, with the goal of shifting trust away from our centralized platform and ensuring verifiable, trustless traceability for all accesses to personal data. As in our previous approach, the user begins by selecting certain information derived from verifiable credentials stored in their digital identity wallet. They then associate this data with specific permissions, authorization rules, the Ethereum addresses allowed to access the data, and corresponding expiration dates. Up to this point, the workflow remains similar to the initial solution. For this proof of concept, we chose Ethereum and Solidity simply because of their mature tooling, extensive documentation, and ease of prototyping. This choice was practical rather than conceptual: the proposed architecture does not rely on any Ethereum-specific feature, and the same design could be implemented on other smart-contract platforms (e.g., Hyperledger Fabric, Substrate, Tezos) as long as they support policy anchoring, event logging, and basic on-chain verification.

The major difference with previous work emerges when the user interacts with a smart contract—through a decentralized application (dApp) built with ethers.js—to register the set of rules and data definitions on the blockchain. The contract then records the necessary metadata along with the list of authorized Ethereum addresses.

When a third party wishes to access this information, they must invoke the smart contract themselves. The contract automatically checks whether the requester's address is on the user's allowlist. If the access conditions are met, the contract returns the corresponding data reference, an on-chain event records requester, data reference, and timestamp. [16] [21].

Data are encrypted client-side with a symmetric content key (e.g., AES-GCM 256). For each authorized Ethereum address, the content key is envelope-encrypted (e.g., ECIES/X25519) and attached as a per-recipient key blob [19]. Revocation removes an address from the on-chain allowlist and the resource is re-published under a new content key; only current recipients receive the updated envelope. Rotation follows the same pattern, issuing a fresh content key and new envelopes.

ODRL policies are hashed and referenced on-chain for integrity and auditability. The smart contract's permissions are based on this ODRL file. On-chain logic enforces identity-based access checks (allowlist, expiry) and emits immutable logs. Obligations that require side-effects (e.g., delete-before) are executed off-chain by trusted services; proof artifacts (time-stamped reports, hashes) are anchored on-chain via events.

The user who originally shared the information thus benefits from a decentralized, immutable, and continuously accessible log, fully independent of the platform retrieving the data. This guarantees that every access to their personal data is transparently recorded, and that no intermediary platform can alter or conceal these records. Through this mechanism, trust is shifted to the blockchain infrastructure itself, providing native auditability and enabling interoperability with any system that handles personal data.

5.3 Architecture overview

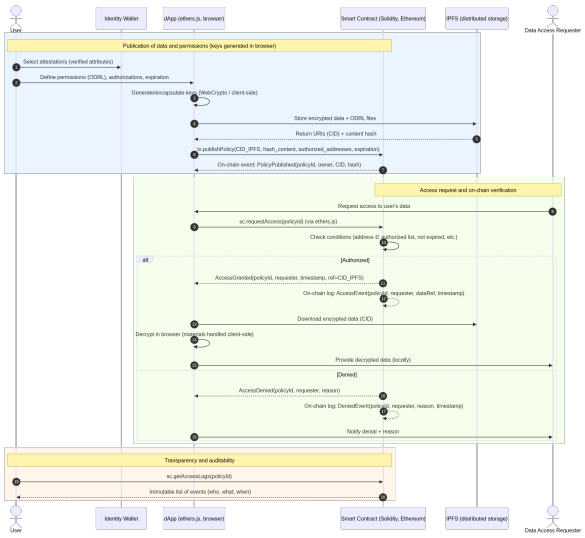

Figure 1: Sequence diagram of proposed solution

1. Data publication.

(1) The user selects their attestations in the Identity Wallet.
(2) Access rules are defined within the dApp.
(3) The dApp generates cryptographic keys and encrypts the data locally.
(4) The encrypted data is uploaded to IPFS, which returns a CID.
(5) The dApp registers the corresponding access policy on the Smart Contract.
(6) The Smart Contract confirms the publication.

2. Access Request.

(7) A Consumer requests access through the dApp.
(8) The dApp queries the Smart Contract.
(9) The Smart Contract verifies whether the access conditions are satisfied.
   - **If authorized:** the Smart Contract emits *AccessGranted*; the dApp retrieves the encrypted data from IPFS, decrypts it locally, and provides the plaintext data to the Consumer.
   - **If denied:** the Smart Contract emits *AccessDenied*, and the dApp notifies the Consumer.

3. Audit.

(12) The user can retrieve immutable access logs from the Smart Contract.

5.4 Identified limitations and future work

Our work currently presents several limitations. First, no systematic evaluation of user experience (UX) or privacy impact assessment has yet been conducted to measure the cognitive load associated with configuring permissions, the level of understanding of ODRL policies, or the acceptability of on-chain mechanisms for end users. Furthermore, although the decentralized architecture has been specified and demonstrated, we have not yet reported quantitative measurements (latency, gas costs, scalability, browser-side crypto overhead, IPFS retrieval times).

Additionally, key management and the coupling between on-chain and off-chain components raise operational challenges, including revocation, rotation, and selective sharing, which still require appropriate tooling. In our next publication, we plan to conduct in-depth technical evaluations (performance benchmarks, cost assessments, robustness/scalability analyses and stress tests), a formal security analysis of the smart contract (including review, property testing, and possibly verification), as well as a lightweight UX study to evaluate how decentralization affects user trust and the comprehensibility of policies. We also intend to carry out an initial privacy-impact assessment focusing on traceability, data minimization, and residual risks.

One of the limitations of our solution stems from the fact that logs on the blockchain are public and not private (even if pseudonymous thanks to crypto wallet addresses). One solution would be to use asymmetric cryptography so that only the data owner can access the history.

6 Conclusion

In this article, we presented an approach for integrating attributes from digital identity wallets into dynamic documents, while applying usage policies expressed in ODRL. Our initial centralized solution demonstrated the feasibility of combining identity wallets, usage control mechanisms, and the automatic enforcement of permissions. However, user feedback highlighted several limitations, including a lack of transparency regarding the actual enforcement of policies, dependence on a single platform, and difficulties in auditing the circulation of personal data.

These observations prompted us to explore a decentralized architecture built on technologies such as blockchain, smart contracts, and distributed storage systems like IPFS. In this new design, wallet-derived attestations and ODRL policy files could be stored in a distributed manner, while smart contracts would orchestrate access control, ensure immutable traceability, and interpret permissions. Nevertheless, certain challenges persist—particularly those related to the encryption required for storing personal data on IPFS and the secure management of the keys needed for decryption [9].

Despite these challenges, decentralization offers a more transparent and trustworthy model for governing usage control over identity data. Our future work will focus on strengthening cryptographic mechanisms and enabling more fine-grained rights management, with the goal of further enhancing user sovereignty within document-centric workflows.

Acknowledgments

We would like to express our gratitude to M. Jérémie BLANC (Docaposte) for his expertise, insights, feedback and invaluable support and guidance throughout this research work.

The authors used AI tools to revise the text and to correct any typos, grammatical errors, and awkward phrasing.

References

[1] Nathaniel Aldred, Luke Baal, Graeham Broda, Steven Trumble, and Qusay H. Mahmoud. 2019. Design and Implementation of a Blockchain-based Consent Management System. doi:10.48550/arXiv.1912.09882 arXiv:1912.09882.

[2] Matthias Babel, Lukas Willburger, Jonathan Lautenschlager, Fabiane Völter, Tobias Guggenberger, Marc-Fabian Körner, Johannes Sedlmeir, Jens Strüker, and Nils Urbach. 2025. Self-sovereign identity and digital wallets. *Electronic Markets* 35, 1 (April 2025), 28. doi:10.1007/s12525-025-00772-0

[3] Aytaj Badirova, Shirin Dabbaghi, Faraz Fatemi Moghaddam, Philipp Wieder, and Ramin Yahyapour. 2023. A Survey on Identity and Access Management for Cross-Domain Dynamic Users: Issues, Solutions, and Challenges. *IEEE Access* 11 (2023), 61660–61679. doi:10.1109/ACCESS.2023.3279492 Conference Name: IEEE Access.

[4] Philippe Genestier, Sajida Zouarhi, Pascal Limeux, David Excoffier, Alain Prola, Stephane Sandon, and Jean-Marc Temerson. 2017. Blockchain for Consent Management in the eHealth Environment: A Nugget for Privacy and Security Challenges. *Journal of the International Society for Telemedicine and eHealth* 5 (April 2017), (GKR);e24:(1–4). https://journals.ukzn.ac.za/index.php/JISfTeH/article/view/269

[5] Lewis Golightly, Paolo Modesti, Rémi Garcia, and Victor Chang. 2023. Securing distributed systems: A survey on access control techniques for cloud, blockchain, IoT and SDN. *Cyber Security and Applications* 1 (Dec. 2023), 100015. doi:10.1016/j.csa.2023.100015

[6] Ali Hariri, Amjad Ibrahim, Bithin Alangot, Subhajit Bandopadhyay, Antonio La Marra, Alessandro Rosetti, Hussein Joumaa, and Theo Dimitrakos. 2023. UCON+: Comprehensive Model, Architecture and Implementation for Usage Control and Continuous Authorization. In *Collaborative Approaches for Cyber Security in Cyber-Physical Systems*, Theo Dimitrakos, Javier Lopez, and Fabio Martinelli (Eds.). Springer International Publishing, Cham, 209–226. doi:10.1007/978-3-031-16088-2_10

[7] Sanjay Kumar Jena, Ram Chandra Barik, and Rojalina Priyadarshini. 2024. A systematic state-of-art review on digital identity challenges with solutions using conjugation of IOT and blockchain in healthcare. *Internet of Things* 25 (April 2024), 101111. doi:10.1016/j.iot.2024.101111

[8] Prasanth Varma Kakarlapudi and Qusay H. Mahmoud. 2021. A Systematic Review of Blockchain for Consent Management. *Healthcare* 9, 2 (Feb. 2021), 137. doi:10.3390/healthcare9020137 Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[9] Thomas Katsantas, Yannis Thomas, Christos Karapapas, and George Xylomenos. 2024. Enhancing IPFS privacy through triple hashing. In *2024 IEEE Symposium on Computers and Communications (ISCC)*. 1–6. doi:10.1109/ISCC61673.2024.10733729 ISSN: 2642-7389.

[10] Tauqeer Khalid, Muhammad Abbas Khan Abbasi, Maria Zuraiz, Abdul Nasir Khan, Mazhar Ali, Raja Wasim Ahmad, Joel J.P.C. Rodrigues, and Mudassar Aslam. 2021. A survey on privacy and access control schemes in fog computing. *International Journal of Communication Systems* 34, 2 (2021), e4181. doi:10.1002/dac.4181 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/dac.4181.

[11] Wei Yan Ng, Tien-En Tan, Prasanth V. H. Movva, Andrew Hao Sen Fang, Khung-Keong Yeo, Dean Ho, Fuji Shyy San Foo, Zhe Xiao, Kai Sun, Tien Yin Wong, Alex Tiong-Heng Sia, and Daniel Shu Wei Ting. 2021. Blockchain applications in health care for COVID-19 and beyond: a systematic review. *The Lancet Digital Health* 3, 12 (Dec. 2021), e819–e829. doi:10.1016/S2589-7500(21)00210-7 Publisher: Elsevier.

[12] Louis Raffin, Karima Boudaoud, and Yves Roudier. 2025. User-Centric Challenges in Digital Identity Wallets: Insights from Industry Experimentation. In *12th International Workshop on ADVANCEs in ICT Infrastructures and Services (ADVANCE 2025)*. Nice, France, 93–97. doi:10.48545/advance2025-shortpapers-4_4

[13] Konstantinos Rantos, George Drosatos, Antonios Kritsas, Christos Ilioudis, Alexandros Papanikolaou, and Adam P. Filippidis. 2019. A Blockchain-Based Platform for Consent Management of Personal Data Processing in the IoT Ecosystem. *Security and Communication Networks* 2019, 1 (2019), 1431578. doi:10.1155/2019/1431578 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2019/1431578.

[14] Sebastian Sartor, Johannes Sedlmeir, Alexander Rieger, and Tamara Roth. 2022. *Love at First Sight? A User Experience Study of Self-Sovereign Identity Wallets.*

[15] Abylay Satybaldy. 2023. Usability Evaluation of SSI Digital Wallets. In *Privacy and Identity Management*, Felix Bieker, Joachim Meyer, Sebastian Pape, Ina Schiering, and Andreas Weich (Eds.). Springer Nature Switzerland, Cham, 101–117. doi:10.1007/978-3-031-31971-6_9

[16] G. Sowmya, R. Sridevi, and K. S. Sadasiva Rao. 2026. Comprehensive Review and Analysis of Formal Verification Methods for Smart Contracts. In *Proceedings of the 7th International Conference on Communications and Cyber Physical Engineering*, Amit Kumar and Stefan Mozar (Eds.). Springer Nature, Singapore, 702–711. doi:10.1007/978-981-95-0269-1_79

[17] Shrabani Sutradhar, Sunil Karforma, Rajesh Bose, Sandip Roy, Sonia Djebali, and Debnath Bhattacharyya. 2024. Enhancing identity and access management using Hyperledger Fabric and OAuth 2.0: A block-chain-based approach for security and scalability for healthcare industry. *Internet of Things and Cyber-Physical Systems* 4 (Jan. 2024), 49–67. doi:10.1016/j.iotcps.2023.07.004

[18] Ruoyu Wang, Junjian Li, Chao Li, Naqin Zhou, and Jiahao Liu. 2023. A Trust Usage Control Approach for Media Player Based on Intel SGX. In *2023 8th International Conference on Data Science in Cyberspace (DSC)*. 533–539. doi:10.1109/DSC59305.2023.00083

[19] Zhengyu Wu, Brian Kondracki, Nick Nikiforakis, and Aruna Balasubramanian. 2024. Secrets are Forever: Characterizing Sensitive File Leaks on IPFS. In *2024 IFIP Networking Conference (IFIP Networking)*. 522–528. doi:10.23919/IFIPNetworking62109.2024.10619838 ISSN: 1861-2288.

[20] Shengmin Xu, Jianting Ning, Xiaoguo Li, Jiaming Yuan, Xinyi Huang, and Robert H. Deng. 2024. A Privacy-Preserving and Redactable Healthcare Blockchain System. *IEEE Transactions on Services Computing* 17, 2 (March 2024), 364–377. doi:10.1109/TSC.2024.3356595 Conference Name: IEEE Transactions on Services Computing.

[21] Huijuan Zhu, Lei Yang, Liangmin Wang, and Victor S. Sheng. 2024. A Survey on Security Analysis Methods of Smart Contracts. *IEEE Transactions on Services Computing* 17, 6 (Nov. 2024), 4522–4539. doi:10.1109/TSC.2024.3463394

Initial Proposal for Automating the Verification of Vulnerability Fixes in Fork-based Projects

Robson S. Santos
Universidade Federal do Ceará (UFC)
Av. Humberto Monte, s/n, Pici - CEP 60440-593
Fortaleza – CE, Brasil
robson.santos@alu.ufc.br

Lincoln S. Rocha
Universidade Federal do Ceará (UFC)
Av. Humberto Monte, s/n, Pici - CEP 60440-593
Fortaleza – CE, Brasil
lincoln@dc.ufc.br

Paulo A. L. Rego
Universidade Federal do Ceará (UFC)
Av. Humberto Monte, s/n, Pici - CEP 60440-593
Fortaleza – CE, Brasil
paulo@dc.ufc.br

Emanuel B. Rodrigues
Universidade Federal do Ceará (UFC)
Av. Humberto Monte, s/n, Pici - CEP 60440-593
Fortaleza – CE, Brasil
emanuel@dc.ufc.br

José Neuman Souza
Universidade Federal do Ceará (UFC)
Av. Humberto Monte, s/n, Pici - CEP 60440-593
Fortaleza – CE, Brasil
neuman@ufc.br

Abstract

Maintaining security in open-source derivative projects (forks) is hindered by low synchronization and the lack of automatic propagation of critical fixes from the original upstream project. Highly divergent forks accumulate independent modifications and indirect vulnerabilities, making patch identification a manual, error-prone, and risky task. This paper proposes a systematic and programmatic framework to track vulnerabilities (CVEs/CWEs), locate their corresponding fixes, and verify their presence or equivalence in forked codebases. The framework integrates automated vulnerability mining with patch similarity analysis and commit-level verification using cryptographic hashes (SHA). Preliminary results confirm that vulnerability metadata alone are often incomplete, requiring direct source-code inspection, which validates the need for a complementary verification mechanism. The expected outcome is a functional prototype and structured reports to support risk mitigation in the open-source software supply chain.

CCS Concepts

• **Security and privacy** → **Software security engineering**; • **Software and its engineering** → *Software maintenance tools*; *Software verification.*

Keywords

Software Forks. Vulnerabilities. Patch/Fix Verification. Code Propagation

1 Introduction

Security and integrity in the software supply chain are fundamental elements for the development and sustainability of modern applications [5]. Many projects critically depend on open-source software (OSS) and adopt the forking mechanism as a direct strategy for software reuse and adaptation [9]. However, fork-based development often results in divergent forks (hard forks), whose evolution occurs independently and without expectation of reintegration into the original (upstream) project [8]. This decentralized nature leads to documented inefficiencies, such as development redundancy and low code integration, which accentuates the structural divergence between projects. The metrics indicate that differences greater than 90% between the fork code and its upstream are common, reinforcing the asymmetry in code evolution [2].

This structural divergence creates a concrete security gap: critical fixes continuously introduced in the upstream project are not automatically propagated to derivative projects, requiring fork maintainers to identify and apply each relevant patch manually, a process that is costly and prone to failure, as evidenced in industrial case studies, producing complex conflicts, compilation errors, and test failures [13]. The difficulty is further compounded by the nature of the patches themselves, as structural fixes require Abstract Syntax Tree (AST) inspection and cannot be reliably detected by simple text comparison tools [1]. Beyond the source code, the risk surface is broadened by indirect vulnerabilities introduced through external dependencies [11], whose updates to patched versions do not always occur systematically. Adding to this scenario, Williams et al. [12] document that 63% of security advisories in the GHSA database and 71% of NVD entries lack patch links, confirming that metadata incompleteness is a structural property of public vulnerability databases and not an incidental limitation, making it particularly difficult to locate, let alone verify, the application of security fixes in derivative codebases.

The state of the art points to a lack of programmatic methods that provide visibility and support for decision-making to verify the application of security patches in highly divergent derivative projects [7]. Although the literature presents initiatives to automate the detection of fix commits [6], a methodological gap persists for a systematic approach that integrates three essential aspects: (i) continuous vulnerability mining (CVEs/CWEs), (ii) tracking of patches applied upstream, and (iii) programmatic verification of the existence or equivalence of these patches in the code of a highly divergent fork. Addressing this gap requires going beyond metadata and version identifiers toward direct source-code inspection, a need that becomes especially acute in forks that accumulate hundreds or thousands of unique commits, making direct merging of upstream updates structurally impractical [4].

The risk of silently missing critical patches is particularly pronounced in forks subject to licensing, architectural, or organizational constraints that prevent direct upstream integration [15]. In these scenarios, the absence of a programmatic verification mechanism means that security remediation depends entirely on manual awareness of upstream activity, a condition that scales poorly and is prone to oversight as divergence grows over time. Williams et al. [12] explicitly frame this as an open research challenge in software supply chain security, identifying the unreliable propagation of vulnerability fixes across derivative artifacts as the problem of residual and orphaned vulnerabilities.

Given these challenges, the central proposal of this work is to develop a framework capable of significantly reducing manual effort by automating the tracking and verification of the presence or equivalence of patches [16]. The framework integrates continuous vulnerability mining from widely recognized databases (NVD and GHSA), automated tracking of patch commits in the upstream repository, and a two-layer verification mechanism that combines literal cryptographic hash matching with patch similarity analysis to tolerate the structural evolution characteristic of long-lived forks.

The general objective of this article is to propose, implement, and evaluate this systematic and automated framework to track and verify vulnerability patches in open-source projects forked. The proposal aims to identify direct and indirect vulnerabilities (CVEs and CWEs) and analyze their severities according to CVSS. The central element of the proposal is the implementation of an automated verification mechanism to confirm the application or equivalence of patches in the forked project code, using patch similarity analysis and literal verification by cryptographic hash algorithms (SHA). The framework is expected to generate systematic security reports and metrics that present the patch status of each vulnerability, establishing a periodic execution strategy for continuous monitoring of the vulnerability chain and mitigating the risks arising from development divergence.

2 Related Works

Literature analysis reveals that code similarity detection and security patch traceability are crucial areas, but existing efforts tend to focus on isolated techniques rather than the systematic orchestration needed for the scenario of highly divergent forked projects. This section discusses works that address central aspects of the problem, highlighting the difference of the proposed systematic framework.

The issue of identifying vulnerable versions and tracing the origin of flaws is central, as addressed by the work *VERCATION: Precise Vulnerable Open-source Software Version Identification based on Static Analysis and LLM (2025)* [3]. This study focuses on OSS (C/C++) and uses an innovative approach, combining program slicing with large language models (LLM) to extract relevant code directly from vulnerability patches. Furthermore, it proposes clone detection based on expanded and normalized Abstract Syntax Trees (ASTs) to locate the vulnerability-introducing commit (VIC), essential for identifying the affected version range. This methodology reinforces the need for advanced structural patch analysis techniques to handle refactorings and code modifications.

In scenarios of structural divergence and software updates, Binary Code Similarity Analysis (BCSA) is fundamental. The work Enhancing Binary Code Similarity Analysis for Software Updates: A Contextual Diffing Framework (2025) [10] addresses the accurate rediscovery of functions in Cross-Version (XV) environments. The method uses version-specific call graphs and recursive neighborhood matching to improve BCSA accuracy. This technique is directly relevant to the challenge of tracking patches in binaries or source code that have undergone structural transformations, a problem that is accentuated in forks with high divergence.

The propagation of vulnerabilities in derivative and clone projects is an intrinsic concern in the free software ecosystem. The work What the Fork? Finding Hidden Code Clones in npm (2022) [14] addresses the hidden clones (shrinkwrapped clones) in the npm ecosystem that may contain vulnerabilities that have already been fixed in the original package but which escape standard auditing processes. The proposal, *UNWRAPPER*, automates the programmatic detection of these clones. This study validates the central premise of this research: security in derivative projects requires programmatic mechanisms to identify whether critical fixes have, in fact, been applied.

Table 1 highlights that existing approaches address isolated aspects of vulnerability tracking, while our proposal focuses on a systematic framework for verifying the application of security patches in highly divergent forks.

3 Proposed Approach

The proposed approach introduces a systematic and programmatic framework designed to fill a critical gap identified in the literature: the reliable verification of the application of security patches in highly divergent derivative projects (forks). While existing studies focus on isolated techniques, such as vulnerability version identification, binary similarity, or clone detection, they do not provide an integrated solution capable of tracking, validating, and contextualizing security patches across different development histories.

As illustrated in the figure 1, the architecture is composed of modular and loosely coupled components that operate as a continuous analysis pipeline. The workflow begins with the automated mining and normalization of vulnerability information from public databases, such as the CVE and CWE repositories. This step establishes the set of security issues to be tracked and allows correlation with the affected software components and versions.

Once vulnerabilities are identified, the framework performs the discovery of patch commits in the upstream repository. This process combines metadata-based strategies, such as commit messages and issue references, with structural analysis of code changes. Given that vulnerability metadata is often incomplete or inaccurate, the framework explicitly considers missing or ambiguous links between vulnerabilities and patch commits.

The main contribution of the proposed approach lies in the verification phase, where the presence or equivalence of a security patch is evaluated in the derived project. Two complementary verification strategies are employed. First, a cryptographic hash-based literal check (SHA) is applied to detect exact matches of patched code regions. Second, when the literal match fails, the framework applies

Table 1: Comparison of Related Work on Tracking and Verifying Vulnerability Fixes in Derivative Projects

Analysis Dimension	VERCATION (2025)	Binary Similarity Diff (2025)	What the Fork? (2022)	This Work
Primary unit of analysis	Patches and commit history (source code)	Binary functions across versions	Code clones in derivative packages	**Patches, commits, and versions in divergent forks**
Identification of patch-relevant code	Yes (program slicing + LLM)	Partial (binary function mapping)	No	**Yes** (structural extraction and comparison of patches)
Handling of structural divergence	Partial (AST-based normalization)	Yes (contextual diffing and call graphs)	Limited	**Explicit** (refactoring-tolerant similarity)
Explicit support for derivative projects (forks)	No	No	Yes (npm ecosystem)	**Yes** (upstream vs. highly divergent fork)
Tracking of vulnerability propagation	Yes (VIC/FIC in OSS)	No	Yes (propagation via hidden clones)	**Yes** (propagation and fixes in forks)
Programmatic verification of patch application	No	No	Partial (detection of unpatched clones)	**Yes** (automated verification of fix presence)
Integration with vulnerability databases (CVE/CWE)	Limited	No	Implicit	**Explicit and continuous** (automated mining)
Scope of the approach	Specialized technique	Specialized technique	Clone-focused tool	**Systematic and extensible framework**
Main objective	Accurate identification of vulnerable versions	Improving binary similarity for updates	Detecting hidden vulnerable clones	**Tracking and verifying vulnerability fixes in derivative projects**

a patch similarity analysis based on syntactic and structural characteristics, allowing tolerance for code transformations commonly observed in long-lived forks.

Based on the verification results, each vulnerability is classified according to its status in the forked source code, such as *patched*, *unpatched*, *equivalent patch*, or *indeterminate*. This classification provides an accurate and reproducible assessment of the security posture that goes beyond traditional version-based vulnerability tracking.

Finally, the approach allows for an iterative revalidation process. Results classified as indeterminate or inconclusive can be re-evaluated as new evidence becomes available, such as updates to the upstream, new vulnerability data, or refinements to similarity methods. Human review and evidence-based decision-making are outside the scope of this work but are considered natural extensions of the proposed architecture.

Within the scope of this study, the focus is on the definition, implementation, and validation of automated verification and tracking mechanisms. Expanding reporting to interactive dashboards, integrating with formal governance processes, and automating corrective actions are left as directions for future work.

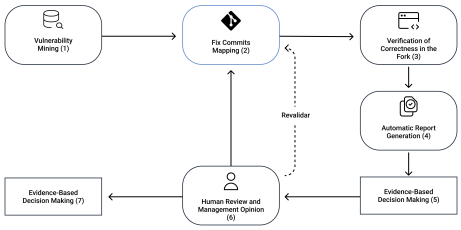

Figure 1: Overview of the proposed structure for tracking and verifying vulnerability fixes in highly divergent forks.

4 Preliminary Results and Discussion

The preliminary results obtained in the vulnerability mining process provide initial empirical evidence supporting the need for a programmatic verification mechanism for security patches in derivative projects. By characterizing the vulnerabilities, their underlying weaknesses, and the nature of patching in the upstream ecosystem, the analysis exposes structural challenges that cannot be reliably addressed solely through version-based or metadata-based approaches.

The mining process identified a total of 30 CVEs affecting the analyzed ecosystem. While the absolute number of vulnerabilities is moderate for a mature environment with multiple projects, the distribution of their severity reveals a non-trivial security exposure. Most vulnerabilities were classified as MEDIUM severity (16 cases); however a significant fraction corresponds to HIGH (9 cases) and CRITICAL (2 cases) severity, including vulnerabilities with CVSS scores as high as 9.8. This distribution indicates that, although many vulnerabilities may require contextual exploitation, several represent immediate risks to confidentiality, integrity, or availability if left unmitigated.

From a vulnerability perspective, the analysis revealed 24 distinct categories of CWEs (Common Weakness Enumeration), highlighting a heterogeneous vulnerability profile. Notably, five CWEs appear simultaneously on the CWE Top 25 and OWASP Top 10 lists (CWE-287, CWE-20, CWE-434, CWE-306, and CWE-89), underscoring the prevalence of known and high-impact security flaws. Among them, inadequate authentication (CWE-287) emerged as the most frequent vulnerability, appearing in four distinct CVEs. The recurrence of authentication, input validation, and access control problems suggests persistent challenges in applying robust trust boundaries, particularly in systems with cryptographic and distributed communication components.

These findings indicate that the ecosystem is affected not only by isolated defects but by recurring classes of vulnerabilities that are known to propagate between projects and versions. Consequently, identifying only vulnerable versions is insufficient; it becomes essential to verify whether corrective measures have been effectively applied to the derived codebases. Corroborating this observation at an ecosystem scale, Williams et al. [12] identify the unreliable propagation of fixes across derivative artifacts as a structural, not incidental, challenge in software supply chain security.

The analysis of security patches, conducted by comparing **24 patched CVEs distributed across six distinct projects**, reveals substantial variability in the scale and complexity of the patches. The number of files modified per patch ranges from 1 to 44, with a

median of 2.5 files, while the total number of lines changed ranges from just 2 to 2595 lines, with a median of 48.5 lines. This highly asymmetrical distribution, observed across SDKs, client applications, backend services, and cryptographic libraries, reflects the coexistence of small, localized patches with extensive, structurally comprehensive changes.

Small patches generally correspond to targeted fixes, such as conditional checks or parameter validation, which are better suited to literal verification methods. In contrast, large patches are indicative of deeper architectural refactorings or systemic design fixes, particularly in core SDKs and cryptographic components. These extensive changes significantly complicate the task of determining whether a fix has been propagated to a fork, since structural divergence and refactoring can obscure direct code matching.

This heterogeneity directly motivates the need for robust patch verification strategies that go beyond exact matching. In highly divergent forks, literal approaches based solely on commit hashes or file-level diffs are prone to false negatives, while version-based heuristics fail to capture partial or equivalent fixes. The observed variability in patch size and structure, therefore, reinforces the central premise of this work: effective security assurance in derived projects requires combining literal verification with an analysis that takes similarity into account, capable of tolerating refactoring and structural evolution.

Taken together, the characteristics of the vulnerabilities and the complexity of the patches highlight a critical limitation of existing vulnerability tracking practices. The presence of high-severity vulnerabilities, recurring weakness categories, and highly heterogeneous patches implies that security remediation is neither uniform nor trivially traceable throughout the project's history. In this context, forks that cannot directly incorporate upstream changes due to licensing, architectural, or organizational constraints face a greater risk of silently missing critical patches.

These preliminary findings corroborate the need for the proposed framework's main contribution: a programmatic mechanism to verify the presence or equivalence of security patches in divergent forks. By grounding the framework's design in empirical observations of real-world vulnerabilities and patches, this study establishes a concrete motivation to move beyond metadata-based tracking toward systematic, evidence-based verification.

5 Conclusion and Future Work

This article focuses on investigating the feasibility of an automated mechanism to verify the propagation and application of security patches in divergent derivative projects. To this end, a systematic framework is proposed that integrates vulnerability mining, mapping of patch commits in the upstream, and complementary verification strategies based on cryptographic hashes and patch similarity analysis. The approach was designed to mitigate limitations observed in existing vulnerability tracking practices, which often rely on incomplete metadata or insufficient version information to handle structurally divergent forks.

Within the scope of this work, the focus is on the automated verification pipeline and the generation of structured reports that allow for the systematic and reproducible assessment of the status of security patches in derivative projects. As a natural continuation of the research, future work includes expanding the verification mechanism, incorporating continuous revalidation cycles, and using these reports to support decision-making and governance of the software supply chain.

Acknowledgments

The authors would like to thank FUNCAP – Ceará Foundation for Scientific and Technological Development Support – for the support provided throughout this work.

References

[1] Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman. 2006. *Compilers: Principles, Techniques, and Tools* (2 ed.). Addison-Wesley.

[2] John Businge, Moses Openja, Sarah Nadi, and Thorsten Berger. 2022. Reuse and maintenance practices among divergent forks in three software ecosystems. *Empirical Software Engineering* 27, 2 (2022), 54.

[3] Yiran Cheng, Ting Zhang, Lwin Khin Shar, Shouguo Yang, Chaopeng Dong, David Lo, Shichao Lv, Zhiqiang Shi, and Limin Sun. 2025. VERCATION: Precise Vulnerable Open-source Software Version Identification based on Static Analysis and LLM. *IEEE Transactions on Software Engineering* (2025).

[4] Daigo Imamura, Takashi Ishio, Raula Gaikovina Kula, and Kenichi Matsumoto. 2022. Bug-fix variants: Visualizing unique source code changes across github forks. In *2022 Working Conference on Software Visualization (VISSOFT)*. IEEE, 157–161.

[5] Hamid Mohayeji, Andrei Agaronian, Eleni Constantinou, Nicola Zannone, and Alexander Serebrenik. 2025. Securing dependencies: A comprehensive study of Dependabot's impact on vulnerability mitigation. *Empirical Software Engineering* 30, 3 (2025), 89.

[6] Truong Giang Nguyen, Thanh Le-Cong, Hong Jin Kang, Xuan-Bach D Le, and David Lo. 2022. Vulcurator: a vulnerability-fixing commit detector. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 1726–1730.

[7] Luyao Ren, Shurui Zhou, Christian Kästner, and Andrzej Wąsowski. 2019. Identifying redundancies in fork-based development. In *2019 IEEE 26th International conference on software analysis, evolution and reengineering (SANER)*. IEEE, 230–241.

[8] Gregorio Robles and Jesús M González-Barahona. 2012. A comprehensive study of software forks: Dates, reasons and outcomes. In *Ifip international conference on open source systems*. Springer, 1–14.

[9] Hamzeh Eyal Salman. 2021. Feature-based insight for forks in social coding platforms. *Information and Software Technology* 140 (2021), 106679.

[10] August See, Moritz Mönnich, and Mathias Fischer. 2025. Enhancing Binary Code Similarity Analysis for Software Updates: A Contextual Diffing Framework. In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*. 1724–1740.

[11] Chungha Sung, Shuvendu K Lahiri, Mike Kaufman, Pallavi Choudhury, and Chao Wang. 2020. Towards understanding and fixing upstream merge induced conflicts in divergent forks: An industrial case study. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Practice*. 172–181.

[12] Laurie Williams, Giacomo Benedetti, Sivana Hamer, Ranindya Paramitha, Imranur Rahman, Mahzabin Tamanna, Greg Tystahl, Nusrat Zahan, Patrick Morrison, Yasemin Acar, et al. 2025. Research directions in software supply chain security. *ACM Transactions on Software Engineering and Methodology* 34, 5 (2025), 1–38.

[13] Xiaoxue Wu, Wei Zheng, Xiang Chen, Fang Wang, and Dejun Mu. 2020. CVE-assisted large-scale security bug report dataset construction method. *Journal of Systems and Software* 160 (2020), 110456.

[14] Elizabeth Wyss, Lorenzo De Carli, and Drew Davidson. 2022. What the fork? finding hidden code clones in npm. In *Proceedings of the 44th international conference on software engineering*. 2415–2426.

[15] Shurui Zhou, Bogdan Vasilescu, and Christian Kästner. 2019. What the fork: a study of inefficient and efficient forking practices in social coding. In *Proceedings of the 2019 27th ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering*. 350–361.

[16] Xin Zhou, Sicong Cao, Xiaobing Sun, and David Lo. 2025. Large language model for vulnerability detection and repair: Literature review and the road ahead. *ACM Transactions on Software Engineering and Methodology* 34, 5 (2025), 1–31.

CARAMEL: A Microservice-Based Infrastructure for Scalable Big Social Data Management

Paulo Freitas Silva Júnior
paulofreitas@macae.ufrj.br
PPGI, Universidade Federal do Rio de Janeiro (UFRJ)
Térreo, Bloco E,CCMN/NCE, Cidade Universitária -Rio de Janeiro, RJ Brazil

Tiago Cruz de França
tcruz.franca@gmail.com,
DECOMP, Universidade Federal Rural do Rio de Janeiro (UFRRJ)
Av. Gov. Roberto Silveira, S/N – Seropédica, RJ, Brazil

Jonice Oliveira
jonice@ic.ufrj.br
PPGI, Universidade Federal do Rio de Janeiro (UFRJ)
Térreo, Bloco E,CCMN/NCE, Cidade Universitária -Rio de Janeiro, RJ Brazil

Abstract

Social media platforms propagate massive volumes of semantically rich data at high speed; however, researchers often face significant technical impediments in data collection and a lack of artifact reuse. Current analysis tools generally rely on monolithic architectures that hinder collaboration, limit independent scalability, and compromise data provenance. To address these challenges, we propose CARAMEL, a microservices-based infrastructure designed for the scalable and collaborative management of Big Social Data. The framework utilizes containerization and asynchronous messaging to achieve temporal decoupling between high-speed data ingestion and multi-stage analysis. We describe the infrastructure's architecture, emphasizing its ability to integrate heterogeneous sources and automate metadata curation to ensure data quality and traceability. Evaluation through large-scale case studies conducted during the 2022 and 2024 Brazilian elections demonstrates the viability of the infrastructure, processing millions of data points across multiple social channels while maintaining resilience under high demand. The proposed CARAMEL infrastructure contributes a flexible, standards-based environment that allows research communities to share, reuse, and scale social data workflows effectively.

Keywords

Social Data Management, Workflow Management, Big Data Infrastructure, Scalability, Reproducibility

1 Introduction

Social media analysis has become increasingly relevant for building an understanding of social behaviors, market trends, and public opinions [2]. In this context, studies range from user interest identification to emergency response and the influence of social media on consumed content [4, 9]. Due to the massive engagement of citizens in these digital spaces, interaction results in the production of large volumes of semantically rich data. Nowadays, to analyze this information, one must collect, store, and manage data that varies significantly in type (text, video, image) and representation [5, 7].

In fact, online social interactions have given rise to the concept of Big Social Data (BSD)—a specialization of Big Data focused on massive, high-velocity, and heterogeneous datasets used to model human behavior [10]. However, significant technical challenges persist. Although current systems offer analysis capabilities, most tools rely on monolithic architectures that centralize all functionalities [12]. This means that components are difficult to reuse, and independent scalability is limited. For example, if a researcher needs to scale only the data collection module due to a viral event, a monolithic system often requires scaling the entire infrastructure, resulting in inefficiency.

Aside from architectural restrictions, data collection strategies are rarely reused, leading to a waste of effort and resources [7]. Collection mechanisms are often built to serve a single purpose, and the data, along with the code used to retrieve it, are not shared among researchers [13]. Consequently, despite their potential to extract valuable insights, domain experts such as sociologists or journalists face barriers to performing efficient data collection due to a lack of specific computational skills.

This paper presents CARAMEL, a microservice-based infrastructure designed for managing scalable Big Social Data ecosystems. Our proposal addresses BSD challenges from a software-oriented perspective by providing a flexible and standardized architecture. By leveraging modern paradigms such as cloud computing and microservices, CARAMEL allows researchers to customize data flows and generate real-time insights from dynamically collected data.

1.1 Relationship with Previous Work

This paper presents a significant evolution of our preliminary research discussed at the Brazilian Symposium on Information Systems (SBSI 2023) [12]. While the previous work introduced the initial concept of the CARAMEL architecture and a limited validation based solely on Twitter data from the 2022 elections, this paper expands the scope and validation in alignment with the iterative cycles of the Design Science Research (DSR) methodology. The specific novel contributions of this work include:

(1) **Multichannel Heterogeneity:** We implement and validate the orchestration of heterogeneous data sources (YouTube transcripts, chats, and video statistics), overcoming the single-source limitation of the previous study.
(2) **Complex Scenario Validation:** The architecture is stress-tested in a new, more complex scenario (2024 Municipal Elections), demonstrating the system's adaptability.
(3) **Advanced Data Pipeline:** We provide a deeper technical evaluation of the temporal decoupling mechanism and introduce a refined data transformation model for cognitive enrichment.

2 Methodology

The research challenge of establishing a software infrastructure that promotes collaboration and reuse in Big Social Data (BSD) research emerged from a real-world problem within a multidisciplinary research group. Although the group possessed technical expertise, solutions created for new projects were rarely reused, and non-technical partners faced significant barriers. Given this practical context, this research adopted the Design Science Research (DSR) method [3], which prescribes iterative cycles of problem awareness, artifact design, development, and evaluation.

2.1 Problem Awareness and Systematic Review

To rigorously ground the artifact design, the first iterative step was a Systematic Literature Review (SLR). The protocol involved searching across five major digital libraries: Science@Direct, Springer Link, ACM Digital Library, Web of Science, and IEEE Digital Library. The initial search yielded 700 studies. We applied a multi-stage filtering process:

(1) **Duplicate Removal:** 26 duplicate entries were identified and removed.
(2) **Title and Abstract Screening:** 612 articles were excluded as they did not address data ecosystem architectures or collaborative analysis.
(3) **Full-Text Analysis:** The remaining 62 articles were analyzed in depth.

Finally, 21 studies were selected for synthesis. This process revealed that while theoretical frameworks for ecosystems exist, there is a distinct lack of implemented, open-source architectures that support the complete data lifecycle with provenance, guiding the requirements for CARAMEL.

3 CARAMEL Architecture

CARAMEL is designed as a reference architecture based on microservices, messaging, and containers. The central objective is to overcome the restrictions of monolithic tools, such as the difficulty in processing large data volumes and the lack of integration capabilities.

3.1 Microservices and Container Approach

To overcome the limitations of monoliths, CARAMEL distributes system functionalities (data collection, processing, storage) into autonomous services. Each microservice is packaged in containers, ensuring that dependencies are encapsulated. This allows for consistent deployment across different computational environments, from local development to the cloud.

As illustrated in Fig. 1, the diagram follows a flow from left to right. The process begins with the Operator injecting necessary configurations. Specific microservices, such as Tweet Collectors or YouTube Collectors, use environment variables for connection and secure extraction. The flow converges to the Messaging System, which acts as a decoupling backbone. All collectors send data to a central topic, ensuring ingestion does not block processing. Subsequently, the flow bifurcates into specialized processors, such as NLP Processors (Sentiment Analysis) and persistence in a Data Lake.

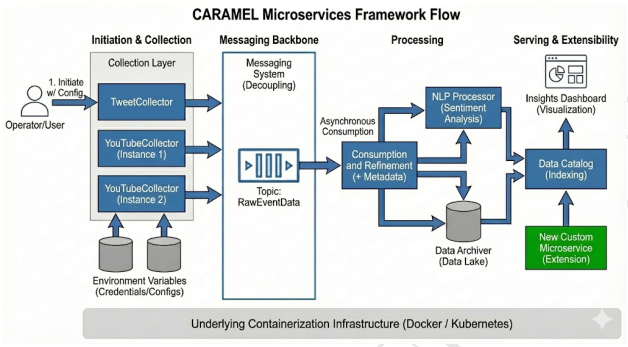

Figure 1: Architectural flow diagram of CARAMEL microservices: From Collection to Indexing.

3.2 Asynchronous Decoupling via Kafka

Messaging plays a crucial role in CARAMEL, ensuring autonomy between microservices. We utilize Apache Kafka to create a "Temporal Gap" between collection and processing.

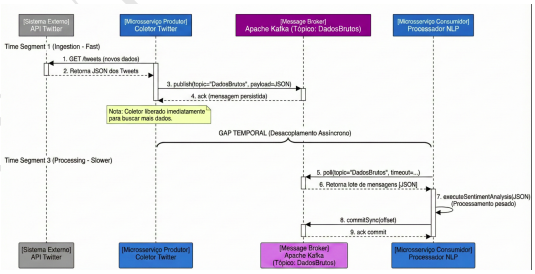

Figure 2: Sequence diagram illustrating temporal decoupling between collection and processing.

As shown in Fig. 2, the asynchronous behavior occurs in two distinct phases. In the **Production Phase**, the Collector fetches data from an external API and immediately publishes the message to the Kafka topic. Once Kafka confirms receipt (ack), the Collector is free and does not wait for data processing to complete. In the **Consumption Phase**, which occurs on demand, the Processor requests data from Kafka via polling, performs heavy processing (e.g., NLP), and only then confirms completion to Kafka. This mechanism ensured the resilience of CARAMEL in high-concurrency scenarios.

3.3 Data Transformation and Refinement

Regarding the data flow, it is important to clarify the refinement process, addressing the logical vs. physical view. In the logical architectural view, "Consumption and Refinement" represents the functional responsibility of cleaning and enriching data—such as stopword removal and tokenization. In the physical implementation, this logic is encapsulated within the *NLP Processor* microservice. When the processor consumes a message from the raw data topic, it internally executes the refinement algorithms before generating the sentiment analysis and persisting the result. This encapsulation

ensures that only high-quality, structured data is indexed into the Data Lake.

3.4 Metadata Curation and Provenance Model

A critical gap identified in BSD tools is the loss of data provenance context [12]. To address this, CARAMEL implements a strict "Schema Evolution" policy. Unlike simple scrapers that dump raw JSONs, our architecture enforces a transformation pipeline that enriches data before persistence.

Fig. 3 illustrates this transformation process. The raw data (left), often containing unstructured or irrelevant fields from the API, passes through the *Refinement Layer*.

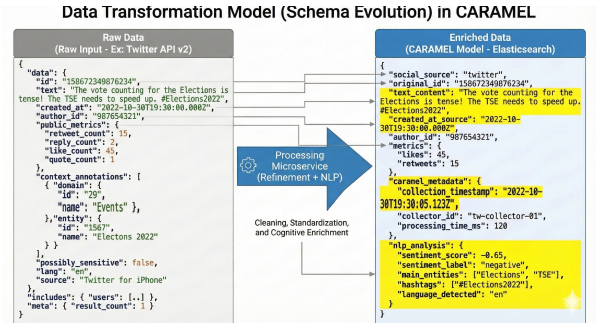

Figure 3: Data Transformation Model: Evolution from Raw API Data to Enriched Ecosystem Artifact with Provenance Metadata.

As depicted, the resulting artifact (right) is a standardized JSON injected with a specific `caramel_metadata` block. This block allows for the traceability of the data lifecycle, containing:

- **Ingestion Context:** The timestamp (`collected_at`) and the specific collector ID, allowing researchers to debug potential bias in data collection windows.
- **Processing Signature:** Tags indicating which normalization filters were applied (e.g., `filter: stop-words-removed`).
- **Cognitive Enrichment:** The injection of new semantic fields derived from the NLP microservices (e.g., sentiment scores or named entities), which were not present in the original source.

This mechanism ensures that the Data Lake contains "Smart Data" rather than just Big Data, facilitating efficient indexing in Elasticsearch and enabling complex queries based on collection context rather than just content keywords.

4 Implementation and Evaluation

The framework was implemented on Oracle Cloud infrastructure. The evaluation was structured into two major experiments to verify technical viability across different contexts (Volume vs. Heterogeneity).

4.1 Scenario 1: High-Volume Real-Time Processing (2022 Elections)

The first scenario tested CARAMEL's ingestion capacity during the 2022 Brazilian General Elections. Over 127 consecutive days, eight collectors executed simultaneous searches every minute.

4.1.1 Stream Processing Implementation. For this high-throughput scenario, we utilized **Quarkus** (a Kubernetes Native Java stack) for the microservices. Quarkus was selected for its low memory footprint and fast startup times, essential for scaling consumers dynamically. The pipeline was designed for immediate insight generation: as data flowed through Kafka, it was consumed by stream processors that performed sentiment analysis before indexing results in Elasticsearch. This enabled real-time visualization of public opinion during critical events, such as the diploma ceremony.

4.2 Scenario 2: Heterogeneity in 2024 Elections

While the 2022 scenario proved volume scalability, the 2024 Municipal Elections scenario validated the architecture's ability to handle variety and complexity. The goal was to integrate heterogeneous data sources into a single analytical workflow.

4.2.1 Fan-out Pattern for YouTube Data. We developed reusable **Python** microservices to collect three distinct data dimensions from YouTube: video statistics, transcripts, and live chat logs. A key architectural pattern applied here was the "Fan-out" messaging strategy. Instead of a monolithic script collecting everything sequentially, a single "Video Discovery" service publishes a `video_id` to a Kafka topic. This single message triggers three parallel consumer groups:

(1) **Transcript Collector:** Fetches subtitles and time-codes.
(2) **Chat Scraper:** Retrieves user interactions and timestamps.
(3) **Stats Monitor:** Logs view counts and likes.

This parallelization significantly reduced the total collection time compared to sequential processing and ensured that failure in one dimension did not block the others.

4.2.2 Results and Persistence. This orchestration generated **505 distinct datasets**, totaling 1.3 GB of raw data. The persistence layer utilized a hybrid approach: structured statistics were indexed in Elasticsearch for fast retrieval, while raw textual logs were stored in a Data Lake structure for batch NLP processing.

To consolidate the results and demonstrate the evolution of the framework's capabilities, Table 1 presents a comparative analysis of the operational metrics from both scenarios.

5 Related Work

The analysis of social data ecosystems is a broad field, yet practical engineering solutions are scarce. [8] identified that while BSD literature is extensive, information remains in silos due to a lack of shared platforms. Existing infrastructures like SoBigData [6] represent major ecosystems currently in operation. However, SoBigData relies on a large-scale, centralized infrastructure that is difficult for smaller research groups to replicate due to high operational costs. Another initiative, the DaLiF framework [11], focuses on government data lifecycles but remains conceptual in nature, lacking an open-source reference implementation.

Table 1: Comparative Analysis: Scalability (2022) vs. Heterogeneity (2024).

Metric	Scenario 1: 2022 (Presidential)	Scenario 2: 2024 (Municipal)
Goal	High-volume ingestion & stream stability.	Heterogeneous, multi-source orchestration.
Duration	127 continuous days (Aug-Dec).	Post-event targeted windows.
Sources	Single: Twitter (X) API.	Multi: YouTube (Stats, Chat, Transcripts), Instagram.
Stack	**Java (Quarkus)** for stream consumption.	**Python** for scraping & batch tasks.
Pattern	**Stream:** Serial real-time processing.	**Fan-out:** Single event triggers parallel collectors.
Throughput	≈ 1.46 million ops.	High-concurrency parallel ingestion.
Output	> 146 million records.	**505 datasets** (1.3 GB raw).
Storage	Immediate indexing (**Elasticsearch**).	Hybrid: **Data Lake** (raw) + Elasticsearch.

From a governance perspective, [1] reviewed architectures for Big Data ecosystems and concluded that existing solutions for challenges such as security and privacy are mostly focused on isolated contexts. Their work explicitly calls for a "distributed and highly capable framework" to support the data lifecycle. CARAMEL responds directly to this call by offering a lightweight, distributed, and replicable architecture based on microservices, specifically designed to foster artifact reuse and preserve provenance in diverse research communities.

6 Final Considerations

This paper addressed the significant challenges in social media data analysis related to the collection and processing of continuously generated massive data volumes. The complexity of this scenario is aggravated by the lack of integration among heterogeneous tools. To solve this, we proposed the CARAMEL framework, designed to support the creation of distributed and scalable BSD ecosystems.

The results from the 2022 and 2024 Brazilian election case studies confirmed CARAMEL's capacity to support complex analysis flows. The integration of different sources (Twitter, YouTube), combined with processing via Kafka and visualization in Elasticsearch, proved the system's flexibility in handling both real-time data and large historical volumes. The framework exemplifies a well-structured ecosystem, connecting actors through a technological infrastructure that facilitates data exchange.

6.1 Limitations

Despite promising results, this study has limitations. **Validation Scope:** Experiments occurred in controlled research environments. The absence of intensive use by an external community prevented the identification of potential orchestration bottlenecks in large-scale, highly heterogeneous ecosystems. **FAIR Integration:** While the architecture is interoperable, native integration for automated deposition into formal scientific repositories (like Dataverse) was not implemented in this version. **Usability:** Developing new components still requires a high technical profile, creating a barrier for researchers in less technical areas (e.g., journalists).

6.2 Future Work

To mitigate identified limitations, we suggest the following future research directions:

- **FAIR Repositories Integration:** Evolving the curation layer to include automatic connectors with repositories like Dataverse, ensuring datasets follow open science standards.
- **Low-Code Interfaces:** Creating abstraction layers to allow collector configuration without deep coding, democratizing ecosystem use.
- **Generative AI Integration:** Incorporating Large Language Models (LLMs) as consumers in the Kafka flow for automatic summarization and narrative discovery.

References

[1] Memoona J. ANWAR, Asif Q. GILL, Farookh K. HUSSAIN, and Muhammad IMRAN. 2021. Secure big data ecosystem architecture: challenges and solutions. *EURASIP Journal on Wireless Communications and Networking* 2021 (12 2021), 130. Issue 1. doi:10.1186/s13638-021-01996-2

[2] Umit DEMIRBAGA. 2023. HTwitt: a hadoop-based platform for analysis and visualization of streaming Twitter data. *Neural Computing and Applications* 35, 33 (11 2023), 23893–23908. doi:10.1007/s00521-021-06046-y

[3] Aline DRESCH, Daniel PACHECO Lacerda, and José Antônio VALLE ANTUNES JR. 2014. Design science research. In *Design science research: A method for science and technology advancement.* Springer, 67–102.

[4] T. C. FRANÇA. 2019. *Andare: Um Framework Para Inclusão Da Análise De Dados De Mídias Sociais No Contexto Da Preparação E Resposta À Emergência Em Situações De Manifestações De Massa.* Tese (Doutorado). Universidade Federal do Rio de Janeiro. https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=7886169

[5] T. C. FRANÇA, F. F. FARIA, F. RANGEL, Claudio M. FARIAS, and J. OLIVEIRA. 2014. Big Social Data: Princípios sobre Coleta, Tratamento e Análise de Dados Sociais. In *XXIX Simpósio Brasileiro de Banco de Dados*, Vol. 14. Ed. Porto Alegre, 38. http://www.inf.ufpr.br/sbbd-sbsc2014/sbbd/proceedings/artigos/pdfs/127.pdf

[6] Valerio GROSSI, Fosca GIANNOTTI, Dino PEDRESCHI, Paolo MANGHI, Pasquale PAGANO, and Massimiliano ASSANTE. 2021. Data science: a game changer for science and innovation. *International Journal of Data Science and Analytics* 11, 4 (5 2021), 263–278. doi:10.1007/s41060-020-00240-2

[7] Valerio GROSSI, Beatrice RAPISARDA, Fosca GIANNOTTI, and Dino PEDRESCHI. 2018. Data science at SoBigData: the European research infrastructure for social mining and big data analytics. *International Journal of Data Science and Analytics* 6 (11 2018), 205–216. Issue 3. doi:10.1007/s41060-018-0126-x

[8] Purva GROVER and Arpan Kumar KAR. 2017. Big Data Analytics: A Review on Theoretical Contributions and Tools Used in Literature. *Global Journal of Flexible Systems Management* 18, 3 (9 2017), 203–229. doi:10.1007/s40171-017-0159-3

[9] Silas P. LIMA FILHO, Jonice OLIVEIRA, and Monica Ferreira DA SILVA. 2020. Detection of Depression Symptoms using Social Media Data. *Simpósio Brasileiro de Banco de Dados (SBBD)* 2020 (2020), 3–8.

[10] Ekaterina OLSHANNIKOVA, Thomas OLSSON, Jukka HUHTAMÄKI, and Hannu KÄRKKÄINEN. 2017. Conceptualizing Big Social Data. *Journal of Big Data* 4, 1 (12 2017), 3. doi:10.1186/s40537-017-0063-x

[11] Syed Iftikhar Hussain SHAH, Vassilios PERISTERAS, and Ioannis MAGNISALIS. 2021. DaLiF: a data lifecycle framework for data-driven governments. *Journal of Big Data* 8, 1 (12 2021), 89. doi:10.1186/s40537-021-00481-3

[12] Paulo Freitas SILVA JÚNIOR, Tiago Cruz FRANÇA, and Jonice OLIVEIRA. 2023. CARAMEL: Ecosystem for Big Social Data. In *Proceedings of the XIX Brazilian Symposium on Information Systems* (New York, NY, USA), Vol. 1. ACM, 136–142. Issue 1. doi:10.1145/3592813.3592898

[13] Xiaoguang WANG, Qingyu DUAN, and Mengli LIANG. 2021. Understanding the process of data reuse: An extensive review. *Journal of the Association for Information Science and Technology* 72, 9 (9 2021), 1161–1182. doi:10.1002/asi.24483

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009

Addiction Markers in Online Betting and Casino Platforms: A Systematic Literature Review

Pierre Kouyoumdjian
I3S & Docaposte
Sophia Antipolis, France

Karima Boudaoud
I3S & CNRS
Sophia Antipolis, France

Abstract

The rapid expansion of online gambling has increased the demand for automated methods to identify problematic player behavior. While psychological research provides clinical criteria for addiction, the computational operationalization of these concepts into platform-level markers remains underutilized. This short paper presents a Systematic Literature Review (SLR) of addiction markers extracted from player tracking data in online betting and casino platforms. By analyzing 9 empirical studies selected from an initial pool of 141, we identified 22 distinct markers, ranging from traditional monetary indicators to platform-interaction signals such as canceled withdrawals and responsible gambling tool settings. Our analysis reveals a methodological evolution: while statistical models remain prevalent, recent studies increasingly leverage machine learning (e.g., Random Forest, Gradient Boosting) to predict high-risk user trajectories and behavioral transitions. The review highlights the need for more standardized definitions and evaluation practices to support the development of data-driven approaches for early detection of gambling-related harm.

CCS Concepts

• **Information systems** → **Data mining**; • **Computing methodologies** → Machine learning; • **Applied computing** → *Health care information systems.*

Keywords

Online gambling, Sports betting, Addiction detection, Machine Learning, Behavioral markers, Player behavior analysis, Risk prediction, Player Tracking Data, AI for behavioral analysis

ACM Reference Format:
Pierre Kouyoumdjian and Karima Boudaoud. 2026. Addiction Markers in Online Betting and Casino Platforms: A Systematic Literature Review. In *Proceedings of ADVANCE (ADVANCE'2026).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Online gambling has experienced a rapid global expansion in the last decade, driven by increased accessibility, mobile technologies, and personalized betting systems. While online gambling offers convenience and entertainment, it also poses significant risks, particularly the development of gambling addiction. Problematic gambling behavior is associated with severe psychological, financial, and social consequences, making early detection a critical public health challenge.

In response, both researchers and regulators have emphasized the importance of detecting addiction markers, i.e., observable behavioral, financial, or temporal indicators associated with problematic gambling. These markers are commonly used to build responsible gambling tools, player risk scores, and, in some cases, automated intervention mechanisms within online platforms. Prior research has explored a wide range of potential markers, including betting frequency, loss chasing, deposit escalation, and abnormal play patterns, often leveraging statistical analysis or machine learning techniques.

However, existing studies are fragmented across disciplines such as computer science and psychology. Traditional approaches to assessing gambling addiction are based on self-reported questionnaires and clinical interviews. These methods use questionnaires such as the DSM-5 and PGSI, which are based on self-reported psychological and social criteria. They suffer from reporting biases and limited temporal resolution. Moreover, these approaches are based on the voluntariness of the user and remain limited in terms of reliability and coverage, making their effectiveness uncertain.

To date, there is a lack of a systematic synthesis that consolidates these markers from a computer science perspective, particularly with respect to platform-level data and computational detection approaches. As a result, comparing findings across studies is difficult, and transferring proposed solutions to operational platforms remains challenging.

To address this gap, this paper presents a Systematic Literature Review of addiction markers used in online betting and casino platforms. The review is guided by the following research questions:

(1) What addiction markers have been proposed to identify problematic gambling behavior in online betting and casino platforms?
(2) How are these markers used?

This paper makes three main contributions: (i) a systematic synthesis of addiction markers proposed in previous research, (ii) a comparative analysis highlighting similarities and differences in how these markers are implemented, and (iii) the identification of current gaps and limitations surrounding these markers. By summarizing and comparing existing approaches, this review provides a clearer basis for the design and evaluation of computational methods for detecting problematic gambling behavior.

ADVANCE'2026, Florianopolis, SC-Brazil

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 Methods

This review follows established guidelines for conducting systematic literature reviews in computer science, adapted from Kitchenham and PRISMA, in order to make the selection process and comparisons between studies explicit and replicable. A systematic search was conducted across three academic databases: Springer Nature Link, Taylor & Francis and AK Journals.

These databases were selected because they cover interdisciplinary research at the intersection of behavioral sciences, gambling studies, and psychology. In particular, they list journals that publish empirical studies on gambling-related behaviors and indicators of addiction derived from platform data.

Although other databases, such as IEEE Xplore or the ACM Digital Library, contain works from a computer science perspective, they were not included in this study, as it focused primarily on behavioral markers of gambling addiction rather than purely technical or algorithmic contributions.

The search strategy combined several keywords related to online gambling platforms, addiction or problem gambling, and behavioral or player account data. An example search query:

(1) **Context:** ("online gambling" OR "online betting" OR "online casino")
(2) **Target:** ("problem gambling" OR "gambling addiction" OR "gambling disorder" OR "risk gambling")
(3) **Data Type:** ("player tracking data" OR "behavioral markers" OR "account-based data")

The search was limited to publications written in English and published between 2010 and 2025, reflecting the period in which online gambling platforms and data-driven monitoring approaches have become prominent.

The study selection process was conducted in three stages: title screening, abstract screening, and full-text review. Studies were included if they :

- focused on online betting or casino gaming platforms
- explicitly defined or used addiction-related markers derived from platform data
- focused on player data
- reported empirical or computational approaches for analyzing markers.

Studies were excluded if they :

- addressed only offline or land-based gambling
- focused exclusively on clinical data or questionnaires without using player data provided by platforms
- focused on the structure of available games/bets rather than player behavior.
- were not in English

The search results led to 141 publications in total, and the results for each database were reported as follows: Springer Nature Link (n = 81), AKJournals (n = 56) and Taylor & Francis (n = 4). After screening the titles and abstracts, 108 articles were deleted since they were off-topic. At this point, 33 studies remained for full-text screening since we could not address inclusion criteria by reading the abstract. After full-text review and quality evaluation, 12 studies were selected. Reasons for exclusion included use of questionnaires only, language barriers, lack of empirical data, or restricted access to full texts. And we deleted 3 articles because they were duplicates. Finally, after reviewing the full text, we selected 9 publications for quality assessment.

For each included study, relevant information was extracted using a structured data extraction form. The extracted attributes included: type of addiction marker, data source, detection or analysis method, and evaluation approach.

The comparative analysis was conducted by systematically comparing the studies according to these dimensions in order to identify similarities and differences. Instead of proposing a new taxonomy, we focus on how comparable markers are operationalized and evaluated in different studies

No automation tools were used for data collection. Data were extracted manually from the text and tables of the included studies. If multiple reports corresponding to a study were available, we used the most recent report and resolved any inconsistencies through discussion.

3 Results

3.1 RQ1: Addiction Markers Used in Online Betting and Casino Platforms

Across the selected studies, we identified a total of 22 distinct addiction markers derived from platform-level data. These markers are summarized in Table 1.

This indicates the number of studies in which each marker appears. Markers are grouped for clarity, for example, "number of bets" encompasses metrics such as "bets per day", "bets per session", and "average number of bets".

Monetary markers were the most frequently reported, including the amount wagered, the number of bets, and the amount deposited into the account. Other markers like night-time activity and failed deposits, appeared less consistently and were often combined with behavioral indicators rather than used in isolation.

The approach proposed in [6] is particularly noteworthy. In addition to conventional markers such as total amount wagered and number of bets, the authors consider a range of platform-interaction indicators, including the number of funding sources used by a player, the frequency of bonus searches, the number of times responsible gambling protections are disabled and the number of canceled withdrawal requests. These actions complement traditional financial markers by capturing aspects of users' self-regulation difficulties and engagement strategies. Several of these markers are also reported in [7], further supporting their relevance in characterizing problematic gambling behavior.

In addition, several studies have taken into account control indicators specific to certain platforms, such as contacting customer service, having multiple accounts, and playing unrelated types of games [9], which also reflect users' difficulties in regulating their gaming behavior. As a result, no single marker is used consistently across all studies. Depending on the author's objective, they will use certain markers and not others.

Overall, the results indicate broad consensus on the types of behavioral markers associated with problem gambling, but there is

still considerable diversity in the markers, and efforts are underway to more precisely redefine certain behaviors (particularly chasing losses).

3.2 RQ2: Usage of Addiction Markers in Online Betting and Casino Platforms

The reviewed studies primarily employ addiction markers for predictive modeling and user classification tasks.

In some studies, markers derived from behavioral, financial, and temporal data are used as input features for supervised learning models aimed at categorizing players based on their actions. Training labels are typically obtained from self-exclusion registries or survey-based annotations. These studies leverage ensemble methods such as Gradient Boosting and Random Forests, alongside unsupervised techniques like K-means, to categorize users based on risk profiles. [3, 5, 9].

In addition, several studies use addiction markers to anticipate user categorization and transitions between behavioral groups over time. [5] Rather than focusing solely on static classification, these works model how users may evolve from recreational to intensive or potentially problematic gambling profiles. Addiction markers are employed to capture early behavioral changes and temporal dynamics that signal such transitions, enabling predictive analysis of user trajectories.

Additionally, several studies focus on refining the definition of specific behaviors by proposing other definitions of complex markers such as chasing losses. Instead of relying on a single formulation, researchers experiment with multiple quantitative proxies to capture this behavior. For instance, some works introduce an across-day chasing metric based on the difference between the amount won and the amount wagered across consecutive days [2], while others define a chasing proxy using short-term financial activity patterns, such as more than three deposits within a twelve-hour window [9]. These diverse formulations illustrate ongoing efforts to empirically ground the measurement of loss chasing and highlight the exploratory nature of marker design in the literature.

4 Discussion

Our literature review highlights several important observations regarding the current use of addiction markers in online gambling research. Across the selected studies, a total of 22 markers are reported, with monetary indicators being the most prevalent. In particular, the amount wagered, the number of bets, and the amount deposited into the account consistently emerge as core markers associated with problematic gambling behavior. Additional markers, such as contacting customer service, the number of active gambling days within a given period, and deposit frequency, are also frequently considered relevant for capturing loss of control and engagement intensity.

An interesting finding is that several markers appear consistently across studies addressing different objectives, including self-exclusion analysis [1, 8], loss-chasing definition [2], and user classification [4]. These general-purpose markers suggest the existence of a common behavioral foundation underlying various manifestations of gambling-related harm. At the same time, more specialized markers are often introduced depending on the specific research goal, data availability, or modeling strategy adopted by the authors.

The reviewed literature also exhibits notable limitations. All analyzed datasets originate from European platforms. This limited domain coverage raises concerns about the generalizability of current findings to other gambling modalities and geographical contexts. Furthermore, platform operators employ heterogeneous data logging practices and feature definitions, which complicates data harmonization and poses significant challenges for cross-platform analysis and large-scale comparative studies.

Three of the nine reviewed studies rely on machine learning techniques [3, 5, 9], while the remaining works employ more traditional statistical models. This suggests that machine learning is increasingly explored, but most studies still rely on classical statistical techniques. The gradual adoption of machine learning nevertheless reflects a growing interest in this field.

Future work should explore the use of addiction markers for real-time detection and early intervention, an application that is currently absent from the reviewed literature. In addition, the development of open benchmarks and shared datasets would facilitate systematic evaluation of predictive models and improve reproducibility across studies. Finally, continued investigation of evolving markers, particularly complex behaviors such as loss chasing, is needed. Combining machine learning techniques with behavioral analysis could support more robust and theoretically grounded definitions of these markers, contributing to more reliable and actionable models of problematic gambling behavior.

5 Conclusion

This paper presented a systematic literature review of addiction markers derived from platform-level data in online betting and casino environments. The review identified 22 distinct markers, with monetary and behavioral indicators being the most frequently reported across studies. While a shared core of commonly used markers exists, their definitions and computational formulations vary substantially, particularly for complex behaviors such as loss chasing, which remains an active subject of refinement through alternative metrics and data-driven approaches.

The findings show that addiction markers are primarily used as features in predictive modeling and user classification tasks, including the anticipation of user transitions between behavioral categories. However, their application is largely limited to offline analysis, and real-time detection or intervention remains unexplored in the current literature. Moreover, the reviewed studies rely on geographically and domain-specific datasets, predominantly from European casino platforms, and exhibit heterogeneous data logging and preprocessing practices, which hinder cross-platform comparability.

In summary, addiction markers play a central role in computational studies of gambling behavior, yet their definitions, data sources, and evaluation protocols remain highly heterogeneous. Advancing toward standardized marker definitions, transparent reporting practices, and shared evaluation frameworks will be essential for improving reproducibility and enabling meaningful comparisons across studies. This would facilitate the development of

models that are not only more reliable, but also easier to validate and deploy on real gambling platforms.

ACKNOWLEDGMENTS

We would like to express our gratitude to Bertrand GOETZMANN (Docaposte) for his expertise, insights, feedback and invaluable support and guidance throughout this research work

References

[1] Michael Auer and Mark D. Griffiths. 2023. Attitude Towards Deposit Limits and Relationship with Their Account-Based Data Among a Sample of German Online Slots Players. *Journal of Gambling Studies* 39, 3 (Sept. 2023), 1319–1336. doi:10.1007/s10899-022-10155-1

[2] Michael Auer and Mark D. Griffiths. 2023. An Empirical Attempt to Operationalize Chasing Losses in Gambling Utilizing Account-Based Player Tracking Data. *Journal of Gambling Studies* 39, 4 (Dec. 2023), 1547–1561. doi:10.1007/s10899-022-10144-4

[3] Michael Auer and Mark D. Griffiths. 2023. Using Artificial Intelligence Algorithms to Predict Self-Reported Problem Gambling with Account-Based Player Data in an Online Casino Setting. *Journal of Gambling Studies* 39, 3 (Sept. 2023), 1273–1294. doi:10.1007/s10899-022-10139-1

[4] Michael Auer and Mark D. Griffiths. 2024. An Empirical Attempt to Identify Binge Gambling Utilizing Account-Based Player Tracking Data. *Addiction Research & Theory* 32, 4 (July 2024), 264–273. doi:10.1080/16066359.2023.2264763

[5] Michael Auer and Mark D. Griffiths. 2024. Predicting High-Risk Gambling Based on the First Seven Days of Gambling Activity After Registration Using Account-Based Tracking Data. *International Journal of Mental Health and Addiction* 22, 6 (Dec. 2024), 3397–3413. doi:10.1007/s11469-023-01056-4

[6] Maris Catania and Mark D. Griffiths. 2022. Applying the DSM-5 Criteria for Gambling Disorder to Online Gambling Account-Based Tracking Data: An Empirical Study Utilizing Cluster Analysis. *Journal of Gambling Studies* 38, 4 (Dec. 2022), 1289–1306. doi:10.1007/s10899-021-10080-9

[7] Paul Delfabbro, Jonathan Parke, Maris Catania, and Karim Chikh. 2024. Behavioural Markers of Harm and Their Potential in Identifying Product Risk in Online Gambling. *International Journal of Mental Health and Addiction* 22, 6 (Dec. 2024), 3451–3469. doi:10.1007/s11469-023-01060-8

[8] Simo Dragicevic, Christian Percy, Aleksandar Kudic, and Jonathan Parke. 2015. A Descriptive Analysis of Demographic and Behavioral Data from Internet Gamblers and Those Who Self-exclude from Online Gambling Platforms. *Journal of Gambling Studies* 31, 1 (March 2015), 105–132. doi:10.1007/s10899-013-9418-1

[9] Bastien Perrot, Jean-Benoit Hardouin, Elsa Thiabaud, Anaïs Saillard, Marie Grall-Bronnec, and Gaëlle Challet-Bouju. 2022. Development and Validation of a Prediction Model for Online Gambling Problems Based on Players' Account Data. *Journal of Behavioral Addictions* 11, 3 (Sept. 2022), 874–889. doi:10.1556/2006.2022.00063

A Appendix

A.1 Tableaux

Table 1: Markers of Addiction in Online Gambling

Marker	Definition / Operationalization	Reference
Session duration	Mean session length (min)	[3, 5–7]
Night-time activity	Activity between 00:00–06:00	[7]
Rapid play intensity	Inter-bet time < threshold	[7, 8]
Number of days between two sessions	Number of days between two gambling sessions	[4]
Number of gambling days	Number of days with gambling activity	[3–7, 9]
Amount of money bet	Total amount of money bet	[1–4, 7–9]
Amount of money deposit	Total amount of money deposited	[1, 3–6, 9]
Session ending with low balance	Sessions ending with a low balance	[2–4]
Amount won	Total amount of money won	[1–3, 5, 9]
Amount lost	Total amount of money lost	[3, 5]
Number of bets	Total number of bets placed	[1, 3, 5, 5, 8, 9]
Number of deposit	Number of deposits made	[2, 3, 5–7, 9]
Multiple source	Use of multiple payment sources	[6, 7]
Failed deposit	Number of failed deposit attempts	[7]
Cancel withdrawal	Number of cancelled withdrawal requests	[6]
Visit bonus page	Number of visits to bonus pages	[6, 7]
Disable protection	Number of times protection was disabled	[6, 7]
Contact support	Number of contacts with support	[6]
Loss chasing	Increased stake after losses	[2, 7, 9]
Self exclusion	Number of self-exclusion requests	[9]
Active accounts	Number of active gambling accounts	[9]
Different game	Number of different games played	[9]

Author Index