BRA-PS: A Blockchain Reference Architecture for Public Sector Citizen-Centric Applications

Sion Israel Sion *, Kaiwen Zhang, Alain April

Posted Date: 8 May 2026

doi: 10.20944/preprints202605.0472.v1

Keywords: blockchain; reference architecture; public sector; interoperability; citizen-centric services; ATAM; architecture trade-off analysis method

 doi:10.20944/preprints202605.0472.v1

Article

BRA-PS: A Blockchain Reference Architecture for Public Sector Citizen-Centric Applications

Sion Israel Sion *, Kaiwen Zhang and Alain April

Department of Software & IT, E'cole de technologie supe'rieure, Montre'al, Canada

* Correspondence: sion-israel.sion.1@ens.etsmtl.ca

Abstract

Public sector organizations face growing pressure to modernize service delivery through digitalization while ensuring transparency, interoperability, and citizen trust. Although block-chain technology offers promising capabilities for addressing these challenges, the absence of clear architectural guidelines for public sector contexts limits effective adoption. This study proposes BRA-PS, a Blockchain Reference Architecture for Public Sector Citizen-Centric Applications, developed from a real-world digitalization project in Quebec, Canada. The architecture organizes components into six layers (presentation, business, communication, smart contract, blockchain, and data) with cross-cutting concerns addressing governance, access control, security, and monitoring. A key design principle is the public-private workflow separation, which enables inter-organizational collaboration while preserving each organization's operational autonomy and data confidentiality. We validated the architecture through a case study involving a vehicle registration process between two public agencies, supported by a proof-of-concept implementation using Hyperledger Fabric. An Architecture Trade-off Analysis Method (ATAM) evaluation, conducted with a panel of five domain experts, identified six architectural risks, including IPFS confidentiality exposure and smart contract inflexibility, six non-risks, six sensitivity points, and six trade-offs across three key quality attributes: autonomy, collaboration, and functional suitability. The results confirm that BRA-PS effectively guides implementation decisions and stakeholder alignment. Practical recommendations derived from the evaluation provide actionable guidance for blockchain adoption in public sector services.

Keywords: blockchain; reference architecture; public sector; interoperability; citizen-centric services; ATAM; architecture trade-off analysis method

I. Introduction

Blockchain offers transformative potential for the public sector and promises to improve transparency, efficiency, and trust in government services [1]. As organizations increasingly adopt blockchain in their systems, clear guidelines for effective integration remain limited, creating challenges for stakeholders and architects responsible for designing solutions that leverage this technology. Traditionally, architects relied on accepted design concepts to create high-quality designs with predictable results [2]. These concepts include reference architectures, architectural design patterns, deployment patterns, tactics, and certain externally developed components such as frameworks [3].

Reference architectures play an important role in the adop-tion of new technologies by providing a common framework for their design and implementation. They allow stakeholders to develop a shared understanding, make informed decisions, and align their efforts. Adopting technology without clear architectural guidelines often leads to suboptimal implementation and underutilization of the potential of the technology. Our experience with a project involving a blockchain-based proof of concept in the public sector confronted us with this architectural problem.

In light of these considerations, this study proposes the following research question: What reference architecture can be employed to facilitate stakeholder alignment and guide blockchain

 doi:10.20944/preprints202605.0472.v1

integration in citizen-centric public sector services? To answer this question, the study pursues three research objectives: (RO1) to design a reference architecture that organizes blockchain-based public sector applications into coherent layers and cross-cutting concerns; (RO2) to validate the architecture through a real-world case study demonstrating interoperability between public agencies while preserving organizational autonomy and data privacy; and (RO3) to evaluate the architecture's trade-offs and quality attributes using the ATAM method, producing actionable recommendations for practitioners. These objectives guide the structure of the paper: Section 3 addresses RO1, Section 4 addresses RO2, and Section 5 addresses RO3.

In this paper, we propose BRA-PS, a reference architec-ture specifically designed for blockchain-based applications in the public sector. Existing blockchain reference architectures, as shown in Table I, address interoperability and security within business or industry-specific contexts but share three gaps when considered for public sector adoption: none are designed for citizen-centric service delivery, none address public sector governance and accountability requirements, and none employ structured architectural trade-off evaluation. BRA-PS addresses these gaps through three concrete design choices: a public-private workflow separation that enables inter-organizational collaboration while preserving each agency's operational autonomy; a centralized off-chain governance model suited to regulatory accountability requirements; and a citizen-facing Presentation Layer that makes blockchain states transparent to non-technical users. To validate our approach, we applied the proposed architecture to a case study involving a vehicle registration process between two public agencies. This paper makes an architectural contribution: its primary goal is to propose and evaluate a reference architecture, not to report experimental performance results. Consistent with established software architecture research practice, we rely on Kruchten's 4+1 view model for design [4] and the ATAM method for evaluation [5], which is the recognized standard for assessing architectural decisions and trade-offs [2]. Empirical results from the proof-of-concept implementa-tion, including transaction outputs and process validation, are reported in our companion paper [6].

This paper complements our previous work [6] by empha-sizing the architectural framework and offering deeper insights into the system's structure and components.

The remainder of this paper is organized as follows. Section 2 provides the background and discusses related work. In Section 3, the proposed reference architecture is introduced. Section 4 outlines the approach and details of the case study and system design. Section 5 presents the evaluation results. Finally, Section 6 discusses the findings and concludes the paper.

II. Background and Related Work

A. Reference Architecture

Reference architectures are frameworks that describe the logical structures of specific types of applications. They pro-vide a reference model aligned with one or more architectural patterns, typically supplemented by artifacts that facilitate their use [3]. The purpose of reference architecture is to standardize practices and align efforts with a common strategic vision, acting as a guide for creating coherent architectures across an organization [7].

Unlike architectural styles, which describe general types of components and relationships in a technology-agnostic way (e.g., "layered architecture" or "microservices"), reference architectures are domain-specific and often integrate multiple styles to suit particular needs. For instance, a layered architec-tural style organizes a system into presentation, business, and data layers, establishing a clear separation of responsibilities without prescribing specific components. However, a reference architecture for a web application may define these layers more concretely by specifying the components and roles within each layer, such as user interface, business, data access, and service agents. Additionally, reference architectures incorporate crosscutting concerns, such as security and communication, ensuring that these essential elements are considered throughout the design, even if they are not immediately visible in the component structure [3].

As a standardized technical framework, reference architec-ture plays an important role in aligning technology systems with strategic business objectives. Establishing guiding prin-ciples and

 doi:10.20944/preprints202605.0472.v1

best practices ensures consistency across technical implementations, and supports a cohesive approach to achieving strategic goals.

B. Blockchain Technology

Blockchain is a distributed ledger technology that promises to address fundamental issues of time and trust, thereby im-proving efficiency and reducing costs across multiple sectors. Its core features include immutability and a shared ledger, where transactional updates are verified through a consensus-based system, enabling digital interactions among multiple parties without the need for a central authority [8].

Smart contracts are fundamental to blockchain platforms, as they allow users to enforce rules in transaction processing. These self-executing protocols independently carry out the terms of an agreement, making it possible to automatically validate steps and encode conditions. On a blockchain platform, participants reach consensus on data formats within smart contracts, along with the rules governing transactions, a process that involves precisely defining rules, exploring exceptions, and setting up frameworks for dispute resolution [9].

In the public sector, blockchain's benefits extend to re-ducing economic costs, time, and automation in information exchanges between government and public-private entities. Verified data on the blockchain are secure from alteration or falsification, supporting reliable public-sector operations through distributed ledgers and pre-programmed, streamlined contracts [10].

Because blockchain establishes a shared chronological record of transactions and events across multiple entities, organizations can coordinate activities end-to-end, achieve process visibility, eliminate disputes, and build trust across organizational boundaries [11,12]. It replaces traditional human-based verification methods with secure peer-to-peer automation and cryptographic authorization, streamlining workflows, and contracts [13].

Three terms are central to this paper and are defined here for consistency. A permissioned blockchain is a distributed ledger in which participation is restricted to known, authenticated entities, as opposed to public blockchains where anyone can join anonymously. A public-private workflow refers to the separation between shared inter-organizational processes visible to all blockchain participants (public) and internal organizational processes that remain confidential (private). A citizen-centric approach denotes a service design paradigm in which the needs, expectations, and experience of the citizen drive architectural and process decisions, rather than institutional or technical constraints [14].

C. Related Work

Blockchain reference architectures aim to provide a structured framework that ensures coherence, security, and interoperability within the systems. These architectures are designed to meet domain needs while addressing technical and governance challenges unique to blockchain-based systems [15].

Existing literature shows a split between generic blockchain reference architectures and those tailored to specific sectors. Studies such as [16,17] proposed models intended for broad applications, whereas other studies have adapted architectures to address domain-specific needs.

Early frameworks such as Blockchain Solution Reference Architecture proposed by [16], focused on forming business networks, integrating members, managing transactions, and establishing governance structures. This foundational model enables organizations to incorporate blockchain into existing systems, with security and governance considerations in place.

Similarly, [17] introduced a layered architecture for segment-ing blockchain functions into infrastructure, platform, API, and user levels, thereby facilitating interoperability and modularity. Several studies have focused on specific domains of ap-plication. For example, [18] developed a microservice-based architecture for supply chains to enhance product traceability and stakeholder verification. [19] focused on crowdsourc-ing and proposed an architecture that supports distributed task management and participant incentives using blockchain principles. [20] proposed a blockchain-

 doi:10.20944/preprints202605.0472.v1

based architecture for automated, transparent compliance in cloud applications. It uses smart contracts and cryptographic proofs to ensure secure, verifiable, and efficient compliance adjustments between cloud providers and organizations. In distributed information systems, [21] designed a blockchain-native architecture that uses distributed ledger technology principles to improve availability, scalability, and security across hybrid cloud environments, with a framework built on Hyperledger Fabric to support secure, interoperable operations. Additionally, [22] introduced a patient-centric healthcare architecture that uses blockchain, cloud, and IoT to enhance semantic interoperability. This five-tier framework protects electronic health records and allows healthcare providers to share data safely. These domain-specific architectures rely on core blockchain technologies such as consensus mechanisms, smart contract execution, and inter-chain interoperability to ensure robust and reliable systems.

Table I summarizes the comparison between BRA-PS and existing blockchain reference architectures across six dimen-sions relevant to public sector citizen-centric applications.

Table I. COMPARISON OF BLOCKCHAIN REFERENCE ARCHITECTURES.

Architecture	Citizen-Centric	Public Sector	Governance Model	Privacy Mechanism	Interoperability	Evaluation Method
Viswanathan [16]	No	No	Business network governance	Member-based access	Cross-organization	Not specified
Liu [17]	No	No	Not specified	Channel-based	API-level	Not specified
Wang [18]	No	No	Supply chain governance	Role-based access	Microservice integration	Prototype
Gong [19]	Partial	No	Incentive-based	Participant-based	Task distribution	Simulation
Weber [20]	No	No	Compliance-driven	Cryptographic proofs	Cloud provider integration	Case study
Aviv [21]	No	No	Distributed	Channel-based	Hybrid cloud	Prototype
Gohar [22]	Partial	No	Patient-centric	Five-tier privacy	IoT/cloud integration	Case study
BRA-PS (ours)	**Yes**	**Yes**	**Centralized, off-chain, coercive**	**Public-private workflow separation**	**Legacy system integration**	**ATAM + PoC**

As shown in Table I, existing architectures address interop-erability and security within their respective domains but share three common limitations when considered for public sector adoption. First, none are explicitly designed for citizen-centric service delivery, where the end user is a member of the public rather than a business stakeholder. Second, none address the specific governance and accountability requirements of public sector organizations, such as regulatory compliance, democratic oversight, and multi-agency coordination. Third, evaluation methods in existing work rely primarily on prototypes or simulations, without a structured architectural assessment of trade-offs. BRA-PS addresses these gaps through a layered architecture specifically designed for public-private workflow separation, a centralized off-chain governance model suited to inter-agency coordination, and an ATAM-based evaluation that systematically identifies risks and trade-offs. A citizen-centric approach represents a paradigm shift in public service design where citizen needs and expectations drive the development process, rather than institutional constraints [14].

 doi:10.20944/preprints202605.0472.v1

III. Proposed Reference Architecture

A. Design Process and Rationale

The development of BRA-PS followed an iterative process grounded in three inputs: (1) requirements elicited from a real-world blockchain digitalization project in the Quebec public sector [6], (2) a systematic review of existing blockchain reference architectures [16–18], and (3) stakeholder consultations conducted throughout the project. Initial requirements were collected through workshops with representatives from both public agencies involved in the case study, covering functional needs (process automation, document traceability, citizen access) and quality attribute priorities (autonomy, interoperability, security). These requirements were mapped to architectural decisions using the ADD (Attribute-Driven Design) method [3], which guided the decomposition of the system into layers and the identification of cross-cutting concerns. Alternative decompositions were considered and rejected: a flat microservices structure was dismissed due to governance complexity in a multi-organization public sector context, and a two-tier client-server model was dismissed as insufficient to isolate the public blockchain workflow from private organizational processes. The resulting six-layer structure and its cross-cutting concerns were iteratively refined through the case study implementation and subsequently validated through the ATAM evaluation described in Section 5. The citizen-centric orientation of BRA-PS manifests concretely in three architectural decisions: (1) the Presentation Layer is explicitly designed to expose blockchain states in non-technical language accessible to citizens without blockchain expertise; (2) the public-private workflow separation ensures that citizens interact only with public processes, protecting organizational data while maintaining process transparency; and (3) the digital wallet mechanism gives citizens direct ownership of their identity credentials, reducing dependence on institutional intermediaries.

The purpose of reference architecture is to establish a structured foundation that bridges business objectives with technical solutions, effectively translating use cases into actionable designs [7,23]. A well-crafted architecture anticipates changes in technology and evolving requirements, minimizing business risks while providing the flexibility to adapt over time. This involves careful consideration of trade-offs between quality attributes, such as performance and security to meet the diverse needs of stakeholders [2].

The architecture presented here provides a structured framework for a blockchain-based application that emphasizes a public-private workflow, ensuring secure and efficient interactions while maintaining the confidentiality of sensitive internal processes. This architecture is divided into six distinct layers: presentation, business, communication, smart contract, blockchain, and data layers. Figure 1 illustrates the Blockchain Reference Architecture for Public Sector Citizen-Centric Applications (BRA-PS), which facilitates the integration of blockchain technology with both user-facing components and legacy systems.

 doi:10.20944/preprints202605.0472.v1

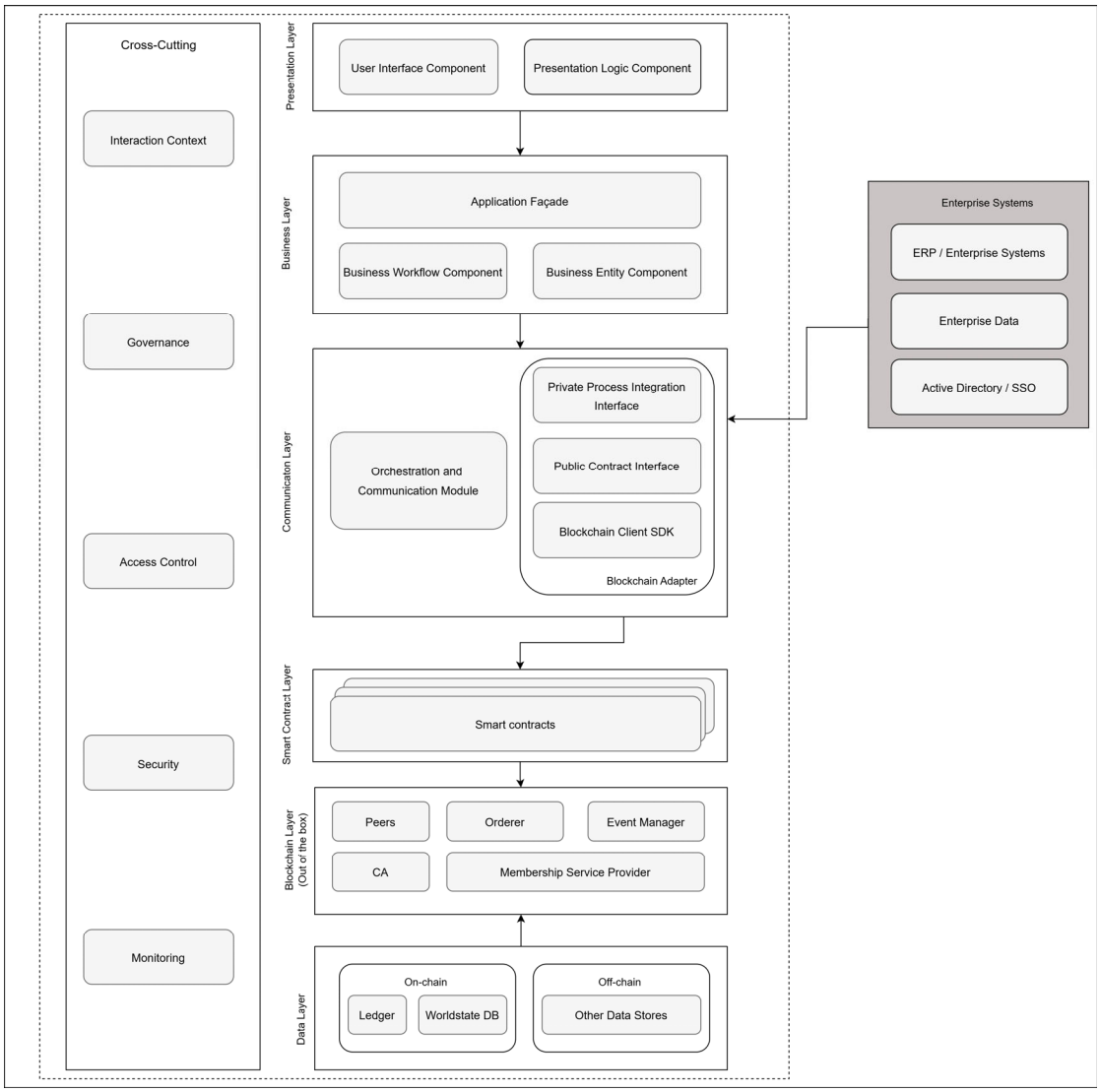

Figure 1. Blockchain Reference Architecture for Public Sector Citizen-Centric Applications (BRA-PS).

B. Presentation Layer

The Presentation Layer serves as the primary interface for users, emphasizing blockchain elements to enhance user transparency and awareness. This layer consists of two main components.

- **User Interface Component:** This component is respon-sible for rendering all interactive elements of the appli-cation, such as buttons, forms, and visualizations that showcase blockchain states. It aims to make blockchain features, such as transaction confirmations and workflow stages, clearly visible and understandable to users. This layer was introduced in response to a stakeholder requirement that citizens be able to monitor the status of their requests without requiring technical knowledge of blockchain operations (SC5, SC6).
- **Presentation Logic Component:** This component man-ages the logic that processes and formats blockchain data for the user interface. By converting raw blockchain data (e.g., transaction status and workflow stages) into user-friendly displays, this component ensures that blockchain operations are effectively communicated to the users.

 doi:10.20944/preprints202605.0472.v1

The Presentation Layer is important for improving user un-derstanding of blockchain functionality, addressing a common gap in existing blockchain applications, where transparency in blockchain processes is often minimal.

C. *Business Layer*

The Business Layer encapsulates the application's core business logic, managing workflows and entities independent of the underlying blockchain technology. This layer includes the following components:

- **Application Fac¸ade:** This component acts as a gateway for the application's business services, providing a sim-plified interface for accessing and interacting with the business logic. It abstracts the complexity of the backend processes, allowing other layers to interact simultaneously with the business services.
- **Business Workflow Component:** This component man-ages the execution of business-specific workflows, such as processing applications or approving transactions. It operates independently of the blockchain, but integrates with it to align private workflows with the public workflow exposed through the Communication Layer.
- **Business Entity Component:** This component is responsible for the management and persistence of business entities. It provides the necessary logic to manage entities, such as users, transactions, and documents, ensuring that they are handled according to business rules.

The Business Layer serves as the functional core of the application, defining and executing processes that fulfill the application's primary business objectives. Separating business logic from blockchain logic was a deliberate decision to preserve each organization's ability to evolve its internal processes independently of the shared public workflow, directly supporting the autonomy quality attribute (SC1, SC2).

D. *Communication Layer*

The Communication Layer is the most important layer for enabling a citizen-centric interorganizational architecture. It implements a public-private workflow approach, which distinguishes between two categories of processes. Public processes define the shared inter-organizational steps that are visible to all blockchain participants and executed through smart contracts, in the vehicle registration case, these include certificate issuance events and registration status transitions. Private processes remain internal to each organization and are never exposed on the blockchain, in the vehicle registration case, these include Org2's internal tax calculation logic and Org1's internal document validation procedures. This separation, grounded in inter-organizational process theory [24], allows each organization to maintain full control over its internal operations while participating in a shared, transparent public workflow. The full empirical demonstration of this approach is reported in [6]. This layer includes:

- **Orchestration and Communication Module:** This module ensures synchronization between public and private processes, managing events and signals between the blockchain and each organization's internal systems. It coordinates interactions to maintain coherence across diverse organizational workflows.
- **Blockchain Adapter:** Serving as a core interface for managing blockchain interactions, the Blockchain Adapter is designed with flexibility to adapt and evolve as needs change. It encompasses the following components:
  - **Private Process Integration Interface:** Provides a secure interface for organizations to connect their internal processes with public workflows without disclosing sensitive data, allowing private workflows to align with the shared, public framework.

 doi:10.20944/preprints202605.0472.v1

- **Public Contract Interface:** Contains the public workflow logic that operates on the blockchain, representing the shared rules and states visible to all participants, thereby supporting transparency and common understanding among stakeholders.
- **Blockchain Client SDK:** Provides essential tools and libraries for interacting with smart contracts on the blockchain, enabling the application to execute contract functions and effectively manage blockchain transactions.

This layer ensures that organizations can engage in a unified workflow, exposing only the necessary public steps while retaining control over their private operations. This layer represents the most novel contribution of the architecture: the public-private separation it enforces was identified as the critical design decision for enabling inter-organizational col-laboration without requiring organizations to expose sensitive internal processes (SC3, SC4).

E. Smart Contract Layer

The Smart Contract Layer encapsulates the smart contracts that automate the execution of business rules on the blockchain. These contracts facilitate transparency and immutability in the public workflow processes. They ensure that shared rules are consistently applied and cannot be tampered with once in place. The Smart Contract Layer is essential for enforcing autonomous public workflow rules and enhancing the reliability and transparency of the shared processes. Encapsulating shared rules in smart contracts rather than in each organization's private system was chosen to ensure that all participants operate under identical, tamper-proof conditions, addressing the automation and compliance requirements (SC10).

F. Blockchain Layer

The Blockchain Layer consists of core network components that build and maintain the blockchain infrastructure. These include network participants, known as peers, and a Certificate Authority (CA) that oversees network authentication. Both peers and the CA can be managed by network members or outsourced to third-party providers through a software-as-a-service arrangement. The use of a permissioned blockchain (Hyperledger Fabric) rather than a public chain was driven by the confidentiality and governance requirements of the public sector context, where membership control and regulatory accountability are mandatory. Network members can participate through multiple channels, which enables segmented data sharing.

These requirements are specific to the public sector context and distinguish BRA-PS from business-oriented blockchain architectures: public agencies operate under legal mandates for transparency and accountability, cannot arbitrarily restrict citizen access to process status, and must comply with data protection regulations that preclude storing sensitive citizen data on a public chain.

G. Data Layer

The Data Layer manages business-related data, facilitating the integration of both on-chain and off-chain storage solu-tions. This includes the Ledger, which records and appends transaction details in a distributed format shared across net-work participants, and the World State Database (LevelDB or CouchDB), which keeps track of the most recent values of ledger entries for efficient access. Additionally, this layer integrates decentralized storage systems for securely storing larger files and documents such as images, analytics data, and other information that complements the blockchain solution. In the public sector context, the use of a public decentralized storage network such as IPFS introduces confidentiality risks, as content stored on a public IPFS network is accessible to any participant who obtains the content identifier. BRA-PS addresses this risk through three complementary mechanisms:

(1) files are encrypted prior to upload using keys managed by the uploading organization, ensuring that content remains unreadable even if the content identifier is discovered; (2) a private

 doi:10.20944/preprints202605.0472.v1

IPFS network, deployed within the organizational perimeter and interfaced with the HLF client, restricts content accessibility to authorized network participants only; and (3) only IPFS content identifiers (hashes) are recorded on the blockchain ledger, ensuring that sensitive file content never resides on-chain. Together, these mechanisms allow BRA-PS to leverage the scalability and decentralization benefits of IPFS while meeting the confidentiality requirements of public sector data governance. The hybrid on-chain/off-chain data strategy was chosen to balance transparency, keeping document states verifiable on the ledger, with performance and cost constraints, as storing large files directly on-chain is prohibitive in a high-volume public service context.

H. Cross-Cutting Concerns

In BRA-PS, several cross-cutting concerns help support the management of blockchain applications. These key design areas influence the overall system, but are not tied to any specific layer. The main cross-cutting concerns include Interaction Context, Governance, Access Control, Security, and Monitoring.

1) *Interaction Context:* The interaction context, as pre-sented in [16], enables the identification of participants and their assigned roles. The type of participation chosen by each organization also has a significant impact on architectural design. Organizations can join the blockchain network in various capacities: larger organizations may operate a peer node, whereas smaller organizations may access services through a presentation-layer user interface provided by another organi-zation's peer node. The interaction context also determines access privileges, guiding which segments of an organization can interact directly with the blockchain network, and which segments will use intermediary interfaces. This flexibility in roles allows for a scalable and adaptable architecture that meets the diverse needs of participating entities.
2) *Governance:* Effective governance is essential for the success of blockchain solutions [9]. For permissioned block-chain solutions, where participation is restricted to authen-ticated entities, a clearly defined, published, and accessible governance framework must be established to ensure coordination and compliance among the participants. Several important considerations are essential for building a strong governance structure:
    - **Shared and Automated Processes:** Establishing agree-ments on shared and automated processes within the blockchain is essential. This includes defining the au-tomated component, method, deployment location, and mechanisms for integration with all the network mem-bers. A consistent mechanism for member participation and endorsement must also be determined to ensure smooth collaboration.
    - **Membership Management:** The governance framework must include guidelines for managing membership, en-compassing processes for inviting new members, veri-fying their identities, and announcing new participants within the network. In addition, membership approval, denial, and revocation procedures should be outlined to maintain network integrity.
    - **Network Operation and Fee Management:** The gov-ernance framework should address how new nodes are deployed, how they impact network functionality, and the structure of fees for network usage. Fee collection methods must be defined to ensure sustainable network operations.
    - **Change Management:** Processes for managing changes to membership rules, updating smart contracts, and re-leasing new versions of the application must be outlined. This includes specifying how modifications are proposed, reviewed, and implemented to maintain system stability and alignment with organizational goals.

 doi:10.20944/preprints202605.0472.v1

- **Dispute Resolution:** A clear framework for resolving disputes must be integrated into the governance model. This includes identifying the types of disputes that may arise, their relationship with smart contracts, escalation procedures, and the parties responsible for dispute medi-ation and resolution.
- **Network Regulation:** Defining the entity responsible for regulating the network is essential for ensuring account-ability. The governance model should describe how deci-sions are made and the protocols for reaching consensus on regulatory matters within the network.

In permissioned blockchain systems, these governance com-ponents are important in establishing a transparent and co-ordinated environment. An early consensus on these aspects ensures that all parties are aligned in their decision-making processes, rule enforcement, and member responsibilities, thereby providing a stable foundation for blockchain-based collaboration.

One recommended governance model is a centralized, off-chain, and coercive governance approach.

- **Centralized:** Governance decisions are managed by a well-defined group of organizations, ensuring control and accountability.
- **Off-Chain:** Major decisions are made outside the block-chain, providing flexibility in decision-making and reduc-ing on-chain overhead.
- **Coercive:** Decisions made by the governance group are implemented directly in the smart contracts, ensuring that governance rules are enforced across the network.

3) *Access Control:* In this architecture, access control is based on two key mechanisms: authentication and privacy.
   - **Authentication Mechanism:** In a blockchain network, identity is represented by digital certificates or crypto-graphic keys stored in a digital wallet. These credentials establish a participant's identity and authorize them to interact with the network.
   - **Privacy Mechanism:** Privacy is ensured by restricting visibility of transactions to only those participants registered on a particular channel. This channel-based privacy is a key element for maintaining data confidentiality because it allows sensitive information to remain accessible only to authorized entities.
4) *Security:* Security is integral to the design and operation of blockchain application. The architecture must ensure that data are protected both on- and off-chains. These include data encryption, secure key management, and secure storage. In addition, mechanisms for monitoring security events and detecting anomalies are necessary to ensure the integrity and reliability of the application.
5) *Monitoring:* Monitoring includes real-time tracking of system activities, logging of interactions and transactions, and alerting mechanisms to identify any issues that may arise. It is essential for maintaining the health and performance of blockchain networks. Continuous monitoring also enables the detection of unauthorized activities and helps ensure that the network functions as intended.

This reference architecture combines a blockchain-based public workflow with confidential private processes to ensure transparency, security, and interoperability across organizations. Each layer plays a specific role, contributing to a seamless and trustworthy ecosystem for blockchain applications.

 doi:10.20944/preprints202605.0472.v1

IV. Approach

To assess the proposed reference architecture, we conducted a structured evaluation through a case study implemented with Hyperledger Fabric (HLF), followed by an architecture-centric assessment using ATAM. The case study applies the architecture to a real-world scenario, allowing us to validate the effectiveness of the design in addressing specific requirements. The ATAM-based evaluation systematically examined architectural trade-offs and key quality attributes. ATAM was selected over alternative architectural evaluation frameworks for the following reasons. The Software Architecture Analysis Method (SAAM) [5] is suited to evaluating individual quality attributes in isolation but does not support the simultaneous analysis of multiple competing quality attributes and their trade-offs, which is central to this study given the competing demands of autonomy, collaboration, and functional suitability. The Cost Benefit Analysis Method (CBAM) extends ATAM with economic reasoning but requires quantitative cost and benefit data that were not available at this stage of the project, as the focus is architectural rather than financial. ATAM was therefore the most appropriate choice because it is specifically designed to identify trade-offs between multiple quality attributes, involves stakeholders directly in the evaluation process, and produces actionable risk and sensitivity analyses suited to guiding implementation decisions in complex multi-organization systems [2,5].

A. Case Study

This case study demonstrates the architecture's application in the context of vehicle registration within the province of Quebec, Canada, involving two primary organizations: Org1 and Org2. They work together to streamline the vehicle regis-tration process, while maintaining data integrity, transparency, and privacy. In the scenario analysed, Org1 is responsible for managing vehicle registration, whereas Org2 issues a tax certificate, which specifies the tax amount required by Org1 during registration.

The process is as follows:

- **Step 1:** A citizen purchases a vehicle and initiates the registration process by applying for a certificate from Org2. Org2 uses its internal processes to determine the appropriate tax amount and issues a certificate.
- **Step 2:** The citizen submits the certificate along with other necessary documents to Org1. Org1 processes the application, collects the tax specified in the certificate, and finalizes registration.
- **Step 3:** Org1 issues a final registration document, VDX, to the citizen, completing the registration process and authorizing the citizen to legally operate the vehicle on the roads of Quebec.

B. System Design

To support this case study, the architecture of the pro-posed solution was presented following Kruchten's "4+1" view model [4], which provides a comprehensive breakdown of the system. Each view was analyzed in detail and supported by UML diagrams.

Table II explicitly maps each step of the vehicle registration process to the corresponding BRA-PS layers and components, demonstrating how the reference architecture concretely structures the case study implementation.

1) *Logical View:* This view elaborates the Smart Contract Layer of BRA-PS, detailing how VdxPaperContract imple-ments the shared public workflow rules defined in the architec-ture. The logical view addresses the functionality of the system from an end-user perspective, focusing on smart contract operations. A smart contract plays a key role in managing data interactions on the blockchain using a unique identifier that combines a document number and the user's digital signature. It enforces function execution based on operator validation, ensuring

 doi:10.20944/preprints202605.0472.v1

compliance with document status and user identity. Only authorized entities can interact with the contract, which records all transaction outcomes for traceability purposes.

The Logical View, illustrated in Figure 2, shows the main components, including:

- **State:** Defines the unique lifecycle states of an entity.
- **StateList:** Serves as a virtual container for ledger states, minimizing transaction conflicts.
- **VdxPaper:** Represents a document in the application, extending the State class.
- **VdxPaperContract:** The primary smart contract that manages state transitions, including methods for querying history and retrieving data.

The Hyperledger Fabric (HLF) API facilitates smart contract development and provides transactional context, extending the base Contract and Context classes to integrate additional functionality, such as mapping application object IDs to composite keys in the state database.

2) *Development View:* This view maps the BRA-PS layers to concrete software components, showing how the Presentation, Business, Communication, and Blockchain layers are instantiated in the implementation. The development view provides a programmer's perspective and describes the components and modules of a systems. It outlines dependencies and organizes components according to their functional roles. Key components include:

- **User Interface (Web Browser):** Offers a user-friendly interface for interacting with the blockchain application.
- **API Service:** Connects the application to smart contracts, enabling citizen and organizational interactions with the blockchain.
- **Digital Wallet:** Manages digital identities on the blockchain, ensuring secure transactions for citizens.

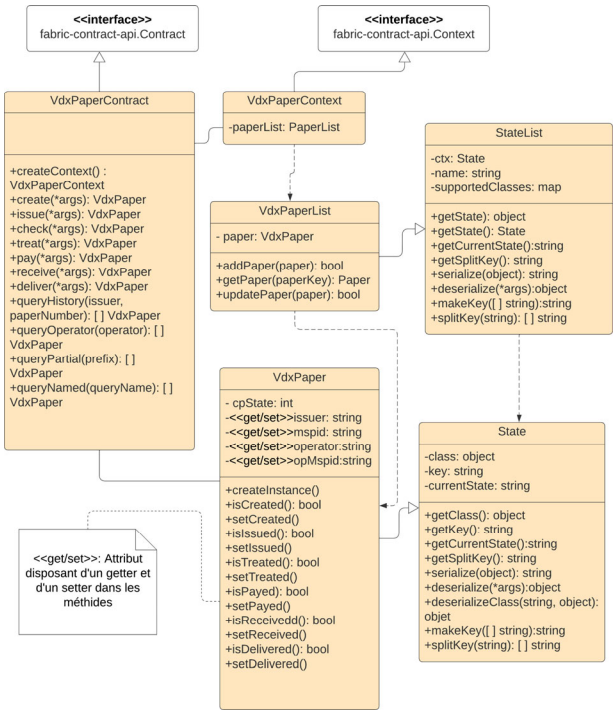

Figure 2. Logical view using the class diagram.

 doi:10.20944/preprints202605.0472.v1

- **Management System:** An organization-specific compo-nent that interfaces public blockchain processes with private workflows.
- **Blockchain (HLF):** Deploys smart contracts and stores data on public processes and document states.
- **Decentralized Storage System (IPFS):** Used for decen-tralized file management, storing files linked to public processes.

3) *Process View:* This view illustrates the runtime behavior of the public-private workflow separation central to BRA-PS, showing how citizen interactions traverse the Presentation and Communication layers while organizational processes remain within the Business layer. The process view focuses on the dynamic behavior of the system during runtime, illustrating the interactions between actors and components for both citizens and organizations. Using activity diagrams, the workflow is as follows:

- **Citizen Workflow:** As presented in Figure 3, citizens log into the platform through a web interface, authenticating with a digital wallet managed. They store data on IPFS, with files encrypted prior to uploading. The IPFS meta-data and file addresses are linked to blockchain requests to ensure secure and verifiable data management.
- **Organization Workflow:** As shown in Figure 4, organi-zation users authenticate through an enterprise system. Internal management processes interface with blockchain processes via an API service, thereby allowing seamless transition between private and public workflows. Based on the public status of a document, the system triggers further actions within the organization's private system, keeping it synchronized with the blockchain.

4) *Physical View:* This view describes how the BRA-PS Blockchain and Data layers are deployed across organiza-tional boundaries, reflecting the architecture's decentralization principle. The physical view describes the system's deploy-ment and infrastructure, emphasizing the connections between software and hardware components. As shown in Figure 5, the deployment setup included the following:

- **Client Device**: A laptop or workstation running a web browser to access the user interface[1].
- **Application Server** [2] : Hosted on Node.js, incorporat-ing HLF SDK for blockchain communication, an IPFS component for decentralized storage, and an Express component for API services [3].
- **Blockchain Network** [4]: Deployed in a series of Docker containers, each hosting elements like nodes, certificate authorities, and the state database. Docker Swarm links these containers, enabling multi-organization networks and scalable environments for public processes.

To ensure decentralization and transparency, the Hyper-ledger Fabric (HLF) network is deployed across multiple hosts[5], with each organization operating its components on separate virtual

[1] https://github.com/besionisrael/BRAPS-web

[2] https://github.com/besionisrael/BRAPS-FabricNetworkLab/tree/release-2. 2/cross-process

[3] https://documenter.getpostman.com/view/3607507/2s8YsozEmas

[4] https://github.com/besionisrael/vde23-lab/tree/release-2.2/cop-network

[5] The full artefact configuration is available at : https://github.com/ besionisrael/BRAPS-HLFMultiHostVM

 doi:10.20944/preprints202605.0472.v1

machines. This setup enables each organiza-tion to manage its own resources independently while partic-ipating in the shared network.

Table II. MAPPING OF VEHICLE REGISTRATION PROCESS TO BRA-PS LAYERS.

Step	Process Activity	BRA-PS Layer	Component
1	Citizen applies for tax certificate from Org2	Presentation	User Interface Component
1	Org2 calculates tax internally	Business	Business Workflow Component
1	Org2 issues certificate on blockchain	Smart Contract	VdxPaperContract
1	Certificate stored off-chain	Data	Decentralized Storage (IPFS)
2	Citizen submits documents to Org1	Presentation	User Interface Component
2	Org1 validates certificate via blockchain	Communication	Public Contract Interface
2	Org1 internal validation process	Business	Business Workflow Component
2	Org1 collects tax, updates blockchain state	Smart Contract	VdxPaperContract
3	Org1 issues VDX registration document	Communication	Private Process Integration Interface
3	Citizen receives and monitors status	Presentation	Presentation Logic Component

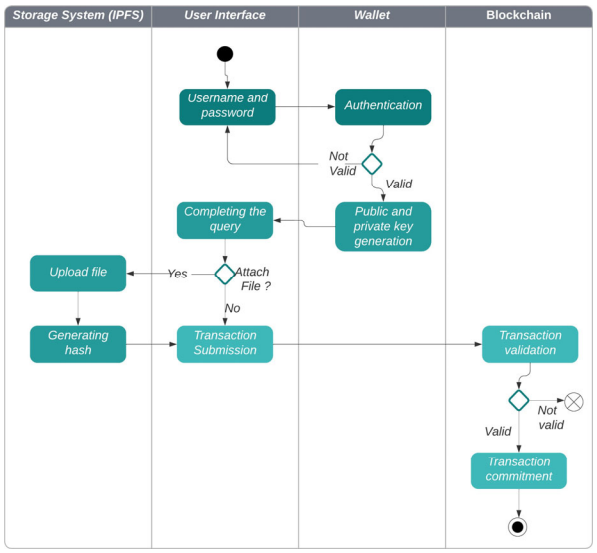

Figure 3. Activity diagram: Citizen view.

 doi:10.20944/preprints202605.0472.v1

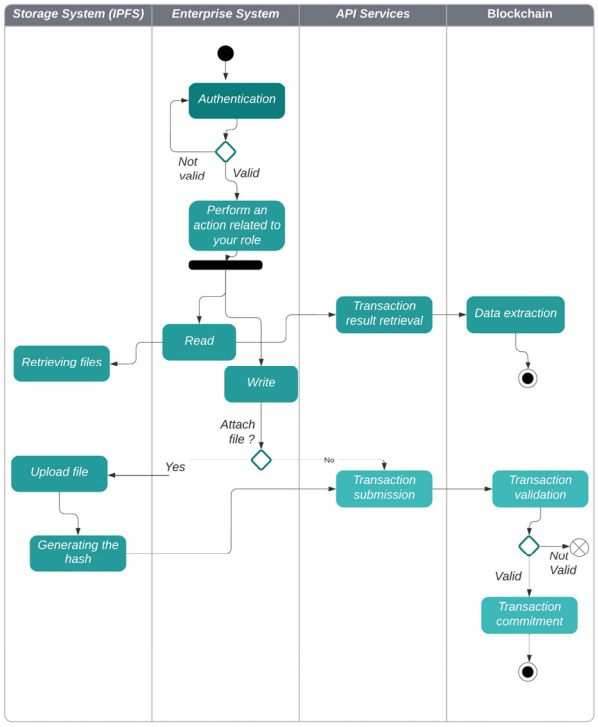

Figure 4. Activity diagram: Organization view.

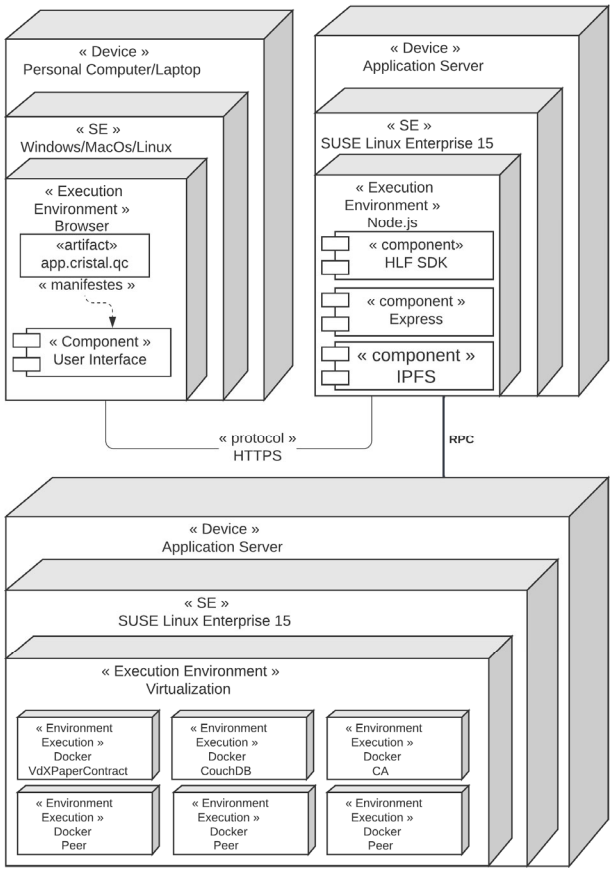

Figure 5. Physical view with deployment diagram.

 doi:10.20944/preprints202605.0472.v1

V. Evaluation

Following the system design, we evaluate the reference architecture through an architecture-centric evaluation using ATAM. This approach is appropriate for architectural contri-bution papers, where the goal is to assess the soundness of design decisions and their trade-offs rather than to measure runtime performance [2,5]. Quantitative performance results from the proof-of-concept implementation are available in [6]. This analysis assesses architectural trade-offs to identify and balance key quality attributes such as performance, security, modifiability, and scalability.

A. ATAM Evaluation

The ATAM-based evaluation provides a detailed examination of how well the architecture meets its intended goals within the constraints of the case study and identifies areas for further refinement. It involves stakeholders as a key part of the evaluation process. Table III outlines the composition of the evaluation team, and Table IV provides a summary of the steps undertaken in the ATAM evaluation process.

Table III. EVALUATION GROUP COMPOSITION.

Role	Description	Citizen Proxy
1	Information Systems Solution Architect	No
2	IT Professional experienced in block-chain applications	No
3	Project Manager and R&D Coordinator	No
4	Associate Professor and Blockchain Technology Specialist	No
5	Non-IT Specialist	Yes

Direct citizen participation in the ATAM evaluation was not feasible given the technical nature of the assessment process, which requires familiarity with architectural concepts and scenario-based reasoning. However, citizen perspectives were represented indirectly through two mechanisms. First, Participant 5, a non-IT specialist with no prior blockchain experience, served as a citizen proxy, ensuring that usability and accessibility concerns were raised during scenario elicitation. Second, the ATAM scenarios developed for functional suitability (SC5 through SC12) were derived directly from citizen-facing requirements identified during the real-world project, ensuring that the evaluation remained grounded in end-user needs. Future evaluations should consider involving citizen panels through structured co-design methods to provide more direct representation.

The utility tree serves to identify, prioritize, and refine the most important quality attribute objectives, as illustrated in Figure 6. The tree's upper-level nodes represent the key quality attributes, which in this instance are autonomy, collaboration, and functional suitability. The tree's leaves depict scenarios for achieving these respective quality attributes. Table V presents the specific scenarios developed to assess these attributes.

 doi:10.20944/preprints202605.0472.v1

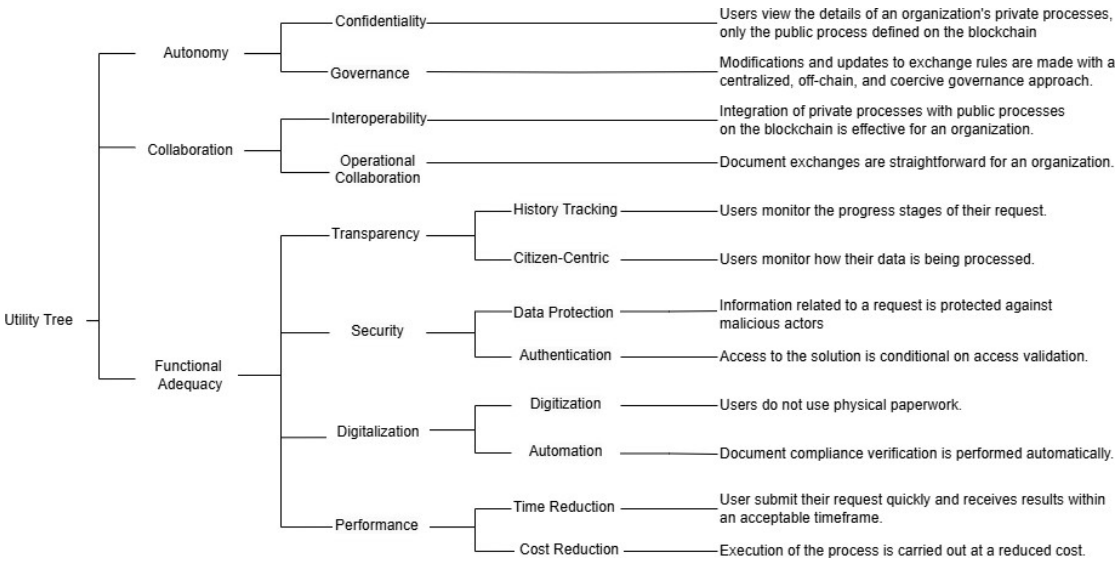

Figure 6. Utility Tree.

Table IV. SUMMARY OF ATAM ACTIVITIES.

Activity	Description
ATAM Overview	The ATAM method was introduced to the coordinator group, covering its architectural approach, analysis framework, and scenario mapping process.
Design Objectives Overview	A presentation of the design objectives was given to create a shared understanding among participants.
Architecture Presentation	The system's architecture was demonstrated using Krutchen's 4+1 model, which includes logical, de-velopment, process, and physical views, along with scenario descriptions.
Identification of Architectural Techniques	The design objectives and approaches to meet these objectives were outlined, covering: Wallet (Citizen-centric focus, Security), Channel (Privacy), World State Database (History tracking), Smart Contract (Automation, History tracking), On-chain and off-chain data (Collaboration), Integration with legacy systems (Interoperability), Blockchain (Security), Public/private key encryption (Authentication, Security), Distributed ledger (Traceability, History tracking, Collaboration), REST API (Reliability, Security).
Creation Of Quality Attribute	Based on participant questions, a utility tree for quality attributes was developed, emphasizing priority quality requirements for specific scenarios.
Utility Tree Architectural Approach Analysis	This step evaluated the architectural approaches by examining the prioritized scenarios to uncover risks, pain points, and trade-offs within the proposed architecture in relation to system interactions6.
Results Presentation	The evaluation results were shared with the partici-pants at this stage.

 doi:10.20944/preprints202605.0472.v1

Table V. ATAM ATTRIBUTES AND SCENARIOS.

Attribute	Scenarios
Autonomy	SC1: Users view only public blockchain processes, not private organizational details. SC2: Exchange rule modifications are approved on-chain by an organization.
Collaboration	SC3: Private processes are effectively integrated with public blockchain processes for an organization. SC4: Document exchanges occur in a streamlined manner for an organization.
Functional Suitability	SC5: Users can monitor the progression of their requests. SC6: Users can view how their data is being handled. SC7: Request information is protected from malicious actors. SC8: Solution access requires access validation. SC9: Process is paperless. SC10: Compliance verification is automated. SC11: Users submit requests and receive responses in an acceptable time frame. SC12: Execution costs are minimized.

B. ATAM Results

Based on the aforementioned activities, a collection of risks, non-risks, sensitivity issues, and trade-offs was compiled. Sensitivity and trade-off points refer to architectural decisions that significantly affect one or more quality attributes. In ATAM, a risk is characterized as an architectural choice that could potentially result in unfavorable outcomes based on the specified quality attribute requirements. Conversely, a non-risk is identified as an architectural decision that, following careful examination, is determined to be safe. Table VI and Table VII show the results.

Table VI. RISKS AND NON-RISKS IN THE PROPOSED ARCHITECTURE.

Risk	Non-Risk
• IPFS is a public network, typically used with a public blockchain. Traffic and content are public, posing a risk to confidentiality, even though data can be encrypted. • Use of an API service poses a security risk if the service can execute unintended commands. • Integration points risk data transformation issues if data exchange between the blockchain and existing systems is not standardized.	• Use of blockchain is a non-risk for autonomy, as it only executes public processes, separate from organi-zations' private processes. • Authentication with the X.509 certificate managed by HLF's default certificate authority is a non-risk for authentication. • The automated operation of various architecture components is a non-risk for minimal human intervention. • Public-private separation in design enhances flexibility and poses no risk.

 doi:10.20944/preprints202605.0472.v1

• A single global channel risks scalability issues if organiza-tions outside the primary use case join the channel. • The approval policy, requiring all channel participants to endorse transactions, risks scalability and performance as more participants join. • Implementing decision logic in the smart contract risks inflexibility if frequent updates are required.	• Lack of an access control list is not a risk if communication layer checks are performed. • HLF's proposed security mechanisms pose no risk if a security analysis relevant to the use case is conducted.

Table VII. SENSITIVITY AND TRADE-OFF POINTS IN THE PROPOSED ARCHITECTURE.

Sensitivity Point	**Trade-Off**
• A private blockchain ensures security, transparency, and collaboration, enabling a citizen-centered approach. • Off-chain storage enhances collaboration. • The API service connecting with the smart contract ensures organization autonomy and secure private-public system com-munication. • CouchDB as the state database impacts performance by supporting rich queries and enhancing security for compliance and data protection. • Multi-host deployment ensures decentralization and transparency. • HLF v2 allows governance mechanisms that enable organizations to agree on update procedures.	• Blockchain operations can limit organizational autonomy. • IPFS as an external network adds integration workload. • Direct citizen participation is a trade-off between transparency and performance. • Encapsulating decision logic within the smart contract code eases management but limits update flexibility. • Using a channel enforces data transaction confidentiality but requires additional collabora-tion effort. • Embedding decision logic in the smart contract reduces overall project costs but limits flexible governance.

C. Threats to Validity

Although ATAM evaluation offers important insights into the proposed reference architecture, certain factors may affect the validity of the results. We address these threats to validity in two categories: internal and external.

1) *Internal Validity:* Internal validity concerns whether the evaluation accurately reflects the assessed quality attributes. Two potential threats were identified and addressed. First, the composition of the evaluation team was carefully considered. The five participants were selected to represent distinct professional perspectives: a solution architect, a blockchain technology specialist, a project manager, an R&D coordinator, and a non-IT specialist. This disciplinary diversity was inten-tional, as participants with different backgrounds are less likely to converge on identical assessments of architectural trade-offs. Second, the risk of groupthink was mitigated through the ATAM session structure itself. Scenarios were presented individually to each participant before group discussion, allowing independent judgments to be formed prior to any

 doi:10.20944/preprints202605.0472.v1

collective deliberation. The ATAM method's utility tree con-struction further structured the elicitation process by requiring participants to independently prioritize quality attributes before reconciling differences, a procedural safeguard that surfaces disagreement rather than suppressing it [5]. The facilitator, who was not directly involved in the architecture's design, ensured that dominant voices did not disproportionately influence the outcome. Despite these precautions, the small group size remains a limitation, and replication with a larger and more geographically diverse panel is recommended for future evaluations.

2) External Validity: External validity refers to the gener-alizability of evaluation findings to other contexts. This evaluation focused on a use case involving two public agencies and citizens interacting through a single communication channel. While this approach provides valuable insights, it may not fully capture the complexity and diversity of multi-channel environments in which citizens and agencies engage through various platforms and communication methods. Expanding future evaluations to include multi-channel use cases would offer a more comprehensive understanding of the adaptability and effectiveness of the architecture across different public sector contexts. Additionally, the evaluation does not look at cases where different public organizations each have their own blockchain but need to work together. This is likely to happen as governments start using decentralized identity solutions [25].

VI. Discussions and Conclusions

Limitations

Despite the contributions of this study, several limitations should be acknowledged. Regarding internal validity, the evaluation team consisted of five participants, which, while sufficient for an ATAM assessment, limits the diversity of perspectives. Although disciplinary diversity was ensured and groupthink was mitigated through structured session protocols, a larger panel would strengthen the generalizability of the evaluation findings. Regarding external validity, the case study focused on a single use case involving two public agencies and citizens interacting through a single communication channel in the context of Quebec, Canada. This scope does not fully capture the complexity of multi-channel environments where citizens and agencies engage through various platforms, nor does it address scenarios involving multiple independent blockchains that must interoperate, which is increasingly relevant as governments adopt decentralized identity solutions [25]. Regarding scope, this paper makes an architectural contribution and deliberately does not report runtime performance benchmarks; quantitative empirical results are reported in the companion paper [6]. Finally, citizen participation in the evaluation was indirect, mediated through a non-IT specialist proxy and citizen-derived scenarios, rather than through direct involvement of citizen panels. These limitations define the boundaries of the current study and motivate the future work directions discussed at the end of this section.

Recommendations

Drawing from the ATAM results, a set of recommendations has been developed to improve the application of the proposed architecture in achieving the design objectives of this research. The identified risks serve as a basis for creating a risk mitigation plan. These recommendations target specific architectural components and can act as best practice guidelines for implementing the architecture.

- Perform a security analysis of HLF, outlining potential threats across different layers of the case study to assess how HLF's integrated functionalities can mitigate these risks.
- Implement a centralized authentication process to estab-lish uniformity across all public services in case this architecture is replicated across various use cases.

 doi:10.20944/preprints202605.0472.v1

- Design a "multi-channel" setup to enable multiple use cases involving organizations on the same network, facil-itating private and confidential transactions.
- Deploy the HLF network across multiple hosts to ensure decentralization and transparency, with each organization deploying its own virtual machines to organize its components.
- Standardize exchange formats and establish inter-service standards between the blockchain and organizations' private management systems to extend implementation within services with similar structures.
- Externalize decision logic to an external rule engine rather than embedding it in the smart contract code, providing organizations with the flexibility to modify decision logic without changing the smart contract code.
- Utilize a decentralized storage solution with a private IPFS network interfaced with an HLF client for data exchange.
- Hash data before calling a smart contract and establish a mechanism to share source data.
- Implement access control lists to restrict data access to specific roles within the smart contract logic.

Practitioner Guide: Adopting BRA-PS

The following step-by-step guide is intended to support practitioners in applying BRA-PS to a new public sector blockchain initiative.

1) **Define the interaction context.** Identify all participating organizations and their roles within the network. Determine which organizations will operate peer nodes and which will access the network through intermediary interfaces. This step establishes the governance perimeter and informs membership management decisions.
2) **Establish the governance framework.** Before any technical implementation, define the governance model covering membership management, change management, dispute resolution, and network regulation. BRA-PS recommends a centralized, off-chain, and coercive governance model for public sector contexts where regulatory accountability is mandatory.
3) **Map public and private workflows.** For each inter-organizational process, explicitly identify which steps are public (shared across organizations and executed on the blockchain) and which steps are private (internal to each organization). This mapping directly informs the design of the Communication Layer and the Smart Contract Layer.
4) **Design the smart contracts.** Based on the public workflow map, define the smart contract logic, state transitions, and endorsement policies. Externalize decision logic to a rule engine where frequent updates are anticipated, to avoid smart contract redeployment costs.
5) **Configure the data strategy.** Determine which data resides on-chain (document states, transaction records) and which resides off-chain (large files, sensitive documents). If using IPFS, deploy a private network within the organizational perimeter and ensure all files are encrypted prior to upload.
6) **Implement the Communication Layer.** Develop the Blockchain Adapter and Orchestration Module to synchronize public blockchain events with each organi-zation's private management system. Standardize data exchange formats between the blockchain and legacy systems at this stage.

 doi:10.20944/preprints202605.0472.v1

7) **Develop the Presentation Layer.** Build citizen-facing interfaces that expose public blockchain states in ac-cessible, non-technical language. Ensure citizens can monitor the progression of their requests and understand how their data is being handled, addressing transparency requirements directly.
8) **Conduct an ATAM evaluation.** Before production deployment, convene a multi-disciplinary evaluation panel including at least one non-technical participant as a citizen proxy. Use the utility tree and scenario-based analysis to identify risks, sensitivity points, and trade-offs specific to the deployment context.
9) **Deploy and monitor.** Deploy the HLF network across multiple hosts, with each organization managing its own infrastructure. Implement real-time monitoring, anomaly detection, and access control lists to maintain network health and security post-deployment.

Conclusions

This study presents a reference architecture for blockchain integration in public sector services, designed to enhance trans-parency and interoperability while maintaining organizational autonomy and data privacy. Through a layered architecture and cross-cutting concerns, we provide a reference architecture that addresses key requirements essential for citizen-centric public sector applications. Our case study on vehicle registration between two public agencies demonstrates the practical viability of the architecture, whereas an ATAM-based evaluation input highlights its strengths and areas for refinement.

The results underscore the architecture's ability to guide effective implementation decisions and foster stakeholder alignment. Moving forward, this reference architecture can be adapted and extended to other public sector scenarios, assisting institutions in navigating blockchain adoption, foster-ing inter-organizational collaboration, and supporting citizen-centric solutions. Expanding future evaluations to include both multi-channel communication scenarios and multi-chain management would offer a more comprehensive understanding of the adaptability and effectiveness of the architecture across different public sector contexts.

Acknowledgments: This article is a revised and expanded version of a paper entitled "Towards Citizen-Centric Services using Blockchain-Powered Digitalization of Public Sector Processes," which was presented at the 2023 Fifth International Conference on Blockchain Computing and Applications (BCCA), Kuwait, Kuwait, October 2023. The authors used Paperpal to improve syntax and grammar.

References

1. E. Commission, J. R. Centre, L. Vaccari, F. Pignatelli, D. Allessie, and M. Sobolewski, Blockchain for digital government—An assessment of pioneering implementations in public services. Publications Office, 2019.
2. L. Bass, P. Clements, and R. Kazman, Software Architecture in Practice, 4th ed. Addison-Wesley Professional, 2021.
3. H. Cervantes and R. Kazman, Designing Software Architectures: A Practical Approach. Addison-Wesley Professional, 2016.
4. P. Kruchten, "The 4+1 view model of architecture," IEEE Software, vol. 12, no. 6, pp. 42–50, 1995.
5. R. Kazman, M. Klein, and P. Clements, "ATAM: Method for architecture evaluation," no. CMU/SEI-2000-TR-004, 2000.
6. S. I. Sion, K. Zhang, and A. April, "Towards citizen-centric services using blockchain-powered digitalization of public sector processes," in 2023 Fifth International Conference on Blockchain Computing and Applications (BCCA), 2023, pp. 361–368.
7. A. Josey, TOGAF® version 9.1-A pocket guide. Van Haren, 2016.
8. E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. D. Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti,

 doi:10.20944/preprints202605.0472.v1

C. Stathakopoulou, M. Vukolic', S. W. Cocco, and J. Yellick, "Hyperledger fabric: a distributed operating system for permissioned blockchains," p. Article 30, 2018. [Online]. Available: https://doi.org/10.1145/3190508.3190538

9. M. Gupta, Blockchain for the Enterprise: The definitive guide for enterprise blockchain adoption, 2018.
10. E. S. Negara, A. N. Hidyanto, R. Andryani, and D. Erlansyah, "A survey blockchain and smart contract technology in government agencies," IOP Conference Series: Materials Science and Engineering, vol. 1071, no. 1, p. 012026, 2021. [Online]. Available: https://dx.doi.org/10.1088/1757-899X/1071/1/012026
11. D. Tapscott and A. Tapscott, "How blockchain will change organizations," MIT Sloan Management Review, vol. 58, no. 2, pp. 10–13, 2017, copyright—Copyright Aˆ © Massachusetts Institute of Technology, 2015. All rights reserved. CODEN—SMRVAO. [Online].Available:https://www.proquest.com/scholarly-journals/ how-blockchain-will-change-organizations/docview/1875399260/se-2?accountid=27231https://etsmtl.on.worldcat.org/atoztitles/link?sid=ProQ:&issn=15329194&volume=58&issue=2&title=MIT+Sloan+ Management+Review&spage=10&date=2017-01-01&atitle=How+ Blockchain+Will+Change+Organizations&au=Tapscott%2C+Don%3BTapscott%2C+Alex&id=doi:
12. J. Mendling, I. Weber, W. V. D. Aalst, J. V. Brocke, C. Cabanillas, F. Daniel, S. Debois, C. D. Ciccio, M. Dumas, S. Dustdar, A. Gal, L. Garc'ıa-Ban˜uelos, G. Governatori, R. Hull, M. L. Rosa, H. Leopold, F. Leymann, J. Recker, M. Reichert, H. A. Reijers, S. Rinderle-Ma, A. Solti, M. Rosemann, S. Schulte, M. P. Singh, T. Slaats, M. Staples, B. Weber, M. Weidlich, M. Weske, X. Xu, and L. Zhu, "Blockchains for business process management—challenges and opportunities," ACM Trans. Manage. Inf. Syst., vol. 9, no. 1, p. Article 4, 2018. [Online]. Available: https://doi.org/10.1145/3183367
13. S. Seebacher and M. Maleshkova, "A model-driven approach for the description of blockchain business networks," in Hawaii International Conference on System Sciences, 2018, Conference Proceedings.
14. W. Eggers, "Taking a citizen-centric approach to state government," 2022-11-26 2018. [Online]. Available: https://deloitte.wsj.com/articles/ taking-a-citizen-centric-approach-to-state-government-1521000134
15. U. Bodkhe, S. Tanwar, K. Parekh, P. Khanpara, S. Tyagi, N. Kumar, and M. Alazab, "Blockchain for industry 4.0: A comprehensive review," IEEE Access, vol. 8, pp. 79 764–79 800, 2020.
16. R. Viswanathan, D. Dasgupta, and S. R. Govindaswamy, "Blockchain solution reference architecture (bsra)," IBM Journal of Research and Development, vol. 63, no. 2/3, pp. 1:1–1:12, 2019.
17. Y. Liu, Q. Lu, G. Yu, H. Y. Paik, and L. Zhu, "A pattern-oriented reference architecture for governance-driven blockchain systems," in 2023 IEEE 20th International Conference on Software Architecture (ICSA), Conference Proceedings, pp. 23–34.
18. Y. Wang, S. Li, H. Liu, H. Zhang, and B. Pan, A Reference Architecture for Blockchain-based Traceability Systems Using Domain-Driven Design and Microservices, 2023.
19. Y. Gong, S. van Engelenburg, and M. Janssen, "A reference architecture for blockchain-based crowdsourcing platforms," Journal of Theoretical and Applied Electronic Commerce Research, vol. 16, no. 4, pp. 937–958, 2021. [Online]. Available: https://www.mdpi.com/0718-1876/16/4/53
20. T. Weber and R. Buchkremer, "Blockchain-based reference architecture for automated, transparent, and notarized attestation of compliance adaptations," Applied Sciences, vol. 12, no. 9, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/9/4531
21. I. Aviv, A. Barger, A. Kofman, and R. Weisfeld, "Reference architecture for blockchain-native distributed information system," IEEE Access, vol. 11, pp. 4838–4851, 2023.
22. A. N. Gohar, S. A. Abdelmawgoud, and M. S. Farhan, "A patient-centric healthcare framework reference architecture for better semantic interoperability based on blockchain, cloud, and iot," IEEE Access, vol. 10, pp. 92 137–92 157, 2022.
23. P. H. D. Valle, L. Garce's, T. Volpato, S. Mart'ınez-Ferna'ndez, and E. Y. Nakagawa, "Towards suitable description of reference architectures," PeerJ Computer Science, vol. 7, p. e392, 2021.
24. V. Peristeras, K. Tarabanis, and S. K. Goudos, "Model-driven egovernment interoperability: A review of the state of the art," Computer Standards Interfaces, vol. 31, no. 4, pp. 613–628, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/ S0920548908001372

 doi:10.20944/preprints202605.0472.v1

25. M. Reece and S. Mittal, "Self-sovereign identity in a world of authentication: Architecture and domain usecases," 2022. [Online]. Available: https://arxiv.org/abs/2209.11647