



Deposited via The University of Leeds.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/240505/>

Version: Accepted Version

Article:

Mamede, W.S., Marotta, M.A., Both, C.B. et al. (2026) A Comparative Analysis of Future Internet Architectures Interoperability With the Current Web Browser Application.

International Journal of Communication Systems, 39 (4). e70419. ISSN: 1074-5351

<https://doi.org/10.1002/dac.70419>

This is an author produced version of an article published in the International Journal of Communication Systems, made available via the University of Leeds Research Outputs Policy under the terms of the Creative Commons Attribution License (CC-BY), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

RESEARCH ARTICLE

A Comparative Analysis of Future Internet Architectures Interoperability with the Current Web Browser Application

William Silva Mamede¹ | Marcelo Antônio Marotta² | Cristiano B. Both³ | Antônio Marcos Alberti⁴ | José Marcos C. Brito¹

¹ICT Laboratory, INATEL - Instituto Nacional de Telecomunicações, Minas Gerais, Brazil

²Instituto de Exatas - Departamento de Ciência da Computação (CIC), University of Brasília - UnB., Distrito Federal, Brazil

³Graduate Program in Applied computing - PPGCA, University of Vale do Rio dos Sinos - UNISINOS., RS, Brazil

⁴School of Computer Scienc, University of Leeds, Leeds, UK

Correspondence

Corresponding author William Silva Mamede, Instituto Nacional de Telecomunicações, Minas Gerais, Brazil
Email: william.silva@mtel.inatel.br

Abstract

Redesigning the Internet to overcome the limitations of the current one has been a topic of discussion for some years. These discussions aim to provide solutions for optimizing resource consumption and providing more flexibility, which are key features in the modern world of interconnected devices and applications. Today, the applications created adhere to the norms of the current Internet, which is based on well-established protocols and tools. However, it is also restricted by its limitations, which are expected to be addressed in content-oriented Future Internet Architecture (FIA) solutions. These approaches aim to provide a network environment in which the data itself is the key in the communication, opposite to the current solution, which relies on hosts' address and location. The transition between different architectures in FIAs is anticipated to be challenging, as existing solutions must be modified to meet FIA protocols and patterns. Therefore, to ensure a seamless evolution of the network, regardless of the environment in which it is used, it is essential to provide a means of enabling this transition. For instance, solutions like NDN.js and COIN, which were developed for NDN, are intended to facilitate communication across different architectures. This survey aims to provide a comprehensive overview of the potential benefits of interoperability between the current Internet/Web and FIAs, as well as how existing applications can benefit from it.

KEYWORDS

Internet, Overlay Networks, Next Generation Networks, Internetworking

1 | INTRODUCTION

The exponential increase in devices connected to the Internet in recent years is projected to reach more than 75 billion devices that transfer data worldwide by 2025, signaling a significant increase in data exchange according to Banafa (2023). However, this surge poses challenges to the traditional client-server model of data dissemination, potentially straining the conventional content distribution methods of the World Wide Web (WWW). As highlighted in Conti et al. (2020), the existing Internet architecture, which is based on dual semantics for host identification and location, may struggle to cope with this increasing demand and may require new paradigms and solutions to offer a more efficient, flexible, and scalable network environment.

Information-Centric Networking (ICN) emerges as a promising alternative to the prevailing host-centered Internet model, prioritizing data over host addresses in information retrieval Ngaffo et al. (2020). This alternative promises a more efficient use of network resources, allowing quicker responses to user

requests by centering communication around content. Various architectural frameworks, including NovaGenesis (NG) Alberti and Singh (2014), Named Data Networking (NDN) Zhang et al. (2014a), Recursive InterNetworking Architecture (RINA) Vrijders et al. (2014), eXpressive Internet Architecture (XIA) Naylor et al. (2014) and Mobility First (MF) Raychaudhuri et al. (2012), have been proposed to embrace this new paradigm. Despite the inherent benefits of ICN, integrating conventional web applications, which rely on host-centric protocols such as the Internet Protocol (IP), into an ICN environment remains a matter of investigation Ahlgren et al. (2012).

The existing WEB does not support interoperability with emerging Internet architectures. Current WEB applications, designed for TCP/IP networks, face inherent limitations in scalability, security, and mobility Ngaffo et al. (2020) Ahlgren et al. (2012) Islam et al. (2025). In contrast, content-oriented WEB pages, underpinned by the ICN principles, offer potential solutions to mitigate these limitations, according to the information provided by Conti et al. (2020). Despite the necessity of a complete reimplementaion for most current WEB applications to operate on alternative Internet architectures, rewriting them

proves financially and temporally prohibitive. Therefore, new studies must be carried out to offer alternatives to WEB applications to transition from a host-centric to a content-oriented WEB using new Internet architectures.

Our research endeavors to assess and compare current solutions for Future Internet Architectures (FIAs) and their applicability to existing WEB applications. Our study examines existing solutions that enable the concurrent use of existing applications across various Internet environments, aiming to create a digital ecosystem that combines the benefits of FIAs with established Web and TCP/IP technologies. The transition towards ICN architectures, while promising, necessitates a strategy for coexistence between ICN and IP to ensure backward compatibility with existing IP-based applications and services. Thus, our research evaluates proposals and solutions that foster interoperability between different Internet architectures at the WEB level, examining methods such as middle-layer proxies, client-side browser extensions, and FIA-compliant WEB browsers. These assessments aim to inform the development of solutions for seamless interconnection between WEB applications and NovaGenesis (NG) FIA, offering insight into potential benefits and challenges posed by this integration.

This article is organized as follows. Section 2 discusses the related work and the approaches followed by the FIAs. Section 3 provides the research questions and methodology behind this article. Section 4 provides a detailed discussion of the answers we offer to the questions. In the last Section 5 we will present the conclusion of this study.

2 | RELATED WORK

Many studies aim to introduce a novel Internet architecture, providing an alternative to replace the existing TCP/IP network, which was developed and designed decades ago, to create a new solution to meet actual needs, e.g., scalability, security, privacy, quality of service, interoperability, and flexibility Conti et al.. Various solutions have been proposed to reach the desired goal, as illustrated in Figure 1, such as NovaGenesis Alberti and Singh (2014), XIA Naylor et al. (2014), RINA Day (2010), NDN Zhang et al. (2014b), and MobilityFirst Raychaudhuri et al. (2012). Notwithstanding, these approaches are well-suited to address the limitations related to the network infrastructure and its design. However, there are other issues to be addressed, including the interoperability of existing and already established Web systems, which have been designed for the current Internet and often need to run over FIA. Therefore, our research aims to evaluate existing FIAs by examining whether they can run legacy applications on new architectures.

To enable interoperability between legacy web applications and emerging Internet Architectures (post-IP), our research identifies several potential methods, presented as **interoperability solutions in Figure 1** and detailed in Sections 3 and 4. A key approach involves the use of intermediate proxies, designed to bridge the gap between existing applications and new network stacks. As depicted in Figure 2, these proxies are strategically positioned at both the access side (between the web client and the FIA) and the far-end side (between the FIA and the web server). For instance, imagine a legacy application that utilizes HTTP requests over a TCP/IP socket to communicate with a server. This “tunneling” approach preserves the application’s request and server’s response

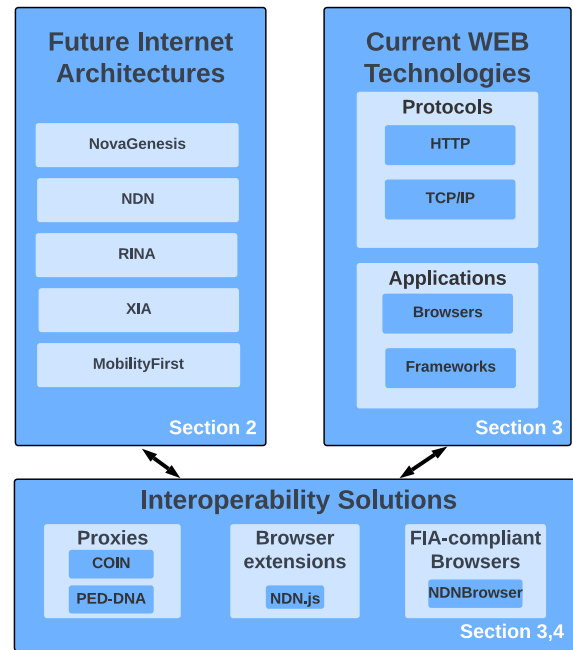


FIGURE 1 Method to connect FIAs with the existing Web.

boundaries. However, it requires enhancing the Future Internet Architecture (FIA) to understand and process these requests and retrieve the requested data from the designated server. This can be achieved by implementing a translation layer that maps HTTP requests to FIA primitives, such as subscriptions, and vice versa. Consequently, this architecture allows for the evolution of the network core while maintaining backward compatibility with the existing web ecosystem. This concept has been previously explored, notably in Siddiqui and Mueller (2012), where the authors proposed a socket-based API translation mechanism to facilitate the transition to future Internet paradigms.

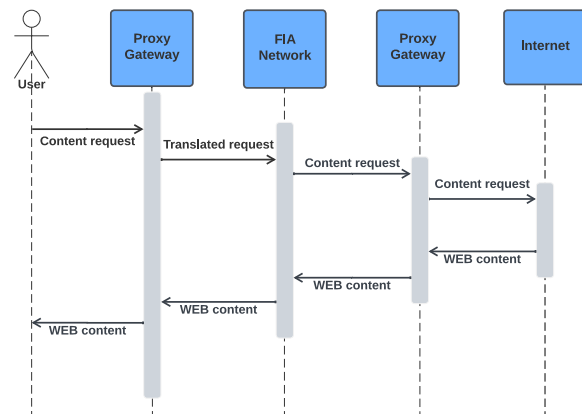


FIGURE 2 Tunneling approach for Web interoperability used by FIA.

The authors from Content-oriented interoperability framework for current and future Internet architectures (COIN) Jahanian et al. (2020b), for example, provided an interoperability tool, similar to the “tunneling” presented previously, in which a proxy was implemented responsible for translating HTTP requests for NDN Zhang et al. (2014b) at the network entry point and vice versa. Therefore, COIN does not imply any change at the application and server levels. A similar solution can also be found in Performance Enhancing Proxy for Deploying Network Architectures (PEP-DNA) Ciko et al. (2021), where a proxy is used to translate messages from TCP to RINA Day (2010) and vice versa. The main difference in this approach is that the proposed proxy has been implemented as part of the Linux operating system kernel noa (2022). This article will discuss how both solutions operate, providing an in-depth examination of how they achieve backward compatibility. Table 1 below aims to anticipate and provide a comparison between the most prominent FIAs, by evaluating several aspects including the communication model, which describes how the data are exchanged between the client and the server; network cache, which indicates whether the a caching mechanism is used to store and retrieve data in the network; object naming, which specifies how it identifies and locates the objects and applications in the network; browser approach, which reports whether the architecture has a dedicated WEB browser or not; and interoperability, which evaluates whether it can interoperate with the existing IP-based hosts and applications.

NDN networks are known to utilize CCNx (Content-Centric Networking) as their underlying protocol Zhang et al. (2014a), which was designed to enable content-centric communication by focusing on the data rather than the host identification. However, a solution has been developed on this architecture to provide WEB support, called NDN.JS Shang et al. (2013), which works as a client-side library, to provide a tool to implement JavaScript applications and allow iterations with the NDN network without the CCNx protocols; in other works, this feature enables the development of applications that can communicate directly with the NDN network. This approach makes it easier to construct and operate WEB pages, as it does not necessitate the installation of any CCNx-related software. Consequently, this solution enables communication by establishing a WebSocket proxy that translates the communication between the application and the network.

Other strategies seek to create a new WEB from the ground up. These solutions could be built using QtWebKit qtW (2022), which provides a framework to embed pages into any program. Following this approach, a solution over RINA Wang and Qiao (2014), which aims to develop a native-based WEB browser, was designed. This solution provides native application development and execution on FIAs; however, there are still challenges to overcome, such as ensuring interoperability with existing WEB technologies, since QtWebKit is not a widely used technology for WEB development; additionally, the development of a new navigation browser requires significant resources and time, which may not be feasible. Despite the difficulties, the creation of this type of implementation could offer a more effective and scalable solution, as applications would be created specifically to run on a particular architecture.

Furthermore, for architectures such as XIA and NovaGenesis, there was no evidence of any implementation that allows Web consumption, even when

considering in-browser design strategies, or developing additional layers to facilitate cross-architecture data translation, as outlined at Table 1. This absence of interoperability solutions not only presents a major barrier to the adoption and evaluation of these promising architectures, but also highlights a critical area for future research and development.

The studies conducted so far have focused on finding ways to create Web applications that can be executed, developed, and used in an FIA environment. However, no implementation or study seeks to make it possible to interoperate with existing web applications, thereby making them backward compatible with the existing applications. In light of this, we recognize the potential and opportunity in this study area that could lead to the future development of solutions allowing current systems to be executed over FIAs, without requiring significant changes to the already established and designed software. To achieve this, our study aims to evaluate existing solutions and provide a knowledge base that can be used to assess the feasibility of implementing features to enable the consumption of existing Web applications in different architectures.

3 | METHODOLOGY

This article was developed using the systematic review methodology (SLR), as described by Kitchenham et al. (2009). Therefore, our SLR is a study that aims to analyze scientific literature-related application interoperability between the current Internet and Future Internet Architectures. To achieve the goal presented, it was decided to use the PICOC framework Liberati et al. (2009) (Population, Intervention, Comparison, Outcome, and Context) to guide our systematic review of the literature, as it provides an efficient way to define the scope of our study and compare its results.

3.1 | Research questions

We defined a central question (MQ) that acts as a guide in the process of our study. In addition, the MQ was broken down into several subquestions (SQ), listed in Table 2, which follow the PIPOC principle, to specify the challenges, possible solutions, and opportunities addressed in detail by the selected articles.

The purpose of the MQ is to present, at a high level, the subject matter under investigation, which is the interoperability of WEB applications between the current Internet and Future Internet architectures. The first sub-question, SQ1, aims to evaluate the type of solution used in ongoing related research and later define a direction forward in our use case, using NovaGenesis Alberti and Singh (2014). The second, SQ2, aims to identify FIAs, enabling the execution of legacy Web applications in its architecture and how it was achieved. The third question, SQ3, aims to evaluate whether any changes are needed in the existing application to make it compatible with other architectures. The fourth sub-question, SQ4, seeks to address the established approach to creating a Web from scratch, developed in response to a specific FIA need. The fifth sub-question, SQ5, aims to assess the primary impacts that this approach would yield. Then, in the sixth question, SQ6, after answering the questions presented above, it will aim to identify possible challenges that can

TABLE 1 Future Internet Web approaches.

Aspects	NDN	RINA	XIA	MF	NovaGenesis
Internet Protocols					
Communication Model	The content is pushed and, when needed, pulled by users. To receive content, the user must know the name of the needed content. Therefore, NDN uses a push/pull model.	For establishing communication between a client and a server, the client must specify the desired application process name, which will then be found by following a network recursive flow forwarding approach. Employs a client/server approach similar to the current Internet.	Employs a client/server approach similar to the current Internet. The model depends on the principal type selected by the application.	MF provides a pull-based model, in which the desired content must be specified in a request.	Provides a publish/subscribe WEB. Web pages are converted to an NG pattern and then published in a domain-level cache. If a user wants the same Web page, it must subscribe for it in the network cache.
Does it use a network cache?	NDN has an in-network cache.	No cache is implemented for RINA.	XIA has a network content cache.	MF has an in-network cache.	NG provides a domain-level cache approach in the network.
Are the objects/applications named?	Objects and applications are named using a hierarchical naming scheme composed of name components separated by forward slashes (/) Team (2014)	Content named using location-independent names. The information about an application process name is handled by an application called DIF allocator. DIFs stands for Distributed Inter-Process Communication Facilities, which are layers on RINA network, that cooperates to provide communication services. DIF allocator is responsible for managing the lifecycle of these network layers.	Yes, objects and applications are named using self-certifying identifiers called eXpressive Internet Architecture identifiers (XIDs) Naylor et al. (2014). These identifiers are unique, global identifiers associated with a public-private key pair. The XID is the hash of the public key.	Yes. Names are unique global identifiers Bhanage et al. (2011), which are used to identify elements present in the network	Yes. Objects, services, and content are named, and it is possible to establish a relationship between them using the name bindings.
Web protocols					
New browser approach	A NDN WEB browser Qiao et al. (2015), with the CCNx and QT framework.	We did not find any implementation	We did not find any implementation	We did not find any implementation	We did not find any implementation
Interoperability	There is a proxy, called COIN Jahanian et al. (2020b), that intermediates the communication between NDN and IP-based hosts. In another work, there is an NDN.JS Shang et al. (2013) that consists of a browser add-on responsible for translating communication within the NDN network.	It has a solution, called PEP-DNA, that contains a proxy responsible for translating TCP requests to RINA.	We have not found any implementation of a proxy.	The COIN proxy can also be used for MF, acting as an intermediary between MF and IP-based hosts to facilitate communication.	No such implementation exists yet, but our research aims to assess the feasibility of creating a solution for NG, which would enhance current web interoperability by utilizing an alternative internet protocol stack.
Real-World Adoption	Not observed.	Not observed.	Not observed.	Not observed.	Not observed.

be faced during study and development. A set of open-ended questions will be provided, serving as a guide for future studies.

TABLE 2 Research questions.

Identifier	Issue
MQ	How to interoperate the current Web with FIAs
SQ1	Which problems the Web could solve?
SQ2	Which solutions are used to interoperate with the Web?
SQ3	How can Web applications adapt to new architectures?
SQ4	How the future Internet Web could be developed?
SQ5	What are the impacts expected from new Web architectures?
SQ6	How to interoperate with AI?
SQ7	How to interoperate with DLTs?
SQ8	What are the challenges and open questions?

3.2 | Search strategy

To find related studies that help answer the research question, it was decided to use IEEE Xplorer, Google Scholar, Scencedirect, Springer, ACM Digital Library, ResearchGate, and MDPI (at Future Internet Magazine) as the primary database sources. The reason for choosing it was based on the number of studies available on the platform related to the study topic, the relevance of the articles published there, and the search tool provided. We also defined a search string shown in Figure 3, which contains the main topics related to our study. **The search string is fundamentally divided into two distinct groups of keywords linked by a mandatory "AND" operator. The initial section includes terms ensuring results pertain solely to the Future Internet, whereas the latter part covers broader technological concepts within interoperability terms. Therefore, the "AND" operator acts as a filtering that will narrow the results significantly, ensuring that only articles that simultaneously discuss concepts from both sets of keywords are retrieved, thus focusing the search on the specific intersection of these domains.**

("future internet" OR "FIA" OR "new internet" OR "new generation network"
 OR "ICN" OR "Information centric networking" OR "future internet web")
 AND ("web" OR "interoperability" OR "application interoperability" OR
 "application socket" OR "legacy application" OR "socket" OR
 "Distributed Ledger Technology" OR "DLT" OR "blockchain" OR "Artificial
 Intelligence" OR "AI")

FIGURE 3 Search string.

3.3 | Article selection

The search string returned an amount of documents that could not exactly reflect the needs of our SLR; therefore, a list of acceptance criteria was developed, presented in Table 3, to filter only articles that match the subject studied.

TABLE 3 Quality assessment criteria.

Section	Description
Criteria 1	Filter looking for a period of 11 years, 2012 to 2023.
Criteria 2	The papers are reported in a peer-reviewed magazine.
Criteria 3	The papers are written in English.
Criteria 4	Remove books and technical reports.

We searched for articles that met all the requirements and selected only those that were compatible with the proposed requirements defined at the beginning of this study. The query was executed over the chosen database, yielding a substantial number of publications across various domains and disciplines. However, not all of them were relevant to our research topic and objectives. Therefore, we applied a set of criteria to filter out irrelevant or outdated articles, retaining only the most relevant and recent ones. The requirements included publication year, journal or conference quality, number of references, research methodology, and alignment with our research questions, according to what was explained in Table 3. Figure 4 summarizes the strategy used to obtain articles from each database before and after applying the criteria. As presented in Table 4, the final number of selected articles indicates the novelty and originality of our research proposal.

4 | RESULTS AND DISCUSSION

Previously in this document, the research questions that we aim to answer were briefly presented. In this Section, we will provide a detailed discussion of these questions by covering a comprehensive discussion about our findings. Therefore, we aim to provide a comprehensive understanding of the existing challenges and potential solutions to address them. Furthermore, by comparing NovaGenesis to existing works, we can see how it compares to the state-of-the-art and how its new features benefit the entire community.

4.1 | SQ 1 - Which problems the Web could solve?

To answer this question, we performed an analysis of each FIA implementation described in Table 4. We base our work on the examination of several essential studies that emphasize interoperability and backward compatibility as significant challenges. The studies Jahanian et al. (2020a), Jahanian et al. (2020b), Zali et al. (2017) and Siddiqui and Mueller (2012) are central to our findings. These works reveal that a critical barrier to the adoption of Future Internet Architectures (FIAs) is the absence of a backward-compatible infrastructure for the large number of current Web applications.

There is a well-established pattern used to develop Web applications, which could consume resources from Application Programming Interfaces (APIs) Ofoeda et al. (2019), Web sockets, among other features provided by the current Internet. Additionally, most existing programming languages and frameworks used in web development are entirely dependent on establishing connections through a host-based model that follows current Internet protocols, such as HTTP, TCP, and IP.

The concept of Software as a Service (SaaS) emerged in the last decade, resulting in the development of numerous web-based solutions that facilitate connections between devices and people. However, if we only consider online pages, there are estimated to be 1.13 billion active sites Huss (2023). Therefore, attempting to create brand new applications to replace existing ones that work natively in FIAs would be an immense waste of time and money. Following a proxy-based approach, migration to other architectures can be done more easily, allowing for the creation of more robust and scalable solutions, without losing already developed Web solutions.

In addition to this, there is an emerging concept known as the Semantic Web Alabdulwahhab (2018), a new proposal from the World Wide Web Consortium that aims to make online content more understandable to machines, enabling more intelligent searches, recommendations, and automation. The goal is to enhance current technology by introducing new features, including decentralized information, security enhancements, and the incorporation of artificial intelligence, to create an interconnected and intelligent Web. The primary objective of this approach is to facilitate consumers' access to data; therefore, we can conclude that content-oriented FIAs will significantly contribute to achieving this goal.

We can conclude, based on the explanation provided, that a solid infrastructure exists that enables the implementation of Web interoperability environments. The COIN study Jahanian et al. (2020b), which was briefly mentioned

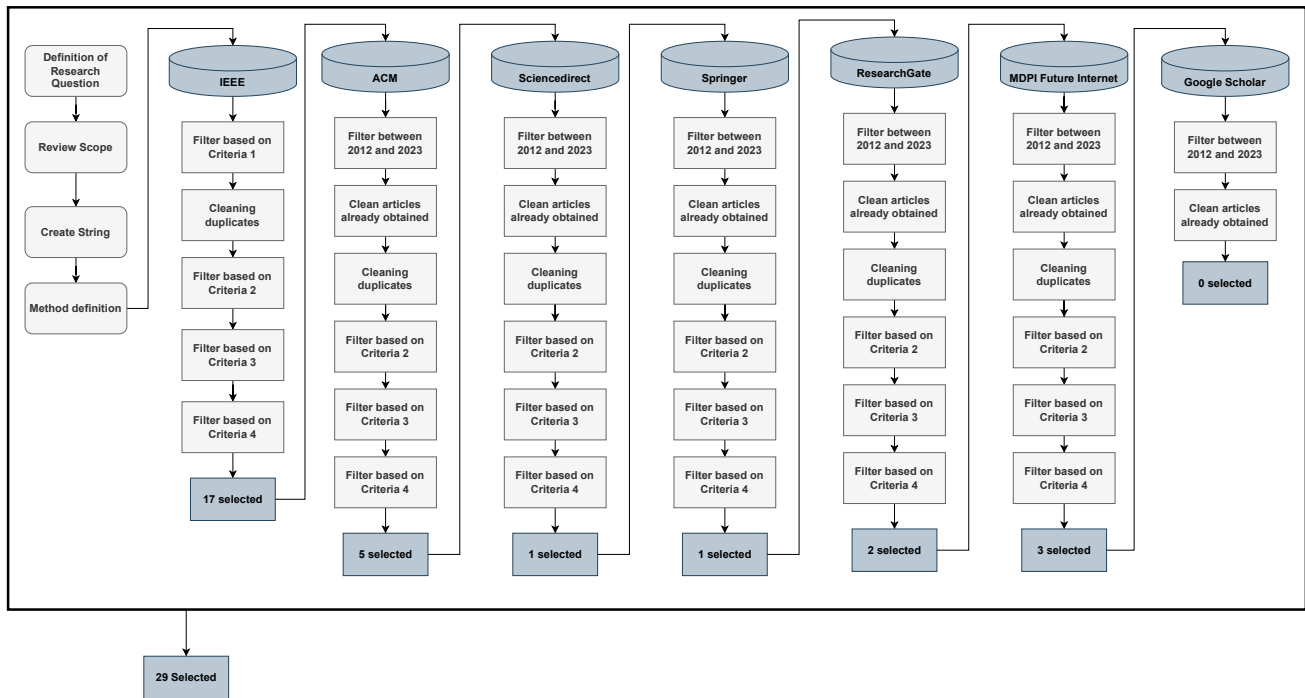


FIGURE 4 Screening of research.

in Table 4, suggests that an interoperable solution is the way forward to allow existing network applications to coexist with FIA. This solution, known as the Content-Oriented Interoperability Framework (COIN), is designed to provide seamless connectivity between multiple network frameworks and the existing Internet.

This approach is believed to allow the development of new networks while preserving key features and mechanisms from other implementations. In addition, it avoids the need to expand domain-specific protocols or namespaces. Instead, an additional layer application is used, called the Object Resolution Service (ORS). This service is responsible for resolving names and providing them to consumers; therefore, it maps names used in one domain to those in another. Furthermore, it provides translation gateways responsible for enabling communication between different architectures, storing information regarding a given protocol and session information, and then performing proper translation of protocols between the source and destination. Therefore, this system intercepts all communication and changes it so that the message can be read and understood by both the sender and the receiver.

Furthermore, the paper Jahanian et al. (2020a) introduces the concept of information-centric interoperability (ICI), which enables different ICN architectures to communicate with each other. To model and analyze their approach, they used the Alloy declarative modeling language Jackson (2002), which allows the concise and expressive specification of complex systems. By using Alloy, they were able to develop an ICI framework that bridges the current Internet, NDN, and MF, allowing for the exchange of content based on its name, regardless of its location, and providing mobility support a key feature of content-oriented ICN. Following a similar concept, other studies aim to provide a backward-compatible solution, such as Zali et al. (2017) and Siddiqui

and Mueller (2012), which have the goal of introducing an additional layer responsible for enabling intercommunication between different networks.

Other existing solutions, such as NDN.JS Shang et al. (2013), described in Table 4, provide a new Web structure that allows Web navigators to communicate with FIA. They provide a set of tools that facilitate communication, allowing the development of Web applications that can be connected to other network patterns. Despite being useful for creating new applications, these solutions are not suitable for existing applications to coexist in the same environment. They are, however, effective for creating applications that can be executed using the most recent Web technologies. This type of solution will enable the creation of applications that can naturally interact with FIAs, providing solutions that can benefit from those architectures.

As discussed previously and according to Zali et al., backward compatible solutions can promote adherence to industry standards and protocols, ensuring compatibility and seamless communication between Web applications executed over a cross-platform environment and multiple FIAs, which could fully potentialize content distribution, since the interoperability environment will enable simultaneous use of various applications and protocols, resulting in smooth progression of the network. Furthermore, according to Siddiqui and Mueller, this type of solution could expand the user base and services, as it would offer a flexible and adjustable approach to controlling network resources. Therefore, we concluded that interoperability enables the execution and distribution of applications created with different architectures, without relying on the Internet's structure.

TABLE 4 Selected articles.

Article	Publisher	Name
Zali et al. 2017	IEEE	Peer-Assisted Information-Centric Network (PICN): A Backward Compatible Solution
Siddiqui and Mueller 2012	IEEE	A Requirement-Based Socket API for a Transition to Future Internet Architectures
Jeon et al. 2015	IEEE	An extended WEB browser for id/locator separation network
Chen et al. 2016	IEEE	Exploiting ICN for Efficient Content Dissemination in CDNs
Saverimoutou et al. 2019	IEEE	Influence of Internet Protocols and CDN on Web Browsing
Wang and Qiao 2014	IEEE	Design and implementation of WEB browser for named data networking in Windows
Rezk and Fofou 2014	IEEE	A Survey of Semantic WEB Concepts Applied in Web Services and Big Data
Shang et al. 2013	IEEE	NDN.JS: A javascript client library for named data networking
Yoo and Ko 2021	IEEE	Implementation and Evaluation of NDN-based Video Streaming System.
Ciko and Welzl 2019	IEEE	First Contact: Can switching to RINA save the Internet?
Grewe et al. 2021	IEEE	On the Design of an Information-Centric Networking Extension for IoT APIs.
Doyen et al. 2019	IEEE	An Orchestrated NDN Virtual Infrastructure Transporting Web Traffic: Design, Implementation, and First Experiments with Real End Users.
Jahanian et al. 2020b	IEEE	Managing the Evolution to Future Internet Architectures and Seamless Interoperation.
Ciko et al. 2021	IEEE	PEP-DNA: A Performance Enhancing Proxy for Deploying Network Architectures.
Conti et al. 2020	IEEE	The Road Ahead for Networking: A Survey on ICN-IP Coexistence Solutions.
Zhu et al. 2021	IEEE	iGate: NDN Gateway for Tunneling over IP World.
Panwar and Bhatnagar 2020	IEEE	Distributed Ledger Technology (DLT): The Beginning of a Technological Revolution for Blockchain.
Qiao et al. 2015	Sciencedirect	NDNBrowser: An extended WEB browser for named data networking
Gündoan et al. 2020	ACM	Toward a restful information-centric WEB of things: A deeper look at data orientation in CoAP.
Kutscher and Oran 2022	ACM	RESTful information-centric networking: statement.
Burke 2022	ACM	Statement: As TCP/IP is to the WEB, ICN is to the?
Gündoğan et al. 2018	ACM	NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT
McCauley et al. 2019	ACM	Enabling a Permanent Revolution in Internet Architecture
Jahanian et al. 2020a	Springer	Formal verification of interoperability between future network architectures using alloy
Fotiou et al. 2017	ResearchGate	CoAP over ICN.
Doumanis et al. 2018	ResearchGate	Improving video QoE with IP over ICN.
Yang et al. 2021	MDPI	An Accelerating Approach for Blockchain Information Transmission Based on NDN
He et al. 2023	MDPI	Lightweight Transmission Behavior Audit Scheme for NDN Industrial Internet Identity Resolution and Transmission Based on Blockchain.
Benmoussa et al. 2023	MDPI	NDN-BDA: A Blockchain-Based Decentralized Data Authentication Mechanism for Vehicular Named Data Networking.

4.2 | SQ 2 - Which solutions are used to interoperate with the Web?

This question aims to present the approaches developed to interoperate Web applications in a heterogeneous environment, composed of multiple architectures. To analyze how such a solution could be implemented, we evaluated several research studies on solutions designed to implement a backward-compatible strategy. Previously in this document, in the Related Work Section 2, we could briefly discuss architectures that adopt this kind of enhancement; additionally, Table 1, previously described, also presents how this was handled for different FIAs, including NDN, RINA, NG, and MobilityFirst.

Our analysis reveals two approaches applied for enabling interoperability:

- **Intermediate layer addition:** Approaches such as Jahanian et al. (2020b), Ciko et al. (2021), Gündoğan et al. (2018), Fotiou et al. (2017), and Gündoan et al. (2020) have been developed to facilitate communication between diverse architectures by incorporating extra layers that handle protocol translation.
- **Built-in Application solutions:** As example, we can mention Shang et al. (2013), Wang and Qiao (2014), Jeon et al. (2015) which aim to enable

Web consumption by integrating FIA communication capabilities directly into the Web client.

Introducing a proxy is the most straightforward approach to meet this demand, as demonstrated in Figure 2. However, this approach has a downside in that it will likely lead to an increase in message processing due to the addition of a new layer. On the other hand, it also has the potential to make the application compatible with new architectures. Several studies have been conducted to explore this type of solution, as discussed above in SQ 1, specifically Jahanian et al. (2020b) and Jahanian et al. (2020a).

COIN is a solution that offers numerous benefits due to its interoperable behavior, which enables seamless connectivity, allowing communication across different architectures. Figure 5 provides an overview of COIN behavior, in which we can observe that an additional layer is present in the communication, performing the necessary translations between the different architectures.

Another solution, called the Performance Enhancement Proxy for the Deployment of Network Architectures (PEP-DNA) Ciko et al. (2021), has been developed for RINA. Similarly, this approach also aims to introduce an additional layer that provides a proxy responsible for translating communication between RINA and the current Internet. To achieve this, PEN-DNA intercepts

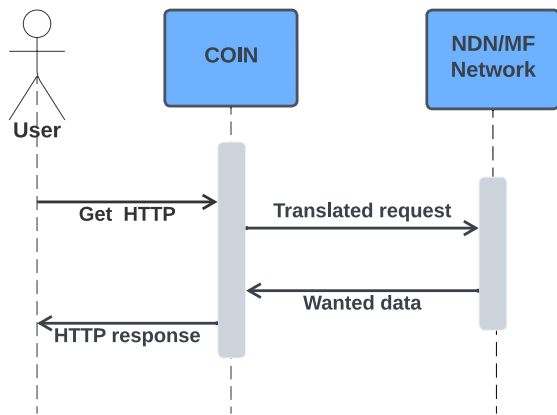


FIGURE 5 IP and NDN/MF communication via COIN.

IP packets using Netfilter[†], a framework installed in the Linux kernel that provides a set of tools, including packet filtering, network address translation (NAT), and port translation. Additionally, to establish and maintain TCP connections, the proxy has a connector component, also known as a connection handler, and works in conjunction with another submodule called the relay engine, which aims to relay between the client and the server.

These modules relay between client and server, translating the traffic between two networks after package interception. Figure 6 presents the connection between these components and expresses how a connection is established based on the examination. If a client application wishes to send data through this system, it will need to take the following steps:

1. An application running on a client computer using the TCP/IP protocol sends a packet to a server located in a different network architecture, such as RINA. This packet is intercepted by the IP-layer hook in PEP-DNA, which is implemented as a Netfilter module in the Linux kernel. The IP-layer hook inspects the packet and decides that it needs to be sent to the server in the RINA network.
2. The IP-layer hook sends the packet to the PEP-DNA connection handler. This handler is responsible for setting up and maintaining the TCP connections between the client and the server on both networks. It examines the state table to determine whether an existing connection exists between the client and the server. If not, it will create a new connection.
3. Once the link is established, the connection handler updates the state table with the new connection data, including the IP addresses and port numbers of both the client and the server in both networks.
4. The connection handler sends the packet to the PEP-DNA data relay engine. This engine is responsible for relaying data between the client and the server and improving the performance of TCP connections. The data relay engine inspects the packet and translates the traffic dynamics between the TCP/IP and RINA networks to enhance the connection's performance.

5. The data relay engine transmits the packet that has been converted to the connection handler, which then sends it to the server on the RINA network.
6. The server in the RINA network receives the packet and sends a reply to the client in the TCP/IP network.
7. The IP-layer hook in PEP-DNA captures the response packet and sends it to the connection handler. The connection handler then examines the state table to determine if there is an existing connection between the client and the server.
8. Once the link is established, the connection handler sends the response packet to the data relay engine. This engine is responsible for translating the traffic flow between the RINA and TCP/IP networks to maximize the connection's efficiency.
9. The data relay engine transmits the converted response packet to the connection handler. The connection handler then sends the packet back to the client over the TCP/IP network.

In conclusion, PEP-DNA facilitates communication across multiple Internet applications by intercepting network packets. Thus, this solution enables communication between hosts that have a network backbone comprising multiple Internet architectures. This solution was created on top of an operating system kernel, suggesting that it could be used to browse the Web. However, no research has been conducted to assess this potential.

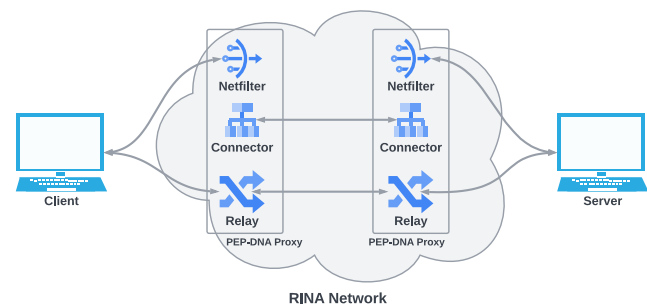


FIGURE 6 PEP-DNA Architecture.

An alternative solution to NDN can be identified, even though it does not offer network interoperability; however, it was intended to facilitate Web navigation. The research paper Shang et al. (2013), which was briefly mentioned in the previous SQ, introduced a JavaScript library named NDNJs. This solution aims to enable the consumption and development of Web pages within NDN network. To do so, it provides a browser extension that forwards all requests in which the URL path contains the prefix "ndn:" to an implemented WebSocket proxy, responsible for intermediating the communication between the navigator and the NDN network. As a result, all page requests performed will be translated into an NDN-based request in this proxy that works as an abstraction layer in the communication with the lower level elements of the NDN network.

[†] <https://netfilter.org/>

To establish a connection with NDN, a solution called PyCCN was used. This is described by the authors as a general-purpose implementation, developed in Python, that enables communication using the content-centric network (CCN) protocol. The utilization of PyCCN allows the host to communicate with the NDN network without the need to install additional tools. Furthermore, NDN.js provides a JavaScript library, which can be obtained from the NDN package manager[†]. This library offers a set of methods that enable applications to exchange data with the NDN network. Figure 7 gives a high-level overview of the environment in which this solution was designed.

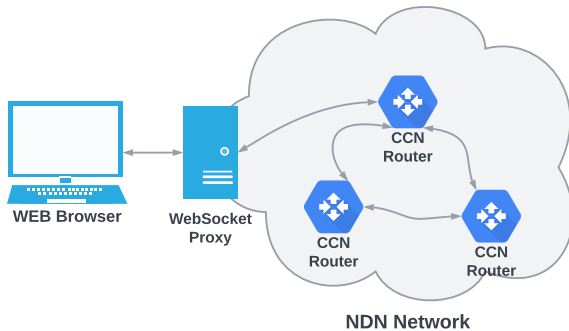


FIGURE 7 Browser add-on that communicates with a FIA network.

It could also be seen that solutions for interoperating Internet of Things (IoT) communications have been developed. Afterwards, several studies, including Gündoğan et al. (2018), Fotiou et al. (2017), and Gündoğan et al. (2020), propose a hybrid approach to enable communication using different protocols for IoT applications. Both of those solutions were designed to make possible the execution of CoAP, which, according to them, is a lightweight transfer protocol that provides interfaces for resource-oriented applications in an IoT environment; however, it relies on the IP network and its limitations that ICN aims to mitigate. Therefore, they would provide proxies responsible for translating IP-related packages into the specific ICN package.

This paper Gündoğan et al. (2018) proposes an intermediate layer to translate the CoAP and MQTT protocols into an NDN network. The authors explain that MQTT is a publish-subscribe protocol that uses a broker to route messages between the sender and the receiver. To evaluate the study, a setup was created with multiple sensors connected to an NDN network, which was then linked to an additional layer that translated into the IP network. The proposed solution involves an MQTT subscriber sending a message to a gateway, which then translates the request into an NDN package and routes it to the corresponding sensor connected to the NDN network.

On the other hand, when an NDN sensor sends a message to the gateway, it translates the message into an MQTT message, which is then forwarded to the appropriate MQTT subscriber. This interface layer has a table that maps MQTT topics to NDN names. When a subscriber subscribes to a topic,

the gateway retrieves the related NDN name. Similarly, when a sensor publishes data, it maps the subscribers who should receive the information. The authors' results demonstrate that this kind of interoperability can improve communication performance in the evaluated IoT environment.

In conclusion, several approaches enable interoperability between FIAs and the current Internet. These approaches provide an additional layer of communication, facilitating the evolution of the network by offering seamless connectivity and enabling applications to be executed regardless of the architecture. Additionally, some solutions aim to leverage current Web technology by creating libraries that allow connections to FIA, but do not provide a way to enable legacy applications to coexist. Therefore, a solution that combines the benefits of having an additional layer and enabling the use of modern Web tools would be the most appropriate, providing an architecture that can grow independently of the already used technology and that is backward compatible.

4.3 | SQ 3 - How can Web applications adapt to new architectures?

The following text will provide the details required for the current applications to run on new Internet architectures. As discussed by Siddiqui and Mueller (2012), and outlined in previous sections, the additional layer approach reduces application-level impacts, as it continues to function according to the same standards. The application will still consume an HTTP endpoint provided by a host-centric network by establishing a connection with the provided abstraction layer.

This proxy will be responsible for translating the request to a compatible FIA, as demonstrated in solutions like COIN Jahanian et al. (2020b) and PEP-DNA Ciko et al. (2021). Therefore, we can conclude that this solution is backward compatible and would only require changes to the URL being consumed, which must be directed to a specific proxy.

Furthermore, in the browser add-on approach, like NDN.js Shang et al. (2013), no changes would be required at the application level, as the HTTP request will be automatically translated and forwarded to the FIA. Hence, we can conclude that this approach not only keeps application compatibility but also implies changes on the client side, due to the need to have an extension installed.

If we consider a scenario in which a Web structure is developed, it can be expected that significant changes are required on the application side. Following this principle, some solutions aim to provide a new environment, such as NDNBrowser Wang and Qiao (2014) and IDNet Browser Jeon et al. (2015), which seek to develop a new browser using an open-source third-party solution provided by QTWebKit qtW (2022) to interpret HTML and JavaScript. Hence, it will provide a direct way to communicate with the NDN, as it is a comprehensive front-end solution.

As mentioned above and also according to the authors Wang and Qiao, the implementation of QT solutions would have a significant effect. Since it requires a complete redevelopment of the application to meet the desired FIA, as there will be no additional layer between the Web browser and the FIA, as illustrated in Figure 8. Furthermore, there would be restrictions as QTWebKit

[†] <https://www.npmjs.com/package/ndn-js>

does not offer the same features as the current Web languages and frameworks, which is evident in the study results. It is challenging to detect content updates following this approach, which would be a significant limitation in a constantly updating Web environment.

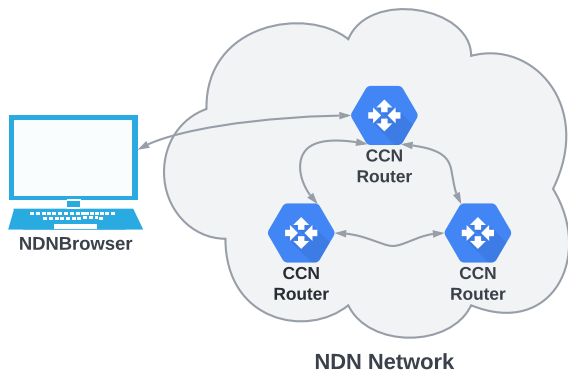


FIGURE 8 NDNBrowser developed under QTWebKit.

It is also worth noting that this framework has been discontinued, which could lead to a lack of support and updates, potentially resulting in further problems, including security vulnerabilities. It is also feasible to move existing applications to operate with solutions similar to NDN.js Shang et al. (2013), which has been discussed before in this article. However, this would require considerable implementation modifications, since existing API calls must be converted using the NDN libraries mentioned previously, to intermediate communication between the Web browser and the network, as illustrated in Figure 7.

According to the research conducted by Siddiqui and Mueller Siddiqui and Mueller (2012), the FIAs must be adjustable to meet the needs of the application. Therefore, the network should be able to communicate regardless of the technology used to build and deploy the solutions. It was suggested to create an intermediate API layer to facilitate the translation of packages between different architectures. This approach, as the authors argued, would allow the network to evolve without necessitating any modifications on the application side. This would break the link between the applications and the network protocols, resulting in a backward-compatible solution that would enable the presence of a multi-application environment. This solution is similar to COIN Jahanian et al. (2020b), PEP-DNA Ciko et al. (2021) previously described and illustrated in Figures 5 and 6, in which an intermediate layer was added between the different architectures, making it possible to establish the connection that is not relayed on current Internet protocols.

In summary, in an ideal situation, the network architecture should be able to adapt to any application, allowing the development of the foundation of any network without necessitating modifications to existing solutions. Furthermore, redeveloping applications to be executed in FIAs would require a significant transformation of the existing programs, which may not be feasible due to the amount of effort and resources required, as well as the frameworks

used in the applications that may only be available in the current environment in which the solutions are deployed.

4.4 | SQ 4 - How the future Internet Web could be developed?

To answer this question, we must consider that a new Web will be implemented directly in some FIA. Therefore, in this scenario, changes would be required in the browser, protocols used, and applications. Previously in this document, in the related work section 2, some solutions were presented to achieve this goal.

If we consider the development of new Web protocols, substantial modifications would be necessary for existing applications. Browsers are currently designed to work with protocols such as HTTP and HTTPS; hence, it would require the creation of a new Web browser, similar to NDNBrowser Wang and Qiao (2014) and Jeon et al. (2015).

These solutions were characterized by their ability to utilize a third-party mechanism for the interpretation of HTML and JavaScript. However, this approach has certain restrictions due to the technologies it supports, which may limit the ability to create applications that utilize the most recent Web technologies.

A potential way to overcome the restrictions mentioned is to modify existing open-source browsers, such as Chromium[†], which are equipped to handle the latest Web features. Geng et al. (2017) proposed a solution that alters the communication protocol used in these browsers by introducing a new architecture called DBCSUEE, which stands for "Docker-Based Chromium Sandbox Using Embedded Environment". This architecture utilizes Docker containers to isolate the processes of the Chromium browser. We believe that this customization concept could be used to provide a customization that would enable the current browser to communicate with FIA architectures.

On the contrary, NDNJs Shang et al. (2013), which have already been discussed, could enable current Web browsers to connect to FIA, as well as solutions that seek to provide interoperability, such as the one proposed by COIN Jahanian et al. (2020b). Thus, as has been discussed, there appear to be three main ways to develop the future Web: new browsers, extensions to existing browsers, and an additional layer to provide interoperability. Figure 9 illustrates current solutions that seek to integrate emerging features over FIAs in a mind map, which is systematically structured around different application domains. The objective of the map is to illustrate research conducted in various fields, with the intention of harmonizing them with the concept of the Future Internet.

4.5 | SQ 5 - What are the impacts expected from new Web architectures?

To answer this question, we must keep in mind that the current Web is designed to be executed under the existing TCP/IP network, and it is evolving since its creation back in the end of 1980s from a static structure to a dynamic

[†] <https://www.chromium.org/developers/>

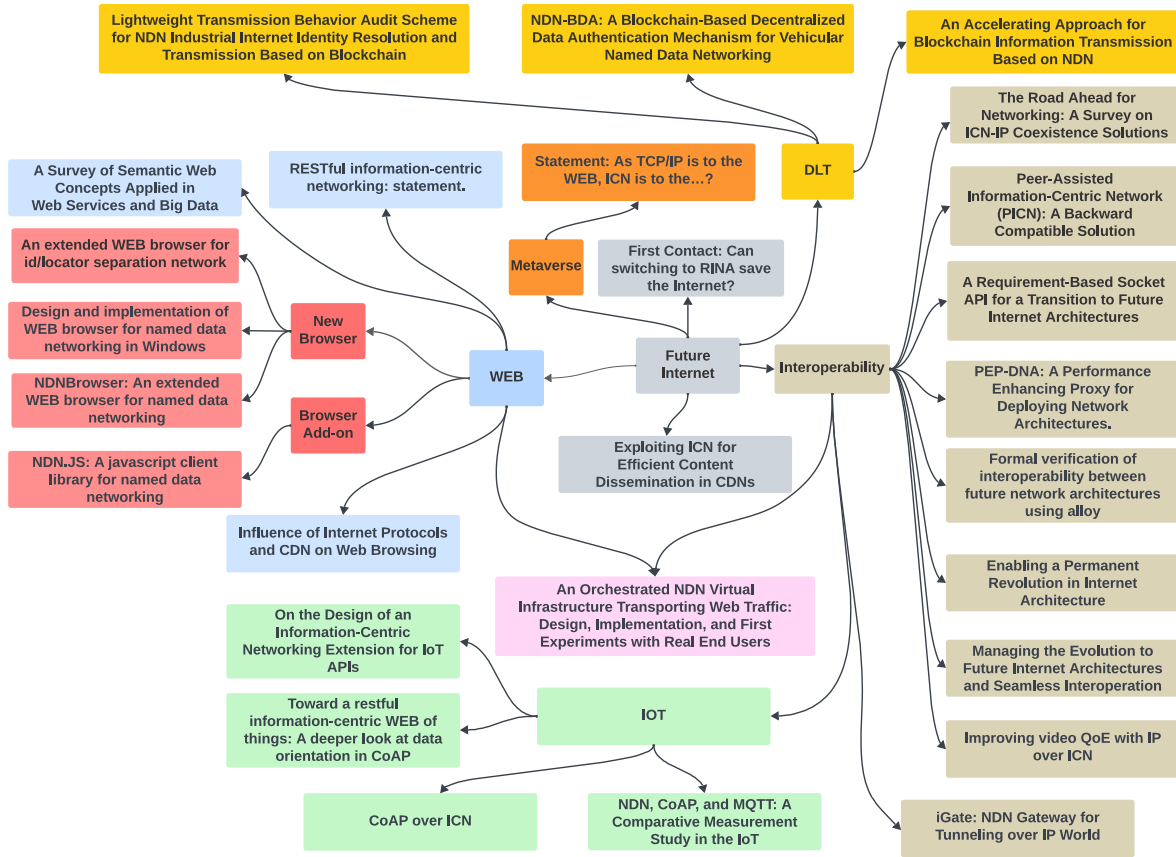


FIGURE 9 A Mind Map of Developed Solutions for FIAs.

and integrative behavior. This evolution emerges to an architecture in which is expected that the network gets improved considering scalability, performance, and security matters. Therefore, the goal of this SQ is to answer the questions that would be posed in a Web application environment deployed on an FIA.

Aiming to reach that, we synthesize findings from a set of studies, which demonstrates the impacts. Initially, it is foreseeable that resource identification will become independent from its physical location, as discussed by Jeon et al. (2015) and Conti et al. (2020). Furthermore, studies such as Saverimoutou et al. (2019) and Doyen et al. (2019), aims to demonstrate how content delivery and scalability could be affected due to that.

The evolution of the Internet is converging to a solution in which the architecture should be designed with the ability to transfer a large amount of data in the shortest time possible, as concluded by Jahanian et al. (2020b) and Alberti et al. (2019). Furthermore, it is expected that it includes the possibility of mobility; thus, a new Web developed under FIAs will benefit from its principals.

In the paper Siddiqui and Mueller (2012), the authors mentioned several benefits of CDNs in Web browsing, comparing the performance observed in different scenarios with HTTP, which is a standard protocol used in the current Internet, and QUIC Iyengar and Thomson (2021), an acronym for Quick UDP Internet Connections, which is a Google-developed transport layer protocol that was designed to reduce latency and enhance the reliability

of Web browsing through encryption, multiplexing, congestion control, and connection migration features. Therefore, we anticipate that the impacts of the solutions described in Table 4 will be extended to a future Web solution, including:

4.5.1 | Identification and location

In the current Internet, the IP address is used to determine the identification (ID) and location (LOC) of a host or server. In other words, IP addresses have dual semantics, which is mentioned in the studies Jeon et al. (2015) and Conti et al. (2020). The authors suggest this could result in limitations in content delivery, scalability, mobility, and availability, in which the benefits will be discussed in the following items. Furthermore, the authors highlighted several advantages related to flexibility, caching, replication, and content distribution. Such benefits are essential for novel Web Services, allowing them to decouple entities from their physical connections and provide dynamic adaptation to a locator for delivery.

4.5.2 | Content delivery

In traditional host-centered networks, data is delivered based on the server's address. On the other hand, in content-oriented networks, content is fetched based on its name, not depending on where the data is hosted. The authors Saverimoutou et al. observed in their study Saverimoutou et al. (2019) that the use of CDNs will have a significant impact on the time that pages are delivered. They found average load time would be reduced 43.1% for HTTP and 38.5% for QUIC. This improvement is a result of the cache mechanism present in content-oriented networks, which reduces the distance that data need to travel and consequently reduces the delivery time. Similarly, as observed in the paper Doyen et al. (2019), a caching mechanism can reduce network traffic, resulting in improved response times for end users.

4.5.3 | Scalability

As the number of Web requests increases, a scalable solution can be advantageous for customers who desire a prompt response from the page they have requested. According to Doyen et al. (2019) and Saverimoutou et al. (2019), content-oriented networks provide well-designed cache systems that enable data to replicate throughout the network, resulting in faster responses even in situations with a large number of content providers and consumers.

4.5.4 | Mobility

Modern Web applications require the assurance of continuous access and the capacity to acquire content regardless of the physical location or network connection of the network elements. In Kutscher and Oran (2022), the author argues that ICN can facilitate mobility by enabling users to change their positions without compromising their application context. Moreover, the paper states that servers can also be relocated and remain accessible to customers. Therefore, in a content-oriented network, content is retrieved by its identification, rather than relying on fixed URLs or server addresses, allowing content to be easily replicated and distributed to the end user. Additionally, since the content can be stored separately from its location, it will result in a decrease in latency, thereby improving the delivery of the content. Furthermore, it allows them to move between different locations or connections, for example, by transitioning from WiFi to mobile data, and have the requests automatically rerouted to the most suitable resource. Therefore, this capability will provide continuous content availability and a better user experience regardless of the connected network.

4.5.5 | Availability

Web content must be accessible to users without interruptions or outages, allowing them to access it consistently. According to studies Chen et al. (2016) and Saverimoutou et al. (2019), we can conclude that availability is a key feature of content-oriented networks. This is achieved through content replication between multiple resource providers by creating copies of content in

multiple geographic locations, as well as through cache mechanisms. Furthermore, the authors claim that a CDN solution can improve content availability by reducing latency, bandwidth consumption, and server load, as well as increasing cache hit ratio and resilience to network failures. This is because the content remains accessible even when a server node became unavailable, as it could be stored in multiple content providers. In conclusion, availability will have a significant impact on user trust and satisfaction, as users can access data without interruption.

4.6 | SQ 6 - How to interoperate with AI?

Generative Artificial Intelligence Feuerriegel et al. (2024) is a topic of high relevance in both academic and business environments, due to its disruptive potential in various domains, which could enable the creation and evaluation of content. Currently, this technology is used for content generation, which can include text, images, videos, sounds, and other possible applications. Additionally, it is applied to content evaluation. Nonetheless, these solutions are developed to work within the existing Internet architecture, adhering to its protocols. Consequently, pre-existing solutions such as ChatGPT OpenAI Platform Docs (2024) and Gemini Google Gemini API (2024) cannot operate on FIAs. Given this context, the response to SQ 6 aims to demonstrate how FIAs can interact with generative AIs and outline the advantages that might be gained by allowing these solutions to interoperate across various networks.

As reported by the authors Feuerriegel et al. and Nah et al., the use of Generative AI will result in several benefits, which are leading the world to an unpredictable transformation of the digital landscape. The potential advantages offered by Generative AI are listed in Table 5, which elucidates the variety of ways in which Generative AI can positively contribute to enhancing user experiences, increasing productivity and efficiency, and unlocking creative potential in the context of Web-related subjects.

Integrating this technology into the core of an Internet architecture fosters the development of a more dynamic, intelligent, and user-focused setting. According to what was mentioned by Karapantelakis et al., convergent ICT architectures could benefit from using Generative AI in several ways:

1. Enhanced Network Management and Optimization:

- (a) **Improved Radio Map Estimation:** In the context of wireless communication, artificial intelligence can be leveraged to develop more precise radio maps from limited measurement data. This approach facilitates the identification of network blind spots, optimizes cell placement, and enhances overall coverage and capacity.
- (b) **Dynamic Resource Allocation:** Data traffic could be analyzed, and based on that, generative models could be applied aiming to identify patterns to perform efficient resource allocation, which optimizes network efficiency, reduces congestion, and consequently improves user experience.
- (c) **Intelligent Error Recognition and Troubleshooting:** Network logs could be provided to an AI model, which could analyze to identify anomalies and predict potential failures, thus facilitating proactive maintenance measures and capabilities for self-repair.

2. Next-Generation Communication Paradigms:

TABLE 5 Benefits of Generative AI

Category	Benefit	Description
Elevating User Experiences	Personalized Content and Services	Tailoring digital experiences to individual needs and preferences, making online interactions more engaging. Feuerriegel et al. (2024), Nah et al. (2023) Examples: Adaptive news feeds, personalized learning, customized marketing.
	Human-like Interaction	Enabling natural and intuitive interaction with information systems through conversational agents. Nah et al. (2023) Example: ChatGPT for streamlined communication.
Boosting Productivity and Efficiency	Automating Routine Tasks	Freeing up human capacity for higher-level tasks by automating repetitive processes. Feuerriegel et al. (2024), Nah et al. (2023) Examples: Report generation, document summarization, language translation.
	Enabling New Forms of Automation	Powering intelligent automation systems that provide context-aware support for enhanced efficiency. Feuerriegel et al. (2024) Example: AI-driven process guidance systems.
Unlocking Creative Potential	AI-Assisted Content Creation	Assisting humans in creative tasks, fostering innovation and engaging outcomes. Feuerriegel et al. (2024) Examples: Music composition, art generation, product design.
	Supporting Design Science Research	Accelerating innovation by assisting in knowledge retrieval for new IT artifact creation. Feuerriegel et al. (2024)

(a) **Semantic Communication:** According to the authors Karapantelakis et al., this is an emerging concept in which the architecture will focus on information rather than raw data, which means that the network will strive to understand and convey the intended significance of the data rather than blindly transmitting every bit. Generative AI can be utilized to convey precise semantic representations of information, thereby reducing bandwidth requirements and enabling more efficient and intelligent communication.

(b) **Personalized Content Delivery:** Furthermore, Karapantelakis et al. observed that generative models possess the potential to produce personalized content for users, tailored to their individual preferences and contextual factors. This application could significantly enhance the array of services provided, potentially encompassing Extended Reality (XR) experiences, by customizing data streams to meet specific user requirements.

Existing generative AIs are integrated into a Web environment by utilizing the provided APIs, which offer a standardized way for different software systems to communicate and exchange data, making them the backbone of AI integration on the Web. Those interfaces could be provided in different ways, which include:

- **Cloud-Based AI Services:** Several leading tech companies are offering cloud-based AI services, including Google Cloud AI Google AI (2024), Amazon AWS Machine Learning AWS Machine Learning (2024), and Microsoft Azure AI Microsoft Azure (2024). These services provide access to pre-trained AI models and powerful computing resources through their APIs.
- **JavaScript Libraries and Frameworks:** A client-side AI application could be developed to process requests directly on the client's Web browser. This development can be achieved by utilizing dedicated

JavaScript libraries and frameworks, which provide a set of tools for creating and consuming AI models. Examples of platforms offering these features include TensorFlow.js TensorFlow.js (2024) and Brain.js Brain.js (2024).

- **Server-Side AI Integration:** Back-end applications could have their own AI models, which could be developed using libraries to be consumed by their programming language, such as TensorFlow TensorFlow (2024), which offers a suite of libraries for use in Python, Java, C, and GO. Those applications could provide their APIs for consumption by other applications.
- **Direct Model Integration:** The utilization of WebAssembly (Wasm) WebAssembly (2024), which is a portable compilation solution which seeks to provide a compact binary format for applications alongside a secure execution environment, thereby facilitating the execution of AI models directly within web browsers without the necessity for any JavaScript libraries or server-side applications. This method promises significant performance improvements and reduced latency for AI-powered web applications, enhancing the user experience and enabling more complex real-time AI features. **According to the results presented in Khelifa et al. (2024), the effectiveness of Wasm runtimes in handling computationally demanding AI tasks remains an area of active exploration and study among researchers.** However, promising results have been demonstrated for smaller, optimized neural networks, and substantial overhead can be observed for larger, more complex models, which makes it clear that further optimization of this solution is needed.

Given the scope of our investigation, an integral part involved the pursuit of methodologies to harness AI solutions already in existence within the framework of Future Internet Architectures. Despite this focus, our review of the literature revealed a notable lack of research specifically targeting the exploitation of these AI technologies in this domain. To address this gap, we

embarked on a detailed examination of methods for interacting with existing Generative AI APIs.

Our objective was to discover methods that facilitate seamless AI integration into the FIA infrastructure, thereby enhancing its functionalities. In particular, APIs from advanced models, such as ChatGPT OpenAI Platform Docs (2024) and Gemini Google Gemini API (2024), offer significant potential in augmenting the capabilities of varied FIA components. This advancement could pave the way for developing innovative AI-driven services within the ICT architecture itself or its Web support, thus expanding its utility and efficiency.

Therefore, the techniques presented in the previous sub-question (SQ 2), which aim to provide interoperability, can be applied here to offer a solution for bridging the gap between existing Generative AI APIs and the diverse network architectures envisioned in FIAs. Furthermore, AI would also be developed naively, following the other approaches presented in this section, such as adding additional libraries to the architecture's source code to enable the creation of customized models. By exposing this feature in the network architecture, developers will be able to create a wide range of innovative applications and services that can go beyond the enhancements proposed by Karapantelakis et al., leading to a more intelligent, efficient, and user-friendly Internet navigation experience.

4.7 | SQ 7 - How to interoperate with DLTs?

Another key aspect to leverage the benefits of FIA would be its integration with other emerging technologies, including its interoperability with existing Distributed Ledger Technologies (DLTs) 101 (2023). Initially, it is necessary to understand that DLTs offer potential benefits in secure, transparent, and decentralized data management, as explained by Panwar and Bhatnagar, which aligns with the concepts of the Future Internet presented in this study. Therefore, question SQ 7 aims to explore how the convergence of FIA and DLTs can unlock novel functionalities and address persistent challenges in realizing the full potential of decentralized network solutions.

However, a critical technical hurdle exists: the overwhelming majority of established DLT ecosystems are designed to operate over the TCP/IP protocol stack. Consequently, simply deploying existing DLT nodes onto an FIA network is not feasible without modification. The primary objective of this analysis is, therefore, interoperability mechanisms allows these TCP/IP-based DLT solutions to be executed within Future Internet Architectures. By bridging this gap, FIAs can leverage the robust, mature security and consensus mechanisms of current DLTs without necessitating a complete redevelopment of these complex distributed systems.

Undoubtedly, the most recognized DLT is Blockchain Panwar and Bhatnagar (2020), which provides a revolutionary way of managing and protecting data. Several existing applications are leveraging this technology, including finance solutions, supply chain management, healthcare, and government services. The main reason for those solutions to adopt a DLT implementation is due to some main capabilities it provides, such as:

- **Decentralization:** Centralized models introduce a potential single point of failure that may jeopardize the availability of the solution. The fundamental principle of DLT involves disseminating the ledger among network participants, ensuring that all maintain a copy of the required data Panwar and Bhatnagar (2020). This distributed framework eliminates the need for a centralized intermediary, enabling direct interaction between users and promoting a more democratic and robust system.
- **Immutability:** As per what was explained in Panwar and Bhatnagar (2020), all transactions recorded on a DLT became immutable, which means that it would not be possible to alter or delete them, which guarantees the integrity of the data.
- **Transparency:** DLTs are open and auditable for all network participants, since they provide a set of public records related to each transaction. This does not imply that all transaction details are visible. However, the fundamental architecture of the ledger and the record of previous transactions are accessible to all participants Panwar and Bhatnagar (2020).
- **Security:** Cryptography mechanisms are applied aiming to secure data, making it difficult for malicious actors to compromise the information present in the ledger, according to what the authors Panwar and Bhatnagar, Yang et al., and He et al. mentioned in their studies.

Given the capabilities of DLTs, it can be asserted that they are congruent with the benefits of the Future Internet presented before in this document. Specifically, they address the growing demand for secure and efficient data management, enhancing user empowerment and fostering trust within a decentralized online environment, as per the Table 6. In addition, DLTs are completely changing the Web landscape, creating what is called Web 3.0 Alabdulwahhab (2018).

We can anticipate some solutions that may emerge when combining FIA and DLTs, as explored by Yang et al., He et al., and Benmoussa et al., respectively. Firstly, Yang et al. seek an innovative approach that aims to mitigate the inefficiencies of blockchain transmission in a traditional TCP/IP network. Due to the broadcasting nature of DLTs, they rely on high overhead and significant transmission delays when running on the current Web. The authors' idea is to develop a solution that would benefit the blockchain from the content-centric behavior of FIAs. They introduced the concept of **Acceleration Information Transmission Strategy (AITS)**, i.e., a feature built upon NDN. The primary advancement of AITS is the deliberate selection and deployment of secondary nodes in the NDN network. The nodes are selected based on their significance, which is determined by evaluating three essential criteria:

- **Node Degree (Q_j):** The number of paths that intersect at a particular node, reflecting its level of connectivity within the network.
- **Transmission Capacity (T_j):** The intrinsic ability of the node to transmit data, which encapsulates its bandwidth and computational capacity.
- **Communication Weight (W_j):** The importance of the node's connections is emphasized by the weight attributed to the data links associated with it.

The importance of node j (l_j) is then calculated using a weighted sum of these factors, as shown in Equation 1:

$$l_j = \alpha_1 Q_j + \alpha_2 T_j + \alpha_3 W_j \quad (1)$$

TABLE 6 Shared Benefits of DLTs and the Future Internet.

Benefit	DLT Contribution	Future Internet Enhancement/New Benefit
Decentralization	Offers decentralized ledger systems, thus removing the need for central authorities for value transfer Panwar and Bhatnagar (2020).	Enhanced: As previously discussed, the FIA endeavors to establish a genuinely decentralized Internet, thereby enhancing user autonomy concerning data management and online interactions. This approach augments resistance to censorship and mitigates vulnerabilities associated with single points of failure.
Security	Utilizes cryptographic techniques to protect data and transactions, thereby minimizing the chances of manipulation and deceit He et al. (2023), Benmoussa et al. (2023).	Enhanced: Enhances the security of FIA by providing an integrating trust and accountability within the infrastructure itself, thereby diminishing dependency on susceptible centralized systems.
Transparency	Provides accessible and verifiable records, promoting transparency Panwar and Bhatnagar (2020).	New: By embedding this feature into FIAs, it will then enable transparent governance models for network infrastructure and service providers, increasing user trust and promoting ethical data practices.

where α_1 , α_2 , and α_3 are weights assigned to each factor, reflecting their relative importance.

The quantification of secondary nodes is determined based on the probability (P) of a node forming connections with other nodes within the network, using a probability-based framework. This approach incorporates the probability distribution of the node's transmission capacity and communication weight, thereby enhancing the model's adaptability to diverse network topologies and conditions. The model also employs a data caching mechanism at the secondary nodes. Furthermore, the miner nodes responsible for calculating the results of blockchain transactions store their data packets at these strategically placed secondary nodes. This approach enables other nodes to request the same data directly by querying the secondary nodes, thereby eliminating redundant transmissions and leading to a significant reduction in overall data overhead.

To validate their proposal, the authors tested their approach using an NDN simulator called ndnSIM Mastorakis et al. (2015). The superior performance of AITS was demonstrated by comparing its results to those of both TCP/IP networks and standard NDN (without AITS). In general, they observed that AITS provided significant reductions in transmission delay and data overhead. Such work offers a first glimpse at an original research opportunity that aims to complete the merging of DLTs on the FIA design itself, a research path that remains entirely unexplored for date.

In addition to this, the solution proposed by He et al. aims to enhance the privacy associated with the identification resolution and transmission process in NDN, particularly in the context of the Industrial Internet of Things (IIoT). To achieve this goal, the authors identified a set of key vulnerabilities to be mitigated.

- **Routing Trust Issues:** According to the authors, NDN suffers from what are called semi-honest and malicious nodes, which could result in risks to the integrity and dependability of the routing mechanism. Semi-honest nodes are known for following the protocol, which means that as part of their role, they extract sensitive data from packets, such as names of content or paths for routing. In contrast, malicious nodes intentionally undermine the network by dropping or altering packets. These actions can

jeopardize both cached data, which is stored in routers to increase efficiency, and computation results, which are crucial for making routing decisions and managing congestion. As a result, ensuring the trustworthiness and integrity of received data becomes difficult, emphasizing the necessity for a secure and trust-reliant mechanism, such as a DLT, for example.

- **Verification Overhead:** The article He et al. (2023) describes that the dependency of NDN on digital signatures for data authentication and integrity verification increases security; however, it implies a significant computational overhead, which could be considered a handicap that could lead to unacceptable latency and hinder the overall performance of the network.
- **Data Leakage Risks:** Data transmitted in an industrial environment is sensitive due to the potential impact on critical infrastructure, operational processes, and proprietary information. The authors He et al. claim that traditional NDN implementations often lack robust access control mechanisms and privacy preservation techniques, making it challenging to prevent unauthorized access and data breaches.

After pointing out the items described before, the authors He et al. proposed the use of blockchain Panwar and Bhatnagar (2020) to mitigate the main problems that hinder the resolution and transmission of secure and efficient identification in industrial NDN networks. Utilizing the fundamental characteristics of decentralization, immutability, and transparency inherent to blockchain technology, the authors propose a framework that aims to establish a reliable and verifiable platform for managing routing information and device behavior. To achieve that, three components were designed:

- **Blockchain as a Trusted Platform:** Aiming to address the trust issues in NDN routing, the authors propose the use of a permissioned blockchain as a trusted third-party platform. Permissioned blockchains are recognized for providing a controlled environment suitable for enterprise use cases. Unlike public blockchains, which allow anyone to join and participate, permissioned blockchains require participants to be identified and granted access. The blockchain functions as an immutable and secure repository for routing information, thereby eliminating the need for

a central authority. Replicating this distributed ledger across multiple nodes can ensure the integrity and availability of the data. Every transaction, including updates to routing tables and device activity logs, is safeguarded with cryptography and verified by consensus among the participating nodes, thereby blocking unauthorized changes or disruptive tampering. This establishes a reliable foundation for managing routing information, promoting secure communication and collaboration among industrial devices.

- **Improved Bloom Filter for Data Compression:** To address verification overhead, the authors employ an improved Counting Bloom Filter (CBF) He et al. (2023) for data compression, which is a probabilistic data structure known for its space efficiency and ability to support membership queries. Unlike traditional authentication methods that rely on computationally expensive digital signatures, the CBF allows for verification through lightweight bitwise operations. Consequently, employing this approach significantly reduces the computational load required to verify data authenticity and integrity.
- **Smart Contract-Based Auditing:** By implementing a scheme that leverages smart contracts, it allows for a transparent and automated auditing environment. Each contract encapsulates information that enables the analysis and verification of compressed behavioral data submitted by consumers. Additionally, once it is deployed on the blockchain, it becomes auditable, allowing for the tracking and verification of the auditing process's execution. This transparency ensures that all verification steps are performed according to the predefined logic and that any discrepancies or potential security violations are identified and recorded. This immutable and publicly accessible audit trail improves accountability, increasing trust between stakeholders and providing verifiable evidence for regulatory compliance or dispute resolution. In addition, the use of smart contracts for audit reduces reliance on manual processes, minimizing the potential for human error, and enabling more efficient and consistent verification throughout the NDN industrial network.

The lightweight proposal presents itself as a promising solution to enhance security in the Industrial Internet of Things. The experiments conducted by the authors present compelling results, demonstrating their effectiveness in mitigating common NDN security threats, such as Interest Flooding and Black Hole attacks, while minimizing computational overhead. Specifically, their findings showcase the scheme's ability to detect and prevent these attacks, ensuring the integrity and availability of the network.

Furthermore, another study aims to enhance the authentication mechanism in a NDN network, called NDN-DBA Benmoussa et al. (2023), which stands for Named Data Networking - Blockchain-Based Decentralized Data Authentication. On the one hand, lightweight solutions have been developed to enhance critical security in industrial solutions. On the other hand, NDN-DBA focuses on addressing the specific challenges of vehicular networks. These networks have a highly dynamic nature, and therefore, communication with low latency is required to provide a more efficient approach to data authentication.

To achieve this, they utilize blockchain's decentralized architecture to enable local data verification, thereby reducing reliance on a central authority

and minimizing delays associated with traditional certificate chain verifications. This focus on efficiency and real-time performance makes NDN-BDA particularly well-suited for time-sensitive vehicular applications, such as sharing safety messages, where timely data validation is crucial for preventing accidents and ensuring road safety. To do this, the authors proposed a solution that is composed of some key components and addresses some aspects of the decentralized authentication process:

- **Centralized Blockchain:** Aiming to store and manage the certificates generated from each Producer Vehicle (PV), the authors developed a centralized blockchain managed by what they called the Management Server (MS). This blockchain functions as a secure and tamper-resistant repository, thereby guaranteeing the authenticity and integrity of the certificates.
- **Hierarchical Trust Schema:** In the established scheme, trust is structured with the Vehicle Registration Entity (VRE) placed as the foundational root of trust. The authority is then delegated from the VRE to the System Certification Authority (SCA) and the MS. This delegation forms a comprehensive and organized chain of trust, which is crucial for the issuance and management of certificates within the system.
- **Producer Registration Process:** There is a procedure that ensures that only authorized vehicles can obtain certificates and participate in data exchange. The implemented mechanism verifies vehicle ownership and identity through communication with the VRE and generates a unique key for each producer. The registration request includes the vehicle owner's Citizen ID (CID) and the vehicle's unique registration ID (VID). This information is securely transmitted to the Management Server (MS), which then initiates a verification process with the VRE to confirm the association between the CID and the VID.
- **Data Authentication Process:** Aiming to request data, consumer vehicles must first obtain the current blockchain from the MS. This procedure ensures they have the latest validated certificates for all registered producer vehicles. As part of the packet received by the consumer, it uses a field called keyLocator, which acts as a pointer to the producer's certificate stored on the blockchain. In instances where the certificate is located, the data are considered authenticated and consequently accepted. In contrast, if the certificate is not found, the data is rejected, thereby avoiding the handling of information that could be harmful or not properly authenticated. This decentralized strategy uses the blockchain as a reliable storage system while facilitating local verification. As a result, it significantly reduces the communication load and decreases latency, making it highly appropriate for real-time applications within vehicular networks.

Taking into account the components presented before, Figure 10 reproduces a sequence diagram containing interactions between the Producer Vehicle (PV), the Management Server (MS), the Vehicle Registration Entity (VRE), and the System Certification Authority (SCA) during the vehicle registration process:

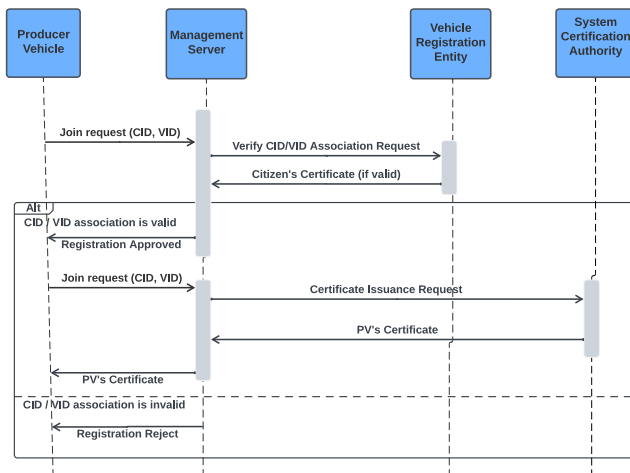


FIGURE 10 NDN-DBA PV registration sequence diagram.

In addition to the solutions presented above, public DLTs such as IOTA Silvano and Marcelino (2020)[‡] it is also possible to integrate a system with a distributed ledger through a set of standardized APIs, creating then a pathway that enables its consumption by third-part solutions, which consequently enables its incorporation with Future Internet Architectures, following the approaches described at SQ 2. According to Chiti and Gandini, this approach aligns perfectly with the core principles of Web 3.0: decentralization, user empowerment, and semantic Web functionalities. As a result, by adopting this approach, FIAs can reap the benefits offered by DLTs while avoiding the need to build and maintain their own intricate DLT infrastructure, which aligns with the descriptions provided in previous SQs, ensuring a seamless network evolution that leads to gradual and harmonious development.

Therefore, based on the solutions presented by the authors Yang et al., He et al., and Benmoussa et al. before, we can conclude that FIAs could benefit from DLT, specifically blockchain, by improving their security, efficiency, and scalability in various application scenarios. Due to the blockchain's working mechanism, as described previously, new Internet activities can leverage its decentralized, immutable, and transparent nature to address key challenges associated with trust, data integrity, and resource management. Furthermore, it is conceivable to advance the implementation of DLTs as a fundamental component of FIAs. Specifically, this involves developing FIAs that incorporate DLT construction methodologies within their Internet solutions. Similarly, the web aspect of these propositions could also be underpinned by the principles of DLTs.

4.8 | SQ 8 - What are the challenges and open questions?

In addition to the advantages of using FIAs in Web browsing, there are still a few difficulties and unresolved questions that could be explored in future

research. One of the main issues raised by several studies, including Jahanian et al. (2020b), Doyen et al. (2019), and Conti et al. (2020), is related to the implications of security and privacy.

Moving from a host-centric network to a content-centric one could potentially introduce new vulnerabilities and threats; therefore, it is essential to address these issues in future studies and develop the necessary mechanisms to mitigate them. Furthermore, Jahanian et al. mentioned that in an interoperable scenario, security becomes even more challenging, since multiple architectures and protocols can be executed over the same environment, each with its security mechanism and specific requirements. Establishing connections between different architectures and ensuring they comply with their respective security specifications can be a challenging task.

Similarly, studies Jahanian et al. (2020b), Siddiqui and Mueller (2012), and Jahanian et al. (2020a) have also addressed the scalability challenge, which can be exacerbated by the number of different architectures and protocols. The authors noted that as the number of network architectures and protocols grows, scalability becomes more challenging to achieve.

Each architecture has its own scalability needs and methods, and ensuring that they are compatible with one another while maintaining scalability is a complex undertaking. Furthermore, in the paper Wang and Qiao (2014), the authors evaluated their proposal in NDN and noted that it is designed to handle large networks. However, they point out that there are still unresolved questions about how efficient a route and cache would be in large-scale networks. The authors are concerned about the sudden expansion of the cache system, which could lead to a significant increase in storage and bandwidth requirements.

Additionally, as the number of nodes in the network increases, the size of the routing table can also become larger, leading to additional costs and decreased performance. Consequently, it is essential to assess thoroughly, in future studies, the effect of scalability incorporated into an interoperable solution.

The adoption of a content-centric approach to the Internet presents a challenge, as noted by Doyen et al.. The transition would require a significant amount of time and resources. Therefore, it is essential to involve researchers and the communications industry to make it happen. In addition, a pattern for interoperability proxies should be established to allow them to be generic and interoperable between different architectures, as suggested in Siddiqui and Mueller (2012). This would result in more independent applications, and Jahanian et al. (2020a) suggests that a formal method should be used to identify potential errors and inconsistencies before and during the deployment of any system. Furthermore, if the development of new applications, following the specific FIA constraints, would be challenging, as discussed in Wang and Qiao (2014), given that it would be necessary to redesign the current application to follow the new models and tools, which could become a barrier to the adoption of those solutions.

5 | FINAL REMARKS

In this work, our objective was to present a detailed review of the interoperability of the Web between different Internet architectures. First, we

[‡] Tangle is a DLT that does not rely on blockchain's traditional chain-of-blocks structure, instead, it employs a Directed Acyclic Graph (DAG) known as the Tangle

described what content-oriented Internet architecture is and the limitations of host-centric networks. Then, we presented several studies that aim to be an alternative to the current Internet, such as NG, RINA, XIA, NDN, and MB. Subsequently, we present the objective of our study, which is to evaluate the potential benefits of executing Web applications in a cross-platform environment.

Our analysis revealed a consensus on the benefits of stitching to a content-centric network architecture, which will impact the user experience when consuming data, including Web navigation. Therefore, the implementation of APIs to enable Web interoperability between the current and future Internet architectures will bring about a solution that joins the existing frameworks and programming languages with the advantages provided by FIAs. Consequently, it was observed that such a solution enables the existing application to be executed in a heterogeneous environment with multiple architectures, without any development changes. This solution allows smooth network evolution, resulting in minimal impact on both the client and server sides.

Additionally, the Web field was observed as an area that presents a vast and promising opportunity for new research and innovation. First, we discovered that there are not many studies in this area; thus, there is a vast scope for further research in this field, which will result in innovative solutions that will benefit future studies aiming to provide similar solutions. Additionally, by exploring Web approaches for novel architectures, it will not only contribute to service providers by providing a solution that allows the network to evolve smoothly, but will also primarily impact data consumers, as user experience will be enhanced due to improvements in scalability, availability, mobility, and latency.

However, we could observe that none of the existing solutions was applied to enable interoperability for allowing the integration of existing real-world Web applications within Future Internet architectures. This bottleneck could be mitigated in a future study, aiming to generically enable seamless communication between any type of application across different network architectures. Our analysis is intended to guide researchers and developers in further exploring this area, particularly in developing this feature for NovaGenesis. As discussed in the text, NovaGenesis currently lacks any interoperability features, although its architecture shows promise for enabling such integration.

Overall, our review emphasizes the importance of providing solutions to interoperate Web applications in future Internet architectures, and how servers and consumers can benefit from the advantages offered by this novel solution. Given the increasing importance of the Web in day-to-day life, we can conclude that it is critical to develop Internet solutions that aim to accommodate the needs of both users and providers simultaneously. Therefore, we believe that this study will serve as a valuable reference for future research in this rapidly evolving field. It is becoming increasingly evident that the current Internet architecture will eventually require replacement. This article addresses the integration of WWW technologies within the framework of emerging proposals. We advocate for the preservation of existing Internet policies while incorporating a diverse array of technologies. The implications of this research on the future evolution of Internet architecture are profound and far-reaching.

ACKNOWLEDGMENT

This work was funded by RNP, with resources from MCTIC, Grant No. 01245.020548/2021-07, under the Brazil 6G project of the *Centro de Referência em Radiocomunicações* (CRR) of the *Instituto Nacional de Telecomunicações* (Inatel), Brazil. The work has also been partially funded by *Fundação de Amparo à Pesquisa do Estado de Minas Gerais* (FAPEMIG RED-00194-23). This work was carried out with support from CNPq, National Council for Scientific and Technological Development - Brazil (303382/2021-0). It was also supported by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES) - Finance Code 001. The authors also thank CNPq, and Vivavox Telecom.

REFERENCES

- (2022) *Qt WebKit - Qt Wiki*.
URL https://wiki.qt.io/Qt_WebKit
- (2022) *What is the Linux kernel?*
URL <https://www.redhat.com/en/topics/linux/what-is-the-linux-kernel>
- (2023) IEEE Standard for Distributed/Decentralized Exchange Framework using Distributed Ledger Technology (DLT). *IEEE Std 2140.4-2023*, 1–19. doi:10.1109/IEEESTD.2023.10101743.
- Ahlgren, B. et al. (2012) A survey of information-centric networking. *IEEE Communications Magazine*, 50(7), 26–36. doi:10.1109/MCOM.2012.6231276.
- Alabdulwahhab, F.A. Web 3.0: The Decentralized Web Blockchain networks and Protocol Innovation. In: *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*, 2018, pp. 1–4.
- Alberti, A.M. & Singh, D. Developing a NovaGenesis architecture model for service oriented future Internet and IoT: An advanced transportation system scenario. In: *2014 IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 359–364.
- Alberti, A.M. et al. (2019) Advancing novagenesis architecture towards future internet of things. *IEEE Internet of Things Journal*, 6(1), 215–229. doi:10.1109/JIOT.2017.2723953.
- AWS Machine Learning (2024) *AWS Machine Learning*. <https://aws.amazon.com/ai/machine-learning/>, accessed: October 28, 2024.
- Banafa, A. (2023) *5 Internet of Things: Myths and Facts*. pp. 23–28.
- Benmoussa, A. et al. (2023) NDN-BDA: A Blockchain-Based Decentralized Data Authentication Mechanism for Vehicular Named Data Networking. *Future Internet*, 15(5). doi:10.3390/fi15050167.
- Bhanage, G. et al. Storage-aware routing protocol for the mobilityfirst network architecture. In: *17th European Wireless 2011 - Sustainable Wireless Technologies*, 2011, pp. 1–8.
- Brain.js (2024) *Brain.js*. <https://brain.js.org/>, accessed: October 28, 2024.
- Burke, J. Statement: As TCP/IP is to the Web, ICN is to The? In: *Proceedings of the 9th ACM Conference on Information-Centric Networking. ICN '22, 2022*. New York, NY, USA: Association for Computing Machinery, p. 153155.
- Chen, J. et al. Exploiting ICN for Efficient Content Dissemination in CDNs. In: *2016 Fourth IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, 2016, pp. 14–19.
- Chiti, F. & Gandini, G. (2023) Distributed Ledger as a Service: A Web 3.0-Oriented Architecture. *Journal of Sensor and Actuator Networks*, 12(4). doi:10.3390/jsan12040057.
- Ciko, K. & Welzl, M. First Contact: Can Switching to RINA save the Internet? In: *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 2019, pp. 37–42.
- Ciko, K., Welzl, M. & Teymoori, P. PEP-DNA: A Performance Enhancing Proxy for Deploying Network Architectures. In: *2021 IEEE 29th International Conference on Network Protocols (ICNP)*, 2021, pp. 1–6.

- Conti, M., Gangwal, A., Hassan, M., Lal, C. & Losiouk, E. (2020) The Road Ahead for Networking: A Survey on ICN-IP Coexistence Solutions. *IEEE Communications Surveys & Tutorials*, 22(3), 2104–2129. doi:10.1109/COMST.2020.2994526.
- Day, J.D. (2010) *Patterns in network architecture: A return to fundamentals*. : Prentice Hall.
- Doumanis, I. et al. (2018) Improving video QoE with IP over ICN. *International Journal of Network Management*, 30, e2057. doi:10.1002/nem.2057.
- Doyen, G. et al. (2019) An Orchestrated NDN Virtual Infrastructure Transporting Web Traffic: Design, Implementation, and First Experiments with Real End Users. *IEEE Communications Magazine*, 57(6), 33–39. doi:10.1109/MCOM.2019.1800730.
- Feuerriegel, S. et al. (2024) Generative AI. *Business & Information Systems Engineering*, 66(1), 111–126. doi:10.1007/s12599-023-00834-7.
- Fotiou, N. et al. CoAP over ICN. In: *IFIP International Conference on New Technologies, Mobility and Security (NTMS), 07 2017*.
- Geng, X. et al. (2017) An novel architecture and inter-process communication scheme to adapt chromium based on docker container. *Procedia Computer Science*, 107, 691–696. doi:https://doi.org/10.1016/j.procs.2017.03.149.
- Google AI (2024) *Google AI Products*. https://cloud.google.com/products/ai, accessed: October 28, 2024.
- Google Gemini API (2024) *Gemini API Documentation*. https://ai.google.dev/gemini-api/docs, accessed: October 28, 2024.
- Grewé, D. et al. On the Design of an Information-Centric Networking Extension for IoT APIs. In: *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, 2021, pp. 1–6.
- Gündoğan, C. et al. NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT. In: *Proceedings of the 5th ACM Conference on Information-Centric Networking. ICN '18, 2018*. New York, NY, USA: Association for Computing Machinery, p. 159171.
- Gündoğan, C. et al. Toward a RESTful Information-Centric Web of Things: A Deeper Look at Data Orientation in CoAP, 09 2020, pp. 77–88.
- He, Y. et al. (2023) Lightweight Transmission Behavior Audit Scheme for NDN Industrial Internet Identity Resolution and Transmission Based on Blockchain. *Electronics*, 12(11). doi:10.3390/electronics12112538.
- Huss, N. (2023) *How Many Websites Are There in the World? (2023)*.
- Islam, R.U., Savaglio, C. & Fortino, G. (2025) Leading smart environments towards the future internet through name data networking: A survey. *Future Generation Computer Systems*, 167, 107754. doi:https://doi.org/10.1016/j.future.2025.107754.
- Iyengar, J. & Thomson, M. (2021) *RFC 9000: QUIC: A UDP-Based Multiplexed and Secure Transport*.
- Jackson, D. (2002) Alloy: A lightweight object modelling notation. *ACM Trans. Softw. Eng. Methodol.*, 11(2), 256290. doi:10.1145/505145.505149.
- Jahanian, M., Chen, J. & Ramakrishnan, K. Formal verification of interoperability between future network architectures using alloy. In: *Rigorous State-Based Methods: 7th International Conference, ABZ 2020, Ulm, Germany, May 27–29, 2020, Proceedings 7*. Springer, 2020a, pp. 44–60.
- Jahanian, M., Chen, J. & Ramakrishnan, K.K. Managing the evolution to future internet architectures and seamless interoperation. In: *2020 29th International Conference on Computer Communications and Networks (ICCCN), 2020b*, pp. 1–11.
- Jeon, H.S., Jung, H. & Chun, W. An extended web browser for id/locator separation network. In: *2015 International Conference on Information and Communication Technology Convergence (ICTC), 2015*, pp. 749–754.
- Karapantelakis, A. et al. (2024) Generative AI in mobile networks: a survey. *Annals of Telecommunications*, 79(1), 15–33. doi:10.1007/s12243-023-00980-9.
- Khelifa, S.e., Bagaa, M., Messaoud, A.O. & Ksentini, A. (2024) Case study of webassembly runtimes for ai applications on the edge., 1–6. doi:10.1109/GIIS59465.2024.10449907.
- Kitchenham, B. et al. (2009) Systematic literature reviews in software engineering—a systematic literature review. *Information and Software Technology*, 51, 7–15. doi:10.1016/j.infsof.2008.09.009.
- Kutscher, D. & Oran, D. RESTful Information-Centric Networking: Statement. In: *Proceedings of the 9th ACM Conference on Information-Centric Networking. ICN '22, 2022*. New York, NY, USA: Association for Computing Machinery, p. 150152.
- Liberati, A., Altman, D.G., Tetzlaff, J., Mulrow, C., Gotzsche, P.C., Ioannidis, J.P.A. et al. (2009) The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health-care interventions: explanation and elaboration. *BMJ*, 339(jul21 1), b2700–b2700. doi:10.1136/bmj.b2700.
- Mastorakis, S. et al. (2015) ndnSIM 2.0: A new version of the NDN simulator for NS-3..
- McCauley, J. et al. Enabling a permanent revolution in internet architecture. In: *Proceedings of the ACM Special Interest Group on Data Communication. SIGCOMM '19, 2019*. New York, NY, USA: Association for Computing Machinery, p. 114.
- Microsoft Azure (2024) *Microsoft Azure*. https://azure.microsoft.com/, accessed: October 28, 2024.
- Nah, F.F.H. et al. (2023) Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3), 277–304. doi:10.1080/15228053.2023.2233814.
- Naylor, D. et al. (2014) XIA: Architecting a More Trustworthy and Evolvable Internet. *SIGCOMM Comput. Commun. Rev.*, 44(3), 5057. doi:10.1145/2656877.2656885.
- Ngaffo, A.N., El Ayeb, W. & Choukair, Z. Information-centric networking challenges and opportunities in service discovery: A survey. In: *2020 IEEE Eighth International Conference on Communications and Networking (ComNet), 2020*, pp. 1–8.
- Ofoeda, J., Boateng, R. & Effah, J. (2019) Application programming interface (API) research: A review of the past to inform the future. *International Journal of Enterprise Information Systems (IJEIS)*, 15(3), 76–95.
- OpenAI Platform Docs (2024) *OpenAI Platform Documentation*. https://platform.openai.com/docs/overview, accessed: October 28, 2024.
- Panwar, A. & Bhatnagar, V. Distributed Ledger Technology (DLT): The Beginning of a Technological Revolution for Blockchain. In: *2nd International Conference on Data, Engineering and Applications (IDEA), 2020*, pp. 1–5.
- Qiao, X., Nan, G., Peng, Y., Guo, L., Chen, J., Sun, Y. et al. (2015) NDNBrowser: An extended web browser for named data networking. *Journal of Network and Computer Applications*, 50, 134–147. doi:https://doi.org/10.1016/j.jnca.2014.06.009.
- Raychaudhuri, D., Nagaraja, K. & Venkataramani, A. (2012) Mobility-First: a robust and trustworthy mobility-centric architecture for the future internet. *ACM SIGMOBILE Mobile Computing and Communications Review*, 16, 2–13. doi:10.1145/2412096.2412098.
- Rezk, E. & Fofou, S. A survey of semantic web concepts applied in web services and big data. In: *2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), 2014*, pp. 767–772.
- Saverimoutou, A., Mathieu, B. & Vaton, S. Influence of Internet Protocols and CDN on Web Browsing. In: *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2019*, pp. 1–5.
- Shang, W. et al. NDN.JS: A javascript client library for named data networking. In: *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2013*, pp. 399–404.
- Siddiqui, A.A. & Mueller, P. A requirement-based socket api for a transition to future internet architectures. In: *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2012*, pp. 340–345.
- Silvano, W.F. & Marcelino, R. (2020) Iota Tangle: A cryptocurrency to communicate Internet-of-Things data. *Future Generation Computer Systems*, 112, 307–319.

- doi:<https://doi.org/10.1016/j.future.2020.05.047>.
- Team, N.P. (2014) *NDN Technical Memo: Naming Conventions*. Named Data Networking (NDN). NDN-0022.
URL <https://named-data.net/publications/techreports/ndn-tr-22-3-ndn-memo-naming-conventions/>
- TensorFlow (2024) *TensorFlow*. <https://www.tensorflow.org/>, accessed: October 28, 2024.
- TensorFlow.js (2024) *TensorFlow.js*. <https://www.tensorflow.org/js>, accessed: October 28, 2024.
- Vrijders, S. et al. (2014) Prototyping the recursive internet architecture: the IRATI project approach. *IEEE Network*, 28(2), 20–25. doi:10.1109/MNET.2014.6786609.
- Wang, Y. & Qiao, X. Design and implementation of web browser for named data networking in windows. In: *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*, 2014, pp. 607–612.
- WebAssembly (2024) *WebAssembly*. <https://webassembly.org/>, accessed: October 28, 2024.
- Yang, Z.P., Hua, L., Gao, N.J., Huo, R., Liu, J. & Huang, T. (2021) An Accelerating Approach for Blockchain Information Transmission Based on NDN. *Future Internet*, 13(2). doi:10.3390/fi13020047.
- Yoo, H. & Ko, N. Implementation and evaluation of ndn-based video streaming system. In: *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, 2021, pp. 1879–1881.
- Zali, Z. et al. (2017) Peer-Assisted Information-Centric Network (PICN): A Backward Compatible Solution. *IEEE Access*, 5, 25005–25020. doi:10.1109/ACCESS.2017.2762697.
- Zhang, L. et al. (2014) Named data networking. *SIGCOMM Comput. Commun. Rev.*, 44(3), 6673. doi:10.1145/2656877.2656887.
- Zhang, L. et al. (2014) Named data networking. *ACM SIGCOMM Computer Communication Review*, 44(3), 66–73.
- Zhu, R., Li, T. & Song, T. iGate: NDN Gateway for Tunneling over IP World. In: *2021 International Conference on Computer Communications and Networks (ICCCN)*, 2021, pp. 1–9.