A Survey on Blockchain Interoperability: Past, Present, and Future Trends

RAFAEL BELCHIOR, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal
ANDRÉ VASCONCELOS, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal
SÉRGIO GUERREIRO, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal
MIGUEL CORREIA, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal

Blockchain interoperability is emerging as one of the crucial features of blockchain technology, but the knowledge necessary for achieving it is fragmented. This fact makes it challenging for academics and the industry to achieve interoperability among blockchains seamlessly. Given this new domain's novelty and potential, we conduct a literature review on blockchain interoperability by collecting 284 papers and 120 grey literature documents, constituting a corpus of 404 documents. From those 404 documents, we systematically analyzed and discussed 102 documents, including peer-reviewed papers and grey literature. Our review classifies studies in three categories: Public Connectors, Blockchain of Blockchains, and Hybrid Connectors. Each category is further divided into sub-categories based on defined criteria. We classify 67 existing solutions in one subcategory using the Blockchain Interoperability Framework, providing a holistic overview of blockchain interoperability. Our findings show that blockchain interoperability has a much broader spectrum than cryptocurrencies and cross-chain asset transfers. Finally, this paper discusses supporting technologies, standards, use cases, open challenges, and future research directions, paving the way for research in the area.

CCS Concepts: • **Computer systems organization** → **Dependable and fault-tolerant systems and networks**; **Peer-to-peer architectures**.

Additional Key Words and Phrases: survey, blockchain interoperability, standards, interconnected blockchains, cross-chain transactions, cross-blockchain communication

ACM Reference Format:
Rafael Belchior, André Vasconcelos, Sérgio Guerreiro, and Miguel Correia. 2021. A Survey on Blockchain Interoperability: Past, Present, and Future Trends. 1, 1 (March 2021), 63 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Blockchain technology is maturing at a fast pace. The development of real-world applications shows real interest from both industry and academia [213, 240]. For instance, applications have been developed in the areas of public administration [20, 24], access control [22, 184], and others [55]. Additionally, blockchain is progressing towards the performance of centralized systems: for example, the Hyperledger Fabric blockchain is predicted to achieve 50,000 *transactions per second* [96, 97]. Figure 1 depicts the number of search results per year for "blockchain interoperability"

Authors' addresses: Rafael Belchior, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal, Rua Alves Redol, 9, Lisboa, 1000-029, rafael.belchior@tecnico.ulisboa.pt; André Vasconcelos, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal, Rua Alves Redol, 9, Lisboa, 1000-029, andre.vasconcelos@tecnico.ulisboa.pt; Sérgio Guerreiro, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal, Rua Alves Redol, 9, Lisboa, 1000-029, sergio.guerreiro@tecnico.ulisboa.pt; Miguel Correia, INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal, Rua Alves Redol, 9, Lisboa, 1000-029, miguel.p.correia@tecnico.ulisboa.pt.

Manuscript submitted to ACM

that Google Scholar returned. In 2015, only two documents were related to blockchain interoperability. In 2016, 2017, 2018, 2019, and 2020, the results were 8, 15, 64, 130, and 207, respectively, showing a steep increase regarding interest in this research area.

Serving multiple use cases and stakeholders requires various blockchain features and capabilities [224]. The need for adaptability is a motivating factor for creating different blockchains, leading to a heterogeneous ecosystem [101, 168, 230]. Choosing new blockchains allows researchers and developers to implement new use case scenarios and keep up with recent endeavors. However, each blockchain has its security risks, as the technology is still maturing, the user base is limited (e.g., in comparison to the web or databases), and there are uncovered bugs, and security flaws [14]. Therefore, developers and researchers have to choose between novelty and stability, leading to a vast diversity of choices [8]. This diversity leads to *fragmentation*: there are many *immature* blockchain solutions (e.g., without extensive testing). Until recently, blockchains did not consider the need for interoperability, as each one focused on resolving specific challenges, leading to *data and value silos* [1, 117, 207].

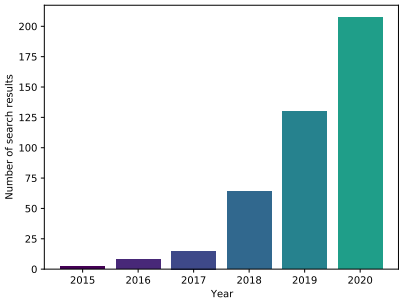

Fig. 1. Research trends on blockchain interoperability

Moreover, what if the blockchain in which a particular service is running becomes obsolete, vulnerable, or is shutdown? If the user requirements or circumstances change over time, a different blockchain might be more appropriate for a specific use case [148]. What if the service to serve is so crucial that it requires seamless dependability? Furthermore, if we want to reproduce our use case to another blockchain, how can we increase *portability*?

In 1996, Wegner stated that "interoperability is the ability of two or more software components to cooperate despite differences in language, interface, and execution platform" [225]. In that context, Wegner established a bridge between the concept of interoperability and existing standards. As authors were influenced by the standards existing at that time, authors nowadays are influenced by the Internet architecture and concepts, in what concerns blockchain interoperability [103, 206]. Thus, reflecting on the Internet's architecture seems like a good starting point to understand how blockchains can interoperate. Thus, it is important to solve the *blockchain interoperability* challenge, i.e., to provide interoperability between blockchains in order to explore synergies between different solutions, scale the existing ones, and create new use cases (see Section 2.3). For example, a user should be able to transfer their assets from a blockchain to another or build *cross-blockchain decentralized applications*.

While information systems evolve, so do the meaning and scope of interoperability. According to the National Interoperability Framework Observatory (NIFO), endorsed by the European Commission, there are several interoperability layers [159]: *technical interoperability*, *semantic interoperability*, *organizational interoperability*, *legal interoperability*, *integrated public service governance*, and *interoperability governance*. For instance, technical interoperability regards the technical mechanisms that enable integration among blockchains, while semantic interoperability concerns whether the application-specific semantics can be conserved across blockchains. Despite interoperability having an extensive scope, we mainly focus on *technical interoperability*, and *semantic interoperability* as most blockchain interoperability work is concentrated. We leave the study of other interoperability layers for future work.

Interoperability does not only conflate flexibility and application portability. It also has the potential to solve some of the biggest blockchain research challenges. In particular, interoperability promotes blockchain *scalability*, as it provides a way to offload transactions to other blockchains, e.g., via sharding [92, 219], it can promote privacy (by allowing the

end-user to use different blockchain for data objects with different privacy requirements), and creates new business opportunities. Given the complexity of this research area, we attempt to answer three research questions:

RQ1: What is the current landscape concerning blockchain interoperability, both in industry and academia?

RQ2: Are technological requirements for blockchain interoperability currently satisfied?

RQ3: Are there real use cases requiring blockchain interoperability?

1.1 Contributions

As a systematization of knowledge on blockchain interoperability, this paper yields four-fold contributions:

- Introduce the blockchain interoperability research area, presenting the necessary background and highlighting definitions tailored both for industry and academia. We define blockchain interoperability and discuss different blockchain interoperability architectures and standards.
- Propose the Blockchain Interoperability Framework (BIF), a framework defining criteria to assess blockchain interoperability solutions.
- Present a *systematic literature review*, where we identify and discuss blockchain interoperability solutions, accordingly to BIF, in three categories: *Public Connectors*, *Blockchain of Blockchains*, and *Hybrid Connectors*. In particular, our analysis is based on several sources (e.g., peer-reviewed papers, whitepapers, blog posts, technical reports), enabling an in-depth understanding of each solution's current state and its *roadmap*, i.e., its creator's plans. To achieve this, *we systematically contacted the authors of grey literature papers and industrial solutions*: this is our innovative attempt to provide the reader with high-quality information in this rapidly emerging research area. This method allows us to obtain up-to-date, reliable information that often is cumbersome to obtain.
- We identify and propose use cases that benefit from a multiple blockchain approach, pinpoint challenges and obstacles to the development of blockchain interoperability solutions and standards, and propose future research directions, paving the way for systematic research in this area.

1.2 Organization

Section 2 provides background on blockchain consensus algorithms, previous results on blockchain interoperability, and blockchain interoperability definitions and architecture. Next, Section 3 presents and discusses related literature reviews, while Section 4 introduces the Blockchain Interoperability Framework. Next, a systematic review and analysis of blockchain interoperability categories is conducted, distributed across three categories, in Section 5: Public Connectors (Section 5.1), Blockchain of Blockchains (Section 5.2), and Hybrid Connectors (Section 5.3). For each category, we provide a detailed analysis and discussion. To provide a holistic view of the blockchain interoperability landscape, we promote a general discussion in Section 6. This discussion compares solutions across categories (Section 6.1), presents standardization efforts (Section 6.2), informs readers regarding use case scenarios with multiple blockchains (Section 6.3), answers to the research questions (Section 6.4), and indicates challenges related to interoperability (Section 6.5). We present research directions (Section 7), and, finally, we conclude the paper (Section 8). Six appendices complement this survey. Appendix A presents the methodology employed. Appendix B presents an architecture for blockchain interoperability, reviewing the various efforts on that topic. Appendix C, D and E presents a description of the surveyed public connectors, blockchain of blockchains, and hybrid connector approaches, respectively. Finally, Appendix F complements the use case section, by presenting more cross-blockchain use cases.

2 BACKGROUND

In this section, we provide the necessary background to the understanding of this survey.

2.1 A Primer on Blockchain Technology

The term *blockchain* has at least two different meanings: a type of system and a type of data structure. In this paper, we use the term blockchain to denominate a class of distributed systems. A blockchain maintains a shared state, specifically a replicated data structure that we denominate *distributed ledger*. This ledger is maintained by a set of machines with computational and storage resources, called nodes (or peers or participants). Nodes are not trusted individually to maintain the distributed ledger; they are trusted as a group due to their number and diversity [54]. A blockchain can also be considered a *deterministic state machine* that provides a certain service, given existing incentives that the network can reward. The first blockchain was part of the Bitcoin system and provided as service transactions of a cryptocurrency, a digital currency, also designated Bitcoin [158]. The service provided by Bitcoin is the execution of transactions of bitcoins.

Most blockchains are programmable, i.e., their state machine is extensible with user programs. These programs are often designated *smart contracts* [51, 204] and their execution is caused by calls also designated *transactions*. Smart contracts are executed in a virtual machine, e.g., in the Ethereum Virtual Machine (EVM) in Ethereum and other blockchains that adopted the EVM for compatibility (that we designate *EVM-based blockchains*). Smart contracts are often used to implement *tokens*, i.e., blockchain-based abstractions that can be owned and represent currency, resources, assets, access, equity, identity, collectibles, etc. [10]. There are several standard token formats, e.g., ERC-20 and ERC-721. These tokens are fungible and non-fungible assets, respectively. A fungible asset is interchangeable with another asset of the same type. Conversely, a non-fungible asset is an asset that is unique and has specific properties.

In many blockchains, transactions are aggregated in *blocks*, linked by the previous block's cryptographic hash. Hence those data structures are also called blockchains - often viewed as deterministic state machines.

Blockchain systems ought to be resilient to faults (e.g., *crash fault-tolerant* or *Byzantine fault-tolerant*), as there may be crashes or malicious nodes on the network [63]. They run a consensus algorithm to create agreement on a global ledger state in the presence of Byzantine faults. Consensus algorithms are important because they define the behavior of blockchain nodes and their interaction [63, 238], and the security assumptions of each blockchain. They, therefore, affect how blockchain peers communicate and operate with each other: in Bitcoin's Proof-of-Work (PoW), peers have to compute a cryptographic challenge to validate transactions, competing with each other. Another blockchain, Tendermint, uses a Byzantine fault-tolerant state machine replication (BFT) algorithm [130], supporting up to a third less one of faulty participants. In Hyperledger Fabric, a widely-used private blockchain platform, a consensus algorithm allows higher transaction throughput than PoW by allowing a subset of nodes to execute and endorse transactions (called endorser peers) and by typically using a weaker consensus (only crash fault-tolerant). The variety of blockchain infrastructures makes it challenging to categorize blockchains, and their interoperability solutions, as there are no *de facto* blockchain interoperability or blockchain architecture standards.

Apart from differences in the consensus, blockchains can be deemed *public* (also called permissionless) or *private* (also called permissioned). Permissionless blockchains do not require authentication for participants to access the ledger. *Bitcoin* [158] and *Ethereum* [51, 228] are examples of such blockchains. Permissioned blockchains are blockchains in which users are authenticated and can be held accountable according to a governance model suitable for enterprise and

governmental needs. Hyperledger Fabric [9], Corda [47], Quorum [119], Tendermint [130], and Multichain [98] are examples of permissioned blockchains.

Figure 2 depicts two blockchains: Hyperledger Fabric, a permissioned blockchain; and Bitcoin, a permissionless blockchain. The supporting layers (e.g., networking, storage, encryption) [120] provide a basis for the consensus engine, which orders transactions and appends them to the chain of blocks. In Hyperledger Fabric, the consensus is modular, based on endorsement policies. In Fabric, a client (C) sends a transaction proposal to the peer nodes (P), and obtains a signed transaction, called an endorsement (steps 1 and 2). An orderer validates the endorsements and builds a block with valid transactions, appending it to the ledger (steps 3 and 4). In Bitcoin, the consensus is based on the notion of Proof-of-Work (PoW), a cryptographic puzzle that mining nodes need to solve in order to build a valid block. This corresponds roughly to Fabric's steps 1-3. After a node finds a solution to PoW, it then can propose a block of transactions to be appended to the ledger (step 4).

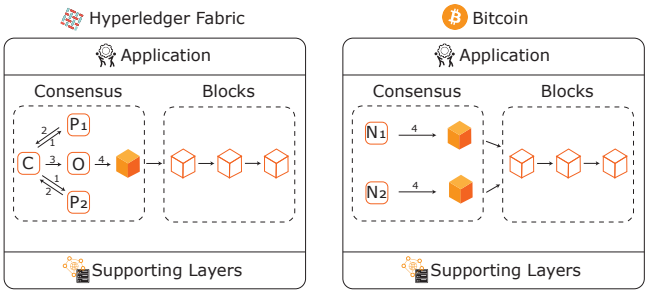

Fig. 2. Representation of two blockchains, Hyperledger Fabric [9], and Bitcoin [158].

Blockchain trust is based on the incentive models that guide the behavior of the nodes. For instance, in Bitcoin, nodes have the incentive to produce blocks of transactions and support the network because they are rewarded Bitcoins. Conversely, nodes do not have the incentive to disrespect the protocol, as attacks are expensive and nodes can get punished [61]. In Hyperledger Fabric, where nodes are identified, they have the business incentive to follow the protocol because parties cooperate towards a common goal, and misbehavior can be punished according to the law or applicable governance model. Decentralization, different goals, and incentives support the trust on the blockchain – parties can share the ledger without relying on a trusted, centralized party.

The ability to distribute trust on a global state fostered the appearance of *decentralized applications* (*dApps*) [10]. A dApp is a computer program running on a decentralized peer-to-peer network. For example, Steemit[1] is a social blogging dApp that rewards content-creators with cryptocurrency. Thus, dApps are based on smart contracts running on a blockchain, but they also have other components that should equally be decentralized.

2.2 Cross-Blockchain Communication

Cross-blockchain communication involves two blockchains: a *source blockchain*, and a *target blockchain*. The source blockchain is the blockchain in which the transaction is initiated to be executed on a target blockchain. While general-purpose interoperability comes down to a blockchain exposing its internal state to other, cross-chain asset transfers rely on an atomic three-phase procedure: 1) locking (or extinguishing) of an asset on a source blockchain; 2) blockchain transfer commitment, and 3) creation of a representation of the asset on a target blockchain [25, 92, 105]. This procedure, later explained in detail, relies on a *cross-chain communication protocol* (CCCP).

[1] https://steemit.com/

A CCCP defines the process by which a pair of blockchains interact to synchronize cross-chain transactions correctly. Hence, a CCCP allows *homogeneous* blockchains to communicate. For instance, sidechains typically use a CCCP (e.g., Zendoo allows communication between Bitcoin-like blockchains systems [93]). Conversely, a *cross-blockchain communication protocol* (CBCP) defines the process by which a pair of blockchains interact to synchronize cross-blockchain transactions correctly. CBCPs allow *heterogeneous* blockchains to communicate (e.g., the Interledger Protocol allows any blockchains that implement the protocol to exchange "money packets" [115]). The differentiation between CCCPs and CBCPs is important because CCCPs typically can leverage the interoperating blockchains' constructs and functionality (e.g., utilize smart contracts to implement a relay [131]), whereas CBCPs normally require blockchains to be adapted. However, CBCPs may leverage specific functionalities of both blockchains [79]. Cross-blockchain, or cross-chain communication, is a requirement for blockchain interoperability. This section provides a few theoretical results regarding cross-blockchain communication, and thus also blockchain interoperability.

Zamyatin et al. [233] prove that "there exists no asynchronous CCC [cross-chain communication] protocol tolerant against misbehaving nodes". The authors use a reduction to the fair exchange problem [13] to prove that correct cross-chain communication is as hard as the fair exchange problem. As a consequence of the presented theorem, the authors state that "there exists no CCC protocol tolerant against misbehaving nodes without a trusted third party". A trusted third party can be centralized or decentralized. Centralized trusted parties are, for example, trusted validators [155]. A decentralized trusted party can be another blockchain, in which their participants agree on the global ledger state via a consensus algorithm. However, the trusted party has to ensure that most participants are honest, guaranteeing the correctness of the process is guaranteed. Cross-chain protocols, therefore "use the consensus of the distributed ledgers as an abstraction for a trusted third party." [233]. Borkowski et al. [43] derive the "lemma of rooted blockchains" that states that a source blockchain cannot verify the existence of data on a target blockchain with practical effort. In particular, the source blockchain would need to be able to mimic consensus from the target blockchain, and it would have to store a (potentially large) subset of the target blockchain's block history. On a recent endeavor, Lafourcade and Lombard-Platet [133] formalize the blockchain interoperability problem, arguing that fully decentralized blockchain interoperability is not possible. More specifically, there is no protocol assuming a full-client that can realize its interoperability functions, such as asset transfer, without a third party's aid. However, a blockchain with two ledgers offers the possibility of interoperability (there is, in fact, the possibility of moving assets from one ledger to the other). This study applies mainly to public blockchains.

The results above are relevant because they lead to an important consideration: *cross-blockchain transactions are not feasible in practice without the participation of a trusted third party*. In other words, although trust assumptions vary greatly from permissionless to permissioned networks, cross-blockchain transactions, as well as cross-chain transactions, require a trusted third party to assure the correctness of the underlying protocol. Most solutions presented throughout this paper present at least one decentralized trust anchor.

2.3 Blockchain Interoperability Definitions

In this section, we define additional technical terms for an understanding of this study.

Vernadat defines interoperability among enterprise systems as [216]: "a measure of the ability to perform interoperation between [...] entities (software, processes, systems, business units...). The challenge relies on facilitating communication, cooperation, and coordination among these processes and units". Abebe et al. propose a general communication protocol as an alternative approach to the "point-to-point" blockchain interoperability approach [1]. Interoperability is defined as "the semantic dependence between distinct ledgers to transfer or exchange data or value,

with assurances of validity". Pillai and Biswas refer that "cross-communication is not intended to make direct state changes to another blockchain system. Instead, cross-communication should trigger some set of functionalities on the other system that is expected to operate within its own network" [167].

A technical report from the National Institute of Standards and Technology (NIST) defines blockchain interoperability as "a composition of distinguishable blockchain systems, each representing a unique distributed data ledger, where atomic transaction execution may span multiple heterogeneous blockchain systems, and where data recorded in one blockchain are reachable, verifiable, and *referable* by another possibly foreign transaction in a semantically compatible manner" [231]. Hardjono et al. define blockchain survivability as "the completion (confirmation) of an application-level transaction [composed of subtransactions] independent of blockchain systems involved in achieving the completion of the transaction."[103] The concept of transactions and subtransactions relates to "*best effort delivery*", that applications must comply to, by ensuring that transactions and their *subtransactions* are completed (i.e., committed) within a certain time frame.

Regarding types of blockchain interoperability, Besançon et al. highlight three [29]: interoperability between different blockchains, interoperability between dApps using the same blockchain, and interoperability blockchain and other technologies (such as integration with enterprise systems). While different definitions tackle different dimensions of interoperability, there is room for improvement. We define several terms that encompass the whole scope of technical interoperability to later provide a holistic definition of technical interoperability (see Figure 3). To recall the definition presented in Section 2.2, a source blockchain is a blockchain that issues transactions against a target blockchain. A *source node* is a node from the source blockchain, and a target node belongs to the target blockchain. When several participants elect a source node and a target node, we achieve decentralization in the context of interoperability [117].

A *Cross-Chain Transaction* (CC-Tx), where "CC" stands for *cross-chain*, and "Tx" for transaction, is a transaction between different chains, which belong to the same blockchain system (homogeneous blockchains), for example, between EVM-based blockchains. We use the CC-Tx, *inter-chain transaction*, and *inter-blockchain transaction* terms interchangeably. A *Cross-Blockchain Transaction* (CB-Tx) is a transaction between different blockchains (heterogeneous blockchains), for example, between Hyperledger Fabric and Bitcoin. Note that the terms CC-Tx and CB-Tx are used as synonyms in the industry, as currently, most solutions connect homogeneous blockchains. A *Cross-Chain Decentralized Application* (CC-dApp) is a dApp that leverages cross-blockchain transactions to implement its business logic. We use the terms CC-dApp and *cross-blockchain decentralized application* (CB-dApp) interchangeably. Other terms with the same meaning in the literature are inter-chain decentralized application and inter-blockchain decentralized application.

A *Internet of Blockchains* (IoB) is a system "where homogeneous and heterogeneous decentralized networks communicate to facilitate cross-chain transactions of value" [206]. We use this definition of IoB throughout this paper.

The term *Blockchain of Blockchains* (BoB) is not used consistently [143, 215]. Verdian et al. use it to describe the structure that aggregates blocks from different blockchains into "meta blocks", organized through a consensus mechanism using *posets* (partially ordered sets) and total order theory [215], thus producing a blockchain of blockchains. A poset consists of a set of elements and their binary relationships that are ordered according to a specific set of rules [32].

Influenced by those authors, we define a BoB *as a system in which a consensus protocol organizes blocks that contain a set of transactions belonging to CC-dApps, spread across multiple blockchains. Such a system should provide accountability for the parties issuing transactions on the various blockchains and providing a holistic, updated view of each underlying blockchain.* Note that BoB solutions belong to the category with the same name. Therefore, the notion of IoB directly

refers to the connection relationships among blockchains, whereas the term BoB refers to an architecture made possible by IoB. BoB approaches are concerned with the validation and management of cross-blockchain transactions.

Figure 3 shows the relationship between the different concepts concerning blockchain interoperability. A CC-dApp realizes the blockchain of blockchains approach. This approach can provide the semantic level interoperability (i.e., concerned at transmitting the meaning of the data, which corresponds to the value level interoperability) required by organizations, mappable by the applicational layer. However, it relies on the existence of an IoB – a network of blockchains. For an IoB to exist, technical interoperability (or mechanical interoperability) is required. In the context of a CC-dApp, cross-chain transactions are ordered by a *cross-chain dApp protocol*. Such protocols should assure transaction atomicity and resolve possible conflicts in transactions spawning across homogeneous and heterogeneous blockchains.

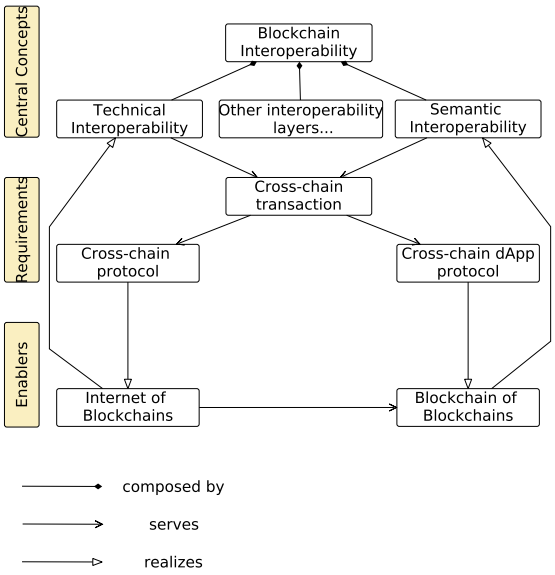

Fig. 3. Concept map, illustrating the relationship between different concepts related to blockchain interoperability

From the several definitions we encountered during our research, we envision *blockchain interoperability* as: *the ability of a source blockchain to change the state of a target blockchain (or vice-versa), enabled by cross-chain or cross-blockchain transactions, spanning across a composition of homogeneous and heterogeneous blockchain systems, the IoB.* IoB transactions are delivered via a cross-blockchain communication protocol, thereby granting technical interoperability, enabling CC-dApps. CC-dApps provide semantic interoperability via the BoB. The BoB approach is realized by a cross-blockchain dApp protocol, which provides consensus over a set of cross-chain transactions, thus enabling cross-chain dApps.

3 RELATED LITERATURE REVIEWS

Due to the novelty and large-breadth of this research area, few updated surveys cover aspects of blockchain interoperability. We compare existing surveys based on the *criteria* and *sub-criteria* shown in Table 1. For example, in the first row, the criteria "public connector" evaluates if a study addresses its sub-criteria: work on sidechains, hash-lock time contracts, and notary schemes. On the second row, the criteria Blockchain of Blockchains evaluates if a study describes BoB solutions (1) and if it performs a detailed comparison, including consensus, security, validators, and performance.

Buterin presents a survey on public connector solutions, including notary schemes, sidechains, and hash-time locking techniques [52]. Similarly, other surveys focus on public connectors [42, 127, 199, 233], with a focus on sidechains and hash lock time contracts. Vo et al. present work mostly on architecture for interoperability, presenting some BoB and HC solutions [206], while Qasse et al. organize solutions across sidechains, blockchain routers, smart contracts, and industrial solutions [175]. Johnson et al. focus on Ethereum as the infrastructure enabling interoperability across several categories of solutions [118]. Siris et al. [200], Kannengieber et al. [121], and Bishnoi et al. [33] tackle a wider range of solutions.

Criteria	Description	Sub-criteria 1	Sub-criteria 2	Sub-criteria 3
Public Connectors (PC)	Addresses public connectors	Sidechains	Hash lock contracts	Notary Schemes
Blockchain of Blockchains (BoB)	Addresses BoBs	Describes solutions	Detailed comparison	N/A
Hybrid Connectors (HC)	Addresses Hybrid Connectors	Trusted Relays	Blockchain agnostic protocols	Blockchain migrators
Architecture (AR)	Addresses architectures enabling CCCPs	Proposes architecture	Presents related work	N/A
Cross-chain Standards (ST)	Addresses standards for interoperability	Present standards	Relate standards to solutions	N/A
Cross-analysis (CC)	Compares across categories	Compare categories	Compare sub-categories	N/A
Use Cases (UC)	Presents use cases using an IoB or BoB	Existing use cases	Predicted use cases	N/A
Open Issues (OI)	Challenges on interoperability	Research directions	Relate interoperability to other issues	N/A

Table 1. Survey comparison criteria, description, and sub-criteria.

We aim at providing a solid, throughout and comprehensive foundation on which researchers can rely upon as a starting point in this field, including a description of the related surveys, which illuminated our research. In contrast to most of the works mentioned above, this paper provides a holistic view of blockchain interoperability by focusing not only on public connectors but also on BoBs and hybrid connectors. By including updated grey literature and focusing on private blockchain interoperability, a comprehensive discussion on standards, use cases, and architecture for interoperability was possible.

	Solution category			Detailed Analysis				
Reference	PC	BoB	HC	AR	ST	CC	UC	OI
Buterin [52], 2016	+	-	-	-	-	±	+	+
Vo et al.[206], 2018	-	±	±	+	±	±	±	+
Borkowski et al. [41], 2018	+	-	-	-	-	±	-	+
Quasse et al. [175], 2019	±	±	±	-	-	±	±	±
Johnson et al. [118], 2019	±	±	±	-	-	-	-	-
Zamyatin et al. [233], 2019	+	-	-	-	-	±	-	+
Siris et al. [200], 2019	±	±	±	±	-	+	-	-
Koens et al. [127], 2019	+	+	-	-	-	±	-	+
Singh et al. [199], 2020	+	-	-	-	-	-	+	+
Kannengießer et al., [121], 2020	+	±	±	-	-	±	-	-
Bishnoi et al. [33], 2020	+	±	±	-	-	-	-	-
this survey	+	+	+	+	+	+	+	+

Table 2. Comparison of related literature reviews: PC (Public Connectors), Blockchain of Blockchains (BoB), HC (Hybrid Connectors), AR (architectures for blockchain interoperability), ST (standards), CC (cross-comparison), UC (use cases), OI (open-issues). Each criterion can be "fulfilled" ("+" in green background), "partially fulfilled" ("±" in orange background) or "not fulfilled" ("-" in red background), if it addresses all, between one and all, or none of its sub-criteria, respectively.

4 BLOCKCHAIN INTEROPERABILITY FRAMEWORK

This section presents the Blockchain Interoperability Framework (BIF), a framework classifying solutions collected through our methodology. To drive criteria for assessing the categories (and specific solutions) of blockchain interoperability, we analyzed the solution space using the six "W" questions: Who, What, Where, When, Why, and How. The "Why" was determined irrelevant to our analysis because its purpose is constant – connecting different chains (CC-Txs), different blockchains (CB-Txs), or even to arbitrary systems (e.g., enterprise legacy systems). This is instead addressed by the "where" question.

4.1 Deriving Evaluation Criteria

The "what" refers to the *assets* exchanged. An interoperability solution can handle different data objects or assets. Hence it is important to know which data representations a solution supports [225]. Assets can be treated as data (arbitrary payloads), as fungible assets, or non-fungible assets [18, 155, 168]. Arbitrary data is often represented via a key-value pair, being the preferred representation of some blockchains [9, 54, 111]. The key-value is also useful to represent the contents of account-based blockchains [56, 77, 119]. Payment tokens are fungible tokens [167]. Utility tokens include tokens used to access a service or application, such as non-fungible tokens (e.g., ERC20 tokens). Finally, asset tokens represent real-world physical or digital instruments, such as blockchain-based promissory notes, regulated by the Swiss Financial Market Supervisory Authority [188] (see more details in Section 6.3), or bonds [18]. An asset has different maturity levels. In particular, an asset may be standardized, (e.g., ERC tokens[217], standardized schema for utility tokens, ERC1400, a security token [192, 193]) and/or regulated [150, 203, 214]. Regulated digital assets are backed by legal frameworks. We consider all asset tokens to be regulated. We envision utility tokens as standardized and asset tokens as standardized and regulated (i.e., asset tokens are emitted by legal entities).

The "who" question refers to whom controls the CC-Tx process and thus accounts for trust establishment [94, 233]). It can be the end-user (e.g., [86, 155]), a consortium (e.g., [15, 191]), or a trusted third party (e.g., cloud services, centralized notary schemes). Some solutions allow different levels of control.

The "where" refers to what are the source and target ledgers, as well as what is the support of conducting the CC process. Solutions can support public blockchains (P) or non-public blockchains (NP). We use NP to designate private blockchains, other decentralized ledger technology (DLT) systems, and centralized systems (e.g., VISA payment network). The supported systems of each solution matter since communication may happen unidirectionally or bi-directionally [155]. Blockchain oracles apart, it often is not feasible to have a solution based on a blockchain system connected to a centralized system (e.g., providing insurance data). A smart contract may be the one conducting an asset transfer (on-chain channel, with on-chain CC-Tx validation) versus an off-chain settlement, e.g., techniques using commitment schemes [2, 93], or via (semi-)centralized system (off-chain channel). Typically, on-chain channels offer more resiliency, but off-chain channels are more scalable. Combinations between off-chain and on-chain channels also exist (e.g., payment networks [174]). Offline channels depend on different proof generation mechanisms [2, 93, 94].

The "when" refers to the set of processes (e.g., executing CC-Txs) that are defined at *design-time* or *run-time*. *Design-time customization* decisions affect the punctual behavior of a CC-dApp concerning when it is executed. At design-time, a user defines the behavior of the solution *apriori*. If a change is needed, a new instance of the solution needs to be deployed. Conversely, *run-time customization* decisions are flexible, allowing the end-user to adjust the conditions defined by business logic as needed. Solutions in which business logic is changed at run-time are called *flexible approaches*, allowing to adjust business logic and conditions that trigger the execution of a CB-Tx or CC-Tx by a CC-dApp. Most literature reviews focus on design-time approaches and public blockchains, leaving a vast range of recent solutions out of scope. In this survey, we also consider private-private and public-private blockchain interoperability, focusing on flexible approaches.

The "how" regards the realization of cross-chain transactions: how are CC-Txs realized on the underlying DLTs? Often, these transactions can be performed using *cross claims*, i.e., by locking/burning an asset on the source blockchain and unlocking/creating its representation on the target blockchain. Cross-claims require two nodes from different blockchains, where one performs one operation in a source blockchain in exchange for its counterparty performing other operations on a target blockchain - each party logs the operation in case a dispute is needed. Typically, cross-claims

operate in semi-trusted environments (e.g., private blockchain, regulated blockchain), and can be operated via a (semi) trusted third party [19, 105, 155]. Escrowed cross-claims are the standard mechanism for asset transfers, operating similarly to cross-claims, but in an untrusted environment, leveraging dispute-resolution mechanisms (e.g., via smart contracts requiring inclusion proofs [2]) or by parties holding custody of assets and collateral, [45, 53, 234]. Inclusion proofs include applying Merkle tree proofs to block header transfer via a coordinating blockchain, block header transfer, or direct signing [182]. Collateralization is the process in which a party performing the transfer of assets provides a certain amount of their assets as a guarantee of following the protocol (e.g., not to steal assets from the end-user). If a party misbehaves (e.g., steals assets), the deposit is given to the victim party. Finally, a mediated CC-Tx includes (an offline) trusted party [155]. In case of a dispute about an asset transfer between a public blockchain and a private blockchain (P-NP) or a public blockchain and an enterprise system (also P-NP), there needs to be a dispute-resolution mechanism. This is due to NP systems' private nature, although several mechanisms exist to prove internal state belonging to private blockchains. Hence, CC-Txs have a trade-off risk-performance: the less centralization there is on the CC-Tx settlement, the worst the performance, but the lesser the risk.

The "how" also relates to the extent to which the implementation of the solution is tested. Solutions might be implemented, tested, and validated (application to a real-world scenario). Testing regards *correctness guarantees*: *behavioral correctness* or *formal correctness*. Behavioral correctness is the ability to guarantee that CC-Txs are issued as intended, without unintended consequences (e.g., asset lock, asset theft). While in practice, behavioral correctness depends on formal correctness, we say a solution has behavioral correctness if it has a suite of test cases [157]. Formal correctness assures that an algorithm is correct with respect to a specification. Formal verification checks the correctness of algorithms against a specification using, for instance, formal methods. Smart contract verification tools allow developers to reduce the probability of creating bugs, thus incurring penalties, as smart contracts are generally difficult to update once deployed [75]. Another point of providing trust to the user is the solution to have an open-source implementation, where the code can be peer-reviewed and corrected if needed.

4.2 Evaluation Criteria

Having discussed the survey's scope, we next define the set of criteria we use to characterize the interoperability solutions. Similarly to Section 3, each criterion can be "fulfilled" "partially fulfilled" or "not fulfilled". If a criterion is a yes/no question (e.g., does the solution support asset type "data"?), we do not explicitly refer to the fulfillment conditions as they are evident. Next, we detail the criteria type (first-level), criteria sub-type (second level), and criteria from BIF:

- Asset: this category refers to properties of an asset involved in a CC-Tx.
  - Type: what type of assets does the solution support?
    (1) Data: can the solution manipulate arbitrary data?
    (2) Payment tokens: can the solution manipulate cryptocurrencies? This criterion is partially fulfilled if the asset is only used as collateral or to reward a service's operational maintenance.
    (3) Utility tokens: can the solution manipulate utility tokens? This criterion is partially fulfilled if the asset is used only as collateral or to reward a service's operational maintenance.
    (4) Asset tokens: can the solution manipulate utility tokens?
  - Infrastructure: what are the systems involved?
    (1) P: This criterion is fully fulfilled if more than two public blockchains are supported. It is partially fulfilled if one or two public blockchains are supported.

(2) NP: This criterion is fully fulfilled if more than two non-public blockchains are supported. It is partially fulfilled if one or two non-public blockchains are supported.

- Trust Establishment: this category refers to how a solution provides trust to the users.
  - Decentralization: who operates the solution instance?
    (1) End-user
    (2) Consortium
    (3) Trusted (third) party

    If multiple criteria are selected, it indicates a solution supports more than one mode of operation.
  - Channel: where are CC-Tx validated?
    (1) On-chain: This criteria is partially fulfilled if proofs are created on-chain but validation occurs off-chain.
    (2) Off-chain: This criteria is partially fulfilled if proofs are created off-chain but validation occurs on-chain.
- CC-Tx Realization: this category refers to how and where a CC-Tx is settled.
  - Mechanism: how are CC-Txs agreed-upon multiple parties?
    (1) Cross-claim
    (2) Escrowed cross-claim
    (3) Mediated
- Extra-functional: this category refers to the design of the solution itself.

(1) Tests: the approach provides a set of test cases.
(2) Implementation: the approach provides an open-source implementation and is validated in the industry. This criterion is partially fulfilled if the implementation is closed-source.
(3) Validation: the approach is validated in an actual use case scenario.
(4) Run-time: the business logic of the solution can be changed dynamically, as needed. This criterion is considered not fulfilled if logic is settled when the solution is instantiated, i.e., changing logic requires a new instance.

5 OVERVIEW OF BLOCKCHAIN INTEROPERABILITY APPROACHES

We conducted a systematic literature review following the protocol described in Appendix A, yielding 80 relevant documents out of the initial 330. By grouping the publications and grey literature, a pattern arises: these works are either about interoperability across public blockchains holding cryptocurrencies, application-specific blockchain generators with interoperability capabilities, or protocols connecting heterogeneous blockchains. We thus classify each study into one of the following categories: *Public Connectors* (Section 5.1), *Blockchain of Blockchains* (Section 5.2), and *Hybrid Connectors* (Section 5.3). Each category is further divided into sub-categories. Table 3 summarizes the work conducted.

5.1 Public Connectors

The first family of blockchain interoperability solutions aimed to provide interoperability between cryptocurrency systems, as stated by Vitalik [52]. This category identifies and defines different chain interoperability strategies across public blockchains supporting cryptocurrencies, including sidechain approaches, notary schemes, and hash time hash-locks. Some solutions share characteristics of more than one sub-category, and thus they can be considered hybrid. We introduce each sub-category, presenting only two illustrative examples of each one for the sake of space. Appendix C depicts a complete list of Public Connectors approaches. After that, a summarized evaluation table is presented using the BIF. These tables are later discussed in Section 5.1.4.

5.1.1 Sidechains & Relays. A *sidechain* (or *secondary chain*, or *satellite chain*, or *child chain*) is a mechanism for two existing blockchains to interoperate [15, 94], scale (e.g., via blockchain sharding [128]), and be upgraded [235] in which

	Asset			Trust Establishment										
	Type			Infra.		Decentral.			Channel		CC-Realization			
Sub-Category	D	P	U	P	NP	U	C	TTP	OC	OF	CC	ECC	M	**References**
Sidechains & Relays	+	±	-	±	-	-	+	-	+	+	-	+	-	[173, 220]
	+	±	-	±	-	+	+	-	+	-	-	+	-	[17, 79, 89, 90, 131]
	-	+	+	+	-	+	+	-	+	-	-	+	-	[11, 116]
	-	+	+	±	-	+	+	-	+	+	-	+	-	[15, 74, 124, 144, 169]
	+	+	-	±	-	-	+	-	+	-	-	+	-	[65, 93, 134, 135]
	-	+	+	±	-	-	+	-	+	±	-	+	-	[26, 70, 100, 181, 194, 195]
	-	+	-	+	-	+	+	+	-	+	+	-	+	[112, 115]
Notary Scheme	-	+	+	+	-	-	-	+	±	-	-	-	+	See Section 5.1.2
	-	+	+	+	-	+	+	-	+	-	-	+	-	[149, 210, 223]
HLTC	-	+	+	±	-	-	+	-	+	-	-	+	-	[45, 60, 66, 91, 145, 186, 234]
Blockchain of Blockchains	+	+	+	±	-	+	+	-	+	-	-	+	-	[129, 130, 227]
	+	+	+	+	+	-	+	+	+	-	-	+	+	[12, 177, 201]
Trusted Relays	+	-	-	±	±	+	-	-	-	+	-	-	+	[48, 83, 120, 161]
	+	+	+	±	+	+	+	-	+	±	+	-	+	[19, 25, 103, 105, 221, 229, 237]
B. Agnostic Protocols	+	+	+	+	+	+	+	-	-	+	+	+	-	[1, 2, 155, 176]
	+	+	+	±	±	+	+	-	+	-	-	+	-	[58, 73, 143, 164, 168, 182]
Blockchain Migrators	+	-	-	±	-	+	-	-	-	+	N/A	N/A	N/A	[86, 189, 226]
	+	+	+	±	±	+	+	-	+	-	-	+	-	[92]

Table 3. Evaluation of blockchain interoperability solutions by subcategory accordingly to the Blockchain Interoperability Framework. N/A stands for not applicable. Public connectors are represented in green, Blockchain of blockchains in orange, and Hybrid connectors in red.

one blockchain (*main chain* or mainchain) considers another blockchain as an extension of itself (the sidechain). The mainchain maintains a ledger of assets and is connected to the sidechain, a separate system attached to the main chain via a cross-chain communication protocol [93]. An example is a *two-way peg*, a mechanism for transferring assets between the main chain and the sidechain [199]. Main chains can be sidechains of each other [52], creating each chain's possible to connect to others. Sidechains are considered layer one solutions (built on top of layer 0 solutions - blockchains) to implement layer-2 solutions, such as payment channels [124]. The second layer allows off-chain transactions between users through the exchange of messages tethered to a sidechain [99]. A sidechain is then a construct that allows for offloading transactions from the mainchain, processes it, and can redirect the outcome of such processing back to the main chain.

For instance, state channels are off-chain sidechains used to implement, for example, payment channels, by offloading transactions of the blockchain [174]. In a payment channel, participants interact, collecting cryptographically signed messages. Those messages update the current state without publishing it to the mainchain. When the payment channel is closed, the final state is published onto the main chain, where an on-chain dispute/closure phase may occur [124]. Payment channels are appropriated for use cases requiring several transactions that can be combined in a single one.

Main chains communicate with sidechains via a CCP, often tightly coupled with the functionality of both chains. The basic components of sidechain design are the mainchain consensus protocol, the sidechain consensus protocol, and the cross-chain communication protocol [93]. Sidechains allow different interactions between participating blockchains, being the most common the transfer of assets between the main chain and the sidechain (two-way peg) [125, 199]. A *two-way peg* works in the following manner: a user, operating on the mainchain, sends X tokens to a custom address that locks assets. Those funds are locked on the mainchain, and a corresponding number of tokens are created on the sidechain. The user can now use the tokens on the sidechain. Eventually, the user can transfer back the tokens to the main chain, which causes assets on the sidechain to be locked or destroyed, depending on the implementation. There are three major types of two-way pegs: simplified payment verification, centralized two-way pegs, and federated two-way pegs. *Simplified payment verification* (SPV) [37, 158] is done by *light clients*, which consist of blockchain clients that can verify transactions on the blockchain without having its entire state. The SPV client only needs the block headers; verifying that a transaction is in a block is to request a Merkle tree proof [205] including that transaction. In particular, transactions are represented as Merkle tree leaves. Given a leaf node as a target and a path comprised of nodes and its siblings to the target, verifying a Merkle tree proof of including the target is to reconstruct a partial Merkle tree root.

A relay solution is an SPV client for a source blockchain running on a target blockchain, enabling verification of transactions [89]. This verification enables conditional logic to occur on a target blockchain. Since relays are between blockchains and those blockchains are using behavior from others (bidirectionally or unidirectionally), relays include the presence of sidechains. This is saying, without a sidechain, there are no relay solutions.

Centralized two-way pegs, on the contrary, trust a central entity, benefiting in terms of efficiency. An example is an *exchange*, an organization, typically a company, that trades cryptocurrencies on behalf of its clients. However, Exchanges are a Notary Scheme, so we defer their explanation to Section 5.1.2. Disadvantages include a single point of failure and centralization. *Federated two-way pegs* try to decentralize the previous solution. In this solution, a group is responsible for locking and unlocking funds instead of just one. Standard implementations rely on multi-signature schemes, in which a quorum of entities must sign transactions to be deemed valid by the network. Although a better option, it does not eliminate centralization.

Figure 4 depicts a system based on the BTC Relay [79]. In *BTC Relay*, parties called *relayers* keep track of the block headers of the main chain (the Bitcoin network in the figure), and input them to the BTC Relay smart contract, hosted on Ethereum. This procedure builds a pool of Bitcoin headers that can be used (via their stored Merkle trees) to verify on-chain information, including the presence of transactions. This way, any party can request a transaction to be verified by the smart contract that holds the headers' knowledge (via SPV). Transaction validation can be relayed to deployed Ethereum smart contracts, allowing several use cases, for example, the issuance of tokens.

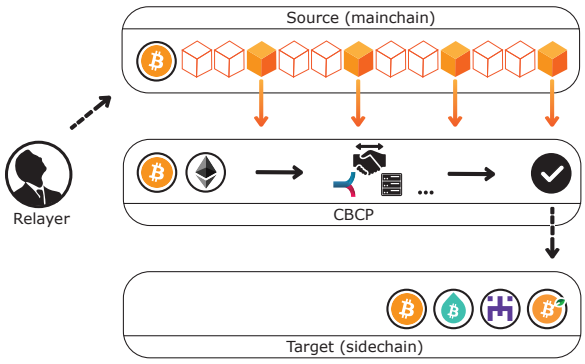

Fig. 4. A general sidechain system [79]

Zendoo is a cross-chain transfer protocol that realizes a decentralized, verifiable blockchain system

for payments [93]. The authors consider a parent-child relationship, where nodes from the sidechain can observe the mainchain's state, but the main chain can only observe the sidechains via cryptographically authenticated certificates. Zk-SNARKSs enable the authentication, validation, and integrity of the information provided by the sidechains via verifiable proofs [27]. Such proofs are used to generate certificate proofs for the mainchain, enabling a secure verification scheme.

5.1.2 Notary Schemes. A notary scheme involves a *notary* that is an entity that monitors multiple chains, triggering transactions in a chain upon an event (e.g., a smart contract is deployed) taking place on another chain [52]. Notary schemes are, in practice, instantiated as centralized exchanges (EXs) or decentralized exchanges (DEXs). The most popular centralized exchanges, by volume, as of the 8th of February 2021 are Binance[2], Coinbase[3], and Huobi Global[4]. Exchanges facilitate signaling between market participants by managing an order book and matching buyers and sellers. If the trust anchor is put on a centralized party, where it holds users' private keys or funds, the notary is a centralized exchange. Otherwise, if exchanges do not execute the trades on behalf of the users, only providing a matching service, they are considered decentralized exchanges. We present the protocols of two decentralized exchanges: 0x [223], and Uniswap [4].

0x implements a decentralized exchange as a set of smart contracts (called automated market makers), replacing an on-chain order book with a real-time price-adjustment model. 0x uses a hybrid implementation, "off-chain order relay with on-chain settlement", combining the idea of a state channel with settlement smart contracts. Two parties participate: *makers* and *takers*. Makers place orders on the exchange, providing liquidity for the network (a set of decentralized exchanges), while takers place orders matched with the makers' orders. 0x uses the ZRX token and the Ethereum blockchain to incentivize users to host and maintain order books (provide liquidity). In exchange, 0x makers choose the rewards they obtain for each trade - although they have to comply with the DEX policies under the possibility of the order not being disseminated. This approach relies on a smart contract set (smart contract) and several smart contracts representing the different tokens supported. First, a maker creates an order to exchange token A for B, at a given rate, right after it approves a DEX to access its balance of token A. A taker discovers this order and wishes to trade its tokens B for tokens A. The taker grants permission to the DEX to access its tokens, and the DEX performs the exchange after several validations (e.g., the order has not expired, and it has not been filled).

Uniswap is a set of smart contracts implementing an automated liquidity pool, serving as a decentralized exchange [4]. Each Uniswap pool provides liquidity for two assets based on the constant set as the reserves' product. Prices for each asset are provided by an on-chain price oracle smart contract. Uniswap can support ERC-20 to ERC-20 trades and even flash loans, a theme explored in the decentralized finance area. A flash loan is a type of loan that does not require collateral, as the debt is repaid within the transaction. Flash loans work because the borrowed asset to be paid within the transaction requesting it [4].

5.1.3 Hashed Time-Lock Contracts. Hashed time-locks contracts (HTLCs) initially appeared as an alternative to centralized exchanges, as they enable cross-chain atomic operations [34]. HTLCs techniques use hashlocks [35] and timelocks [36] to enforce atomicity of operations, normally between two parties. A trader commits to make the transaction by providing a cryptographic proof before a timeout to the other. This scheme allows for the creation of multiple outputs (such as multiple payments), depending on solely one hashlock. HTLCs are used in Bitcoin for conditional payments, or

[2]https://www.binance.com/en
[3]https://www.coinbase.com/
[4]https://www.huobi.com/

cross-chain payments (Bitcoin-Ethereum), i.e., *atomic swaps* [38, 68, 107]. Atomic swaps can be thought as a form of distributed commitment resilient to Byzantine adversaries. Thus, an atomic cross-chain swap is a distributed atomic transaction [108], settled on-chain.

Several projects implement HTLCs differently, providing different correctness guarantees. However, the general algorithm is quite similar in most of the solutions. Let us consider an HTLC-supported atomic swap between Alice (holding assets of type a in blockchain $\mathcal{B}_a$) and Bob (holding assets of type b in blockchain $\mathcal{B}_b$). An atomic swap can be realized as follows [26, 236]: 1) Alice generates and hashes a secret s, yielding h. The protection of a smart contract with hash h is called a hashlock because it will lock a smart contract - only parties with knowledge of secret s can know it since secure hash functions are pre-image resistant (i.e., a hash function cannot be inverted). Alice also creates a timelock t_b, corresponding to an upper bound in which the created hashlock can be unlocked, i.e., Bob can unlock the smart contract up to t_b, where t_b corresponds to a specified future time or block height; 2) Alice publishes the smart contract in $\mathcal{B}_a$. Bob verifies the deployment, and records h and t_b; 3) Bob publishes a smart contract in $\mathcal{B}_b$ locking b with hashlock h, but with timelock ta such that $t_a < t_b$, i.e., Alice can claim b before t_a. 4) Alice checks that Bob's smart contract has been published and gives as input secret s, before t_a, acquiring asset b. In practice, this triggers a transfer; 5) Bob now sends s to Alice's smart contract in the interval $[t_a, t_b]$, acquiring a. Note that if Bob issues the transaction after t_b, Bob will not obtain access to b. Some solutions utilize the notion of HLTC and enhance it, providing an additional on-chain trust anchor. In particular, two solutions are presented: XCLAIM [234] and the Ligthning Network (LN) [174].

XClaim uses a combination of HLTCs, collateralization, and escrow parties, realizing non-interactive cross-chain atomic swaps [234]. This protocol includes several actors: the requester, the sender, the receiver, the redeemer, the backing vault, and the issuing smart contract. Requesters lock coins to issue tokens, while the redeemer burns tokens to receive coins. The sender sends tokens, while the receiver receives them. After that, the vault smart contract fulfills requests of asset backing and ensures correct redeeming. An issuing smart contract issues and exchanges representations of a token (cryptocurrency-backed assets) and enforces the vault's correct behavior. Considering a transaction between Bitcoin and Ethereum, firstly, the vault locks collateral in Ethereum smart contracts. This collateral defines the amount of CBA that the vault can issue. A user that wants to issue Bitcoin-backed tokens sends Bitcoin to the vault. User A then sends a proof of transaction submitted to the Bitcoin mainchain to a chain relay, e.g., BTC Relay. The chain relay verifies the submitted transaction and alerts the issuing smart contract. The smart contract releases the Bitcoin-backed assets to the user. On the other hand, a user issues a transaction against the smart contract, locking/burning its backed tokens. The vault releases the Bitcoin to the user, and it submits a proof of the involved operations to the chain relay. The chain relay verifies the proof and only then releases the collateral to the vault. XClaim currently supports exchanges between Bitcoin and Ethereum[5]. The protocol execution consumes substantially lower Ether than traditional HTLCs.

LN enables high-volume, low latency micro-payments on the Bitcoin network [174]. LN is a payment scheme (i.e., an off-chain sidechain). LN allows several parties to open a payment channel, transact amongst them, and when all the intermediary payments are completed, the final output is sent to the mainchain. LN works as follows: 1) funds are placed into a multi-signature Bitcoin address (two-party multi-signature if only two people are transacting). In order for funds to be changed, two signatures are required. After that, the funds will be managed off-chain via commitment transactions (i.e., a commitment to pay part of the available funds to the other party); 2) Parties can now transact offline under the regime they choose; 3) To settle the payments performed off-chain, both parties sign a new exit transaction. Note that parties can unilaterally close the payment channel in case of conflict. LN is considered a precursor of HLTCs

[5]https://github.com/crossclaim/xclaim-sol

because its bi-directional payment channels allow payments to be routed across multiple payment channels using HLTCs.

5.1.4 Discussion on Public Connectors. Public Connectors started emerging as early as 2015 [241], when researchers and practitioners alike saw the potential in cross-chain transactions to support, for instance, atomic swaps [38, 68, 107], and payment channels [174]. Sidechains are the solutions increasing the main network's scalability by processing and batching large amounts of transactions before submission on the main blockchain [135, 173, 199]. Relays can fetch block headers from sidechains, enabling data verification [79, 131, 132]. While sidechains are mainly used on public blockchains, there are also permissioned blockchain sidechains [137]. We note that some sidechains may have a cross-chain mechanism realization HLTCs, being a solution belonging to multiple categories (e.g., [174]).

Most sidechains use Ethereum and have a sidechain consensus mechanism, which is allusive to bidirectional transfers [93]. Simple relay schemes, which verify transactions on other chains, such as BTC Relay, have a simple sidechain consensus, as the information flow is unidirectional [79]. In particular, validators can sign events that happened on the source chain (if validation happens across EVM-based chains) or transfer block headers (via users or aggregation chains) [182]. Liquid [74], and POA [11] rely on a consortium of validators running trusted hardware to execute smart contracts and validate transactions. Other solutions, such as Wanchain [85] rely on a trusted consortium, but without running trusted hardware.

However, sidechains suffer from several limitations. Safe cross-chain interactions are rooted in the assumption that the main chain is secure, i.e., the network cannot be successfully attacked. Compromising the main chain would invalidate the sidechain logic. Conversely, centralization in sidechains tends to exist to a higher degree than on mainchains, because typically there is a trade-off between decentralization-performance (e.g., lesser validating nodes versus higher throughput). Consequently, if an attacker can obtain control on a (potentially small) set of validators, funds can be stolen from users. Therefore, it is important to have different stakeholders with different incentives, diminish the likelihood of collusion, and rely on a reasonable quorum of validators to sign each transaction (e.g., 8 out of 11 versus 3 out of 10). If a sidechain has a strong security model, it may lead to a slow transaction settlement, stalling assets, and lowering liquidity. For example, the RSK sidechain [135] takes approximately the time to confirm 100 Bitcoin blocks (around 15 hours) to convert BTC to RBTC[6]. Finally, sidechains typically do not allow for arbitrary code to specify conditions on the pegging mechanism, thus not empowering them to develop more complex applications.

Notaries on the Public Connectors category are cryptocurrency exchanges. EXs have the majority of the market share, comparatively to DEXs. While EXs provide services to the end-user, decentralized exchanges tend to provide better exchange fees and security. The trade-off is, therefore, comfort and speed - security. This subcategory provides great flexibility at run-time because EXs and smart contracts that DEXs support triggers (e.g., stop-loss orders).

Notary schemes have to capture the logic of smart contracts in both chains. Although they can capture the full spectrum of interoperability – both at the value and mechanical levels (see Section 5.3), practical applications are limited. In summary, notary schemes are intermediaries between blockchains. EXs are notaries because they execute actions on behalf of the end-user (e.g., buy cryptocurrencies conditionally). DEXs are notaries because they provide matching for the end-users by pinning and advertising trade offers encoded in smart contracts.

The HTLCs category was the first one to allow asset exchange in a trustless way. HTLCs allow atomic swaps between different blockchains, funding bidirectional payment channels. HTLCs are flexible because they can be chained after each other [236], and therefore enable trades even if there is no direct connection between the trading parties. As they

[6]https://developers.rsk.co/rsk/

serve as programmable escrows, they represent the most trustless and practical approach of the three. However, hashed timelocks might lead to capital retention and unfair trade, as the trader issuing a cross-blockchain asset transfer may only provide the secret on specific conditions (exploring the spread of the cryptocurrency exchange rate) [234]. Many solutions are hybrid, sharing characteristics of HTLCs and sidechains, either exploring collateralization-punishment schemes rooted on smart contracts ([187, 194, 234], or locking-in and locking-out assets [45, 91, 151, 152]. HLTCs are practical solutions across public blockchains. HLTCs could also provide asset transfers between private blockchains, but only under the participation of a third party blockchain and a semi-trust environment [101], or if both parties belong to both private blockchains. Current efforts to address these limitations include Hyperledger Cactus [155].

Concluding, Public Connectors are the best approach to perform cryptocurrency trades and moving fungible and non-fungible assets across public blockchains. We encourage the reader to refer to some related surveys focusing on sidechains to complement this survey (see Section 3).

5.2 Blockchain of Blockchains

Blockchain of Blockchains are frameworks that *provide reusable data, network, consensus, incentive, and contract layers for the creation of application-specific blockchains (customized blockchains) that interoperate between each other.* We briefly present Polkadot [49, 227] and Cosmos [130], the most widely adopted Blockchain of Blockchains in terms of market capitalization[7]. A detailed comparison between Polkadot, Cosmos, and Ethereum 2.0 (the baseline) is deferred to Appendix D. Other Blockchain of Blockchains include Ark [12], Komodo [129], and AION [201].

Wood proposes *Polkadot*, a network that aims to connect blockchain networks [227]. Polkadot provides the foundation for *parachains*, i.e., "globally-coherent dynamic data structures" hosted side-by-side. Parachains are, thus, the parallelized chains that participate in the Polkadot network. Specialized parachains called bridges link independent chains [227]. Polkadot is based on *Substrate*, a framework for creating cryptocurrencies and other decentralized systems. It guarantees cross-language support with WebAssembly, a light client, and off-chain workers, allowing for integration with other technologies.

Polkadot enables interoperability based on state transition validation, done by the chain-relay validators. Parachains communicate through the Cross-chain Message Passing Protocol (XCMP), a queuing communication mechanism based on a Merkle tree [170]. Communicating state transition proofs from parachain to relay chain is achieved via an erasure-coding scheme. Polkadot scales by connecting up to 100 parachains directly to the relay chain in the short-medium term. A long-term solution is being studied, where second and third-level parachains are added in parallel.

Cosmos is a decentralized network of independent parallel blockchains, called *zones* [130]. The zones are essentially Tendermint blockchains [208]. Zones can transfer data to other zones directly or via *hubs*. Hubs minimize the number of connections between zones and avoid double spendings. For example, zone A can connect to zone B via Hub C and receive tokens from zone B. Zone A would need to trust the tokens from zone B and Hub C. This scheme allows zones to maintain a reduced number of connections. Both ways utilize the inter blockchain communication protocol (IBC) [113].

IBC resembles the Internet network layer as it routes arbitrary data packets to a target blockchain. A target blockchain can know that a certain ordered packet with arbitrary data came from another blockchain. By handling transportation and order, the protocol has several steps to achieve cross-zone transactions. First, each chain involved tracks the headers of the others, acting as a light client. When a transfer is initiated, the protocol locks the assets on the origin chain. After that, the proof is sent to the target blockchain, which then represents the locked assets. A similar mechanism is used to

[7]USD 22.1B and USD 3.6B respectively, as of February 2021

Table 4. Comparison of Blockchain Engine interoperability solutions [130, 166]

	Communication		Properties						Community	
	Cross-chain Protocol	Cross-blockchain interoperability	Consensus Mechanism	Security assumption	Validator number	Maximum Throughput	Number of instances	Smart Contracts	Launch	Roadmap
Polkadot [227] ✓	XCMP	●	BABE and GRANDPA	SM	197	10^3	200	WASM	November 2019	Main network launch
Cosmos [130] ✓	IBC Protocol	◐	Tendermint	SM	125	10^3	> 70	WASM	March 2019	Governance updates
ARK [12] ✓	SmartBridge	◐	Delegated proof of stake	M	51	18.5	Unlimited	WASM*	May 2019	ARK Swap Market
AION [201] ✓	Interchain transactions	○	Proof of intelligence	M	×	×	×	Aion Language	April 2018	Market assimilation

✓our description was endorsed by the authors/team
× not known
∗ some languages compilable to WASM, such as Go and .NET, but not all of them
● can interoperate with instances of the same blockchain engine. Interoperate with more than two heterogeneous blockchains
◐ can interoperate with instances of the same blockchain engine. Interoperate with up to two heterogeneous blockchains
○ can interoperate with instances of the same blockchain engine

recover the original tokens. This scheme allows for interoperability among Tendermint blockchains. Other kinds of blockchains can interoperate with a Cosmos chain via peg zones. Peg zones resemble the pegged sidechain mechanism [15], in which a representation of the locked token of the source blockchain is created on the target blockchain.

Cosmos abstracts the development of a blockchain into three layers: networking, consensus, and application. Tendermint BFT realizes the networking and consensus layers. The Tendermint BFT engine is connected to the application layer by a protocol called: the Application Blockchain Interface (ABCI). The Cosmos SDK realizes the applicational layer, allowing developers to develop smart contracts in languages that can be compiled to WASM[8].

5.2.1 Discussion on Blockchain of Blockchains. Blockchain of Blockchains implementations are similar to relays and sidechains, as there is typically the main chain (often called relay chain) that connects the secondary chains, which can be application-specific blockchains. This scheme allows high throughput and flexibility to the end-users, providing interoperability capabilities between their platform instances. For example, Cosmos's Tendermint-based blockchains interoperate (instant finality assured), while Polkadot provides interoperability on Substrate-based blockchains (for instance, via Cumulus[9], a tool for connecting a blockchain to Polkadot). To connect to other chains, Cosmos, Polkadot, AION, and utilize a mechanism similar to pegged sidechains or hashlock time contracts (ARK [12]) to interact with other blockchains, commonly called bridges.

Table 4 maps out the current blockchain engine landscape by extracting and evaluating their main characteristics. Some information was not possible to obtain due to the lack of details on the whitepapers. It is possible to observe that Blockchain of Blockchains is very recent: Polkadot's test network, Kusama [171], was released in November 2019; Cosmos' main network was launched in March 2019. ARK launched in May 2019. AION launched in April 2018. Blockchain of Blockchains has different cross-chain communication protocol, e.g., in Polkadot, cross-chain message passing[10]; in Cosmos, the inter-blockchain communication protocol [130]. Cosmos and Polkadot have some differences regarding their approach: in Cosmos, the idea is to provide blockchains tailored to specific applications. IBC is more generic than XCMP, letting users customize their zones with higher freedom: security and validation are decided per zone. Polkadot restricts this customization but offers a shared security layer, making a trade-off security-customization.

The security assumptions criteria depict the number of nodes assumed to be honest. A supermajority (SM) assumes that at least two-thirds of the nodes are honest, a common condition required by Byzantine fault-tolerant consensus algorithms ($n > \frac{2}{3}$), while the majority (M) assumes at least half of the nodes are honest ($> \frac{1}{2}$). The validator number on a network comes with a trade-off: while a higher number is generally better for decentralization and robustness, it

[8]https://blog.cosmos.network/announcing-the-launch-of-cosmwasm-cc426ab88e12
[9]https://wiki.polkadot.network/docs/en/build-cumulus
[10]https://wiki.polkadot.network/docs/en/learn-crosschain

comes with an increase of latency towards block production – and consequently lower throughput. Polkadot currently has around 297 validators, and this number is gradually increasing in the short-term to support up to 100 first-level parachains. At the time of writing, Polkadot is developing bridges for Bitcoin [234], Tendermint, Libra [140], and Ethereum. Interoperability between parachains is provided by Substrate.

Currently, Cosmos has 125 validators. The number of validators can rise to 300. Currently, there are around 70 zones, and "the number is growing". While Cosmos does hold a limit for zones (as each zone is self-sovereign), there is no limit for how many zones can be attached to a Hub. Cosmos can interoperate with Etheruem. The Cosmos SDK provides interoperability between zones. Cosmos supports multiple peg zone implementations for Bitcoin and one for Ethereum. ARK has 51 validators, which can validate the transactions of a number of blockchains bound to the company's physical resources (instances managed by ARK). ARK can send and receive ERC-20 tokens to the Ethereum blockchain. We found no information regarding AION's validator number, throughput, or maximum sub-chains [201]. The theoretical throughput of the presented solutions varies: Polkadot's relay chain supports around 1000 transactions per second, considering that a block can include around 7,000 transactions at a 6-second block time (considering current weights, March 2021). Cosmos theoretical throughput can achieve up to dozens of thousands of transactions per second (tps) with two validators. With 64 validators, it falls into several thousand transactions per second. ARK can achieve around 18.5 transactions per second, relying on a proof of work consensus. The number of validators is set to 51. ARK is not a completely decentralized solution, as it manages instances of ARK blockchains. There is no theoretical limit of bridge chains, except the service provider resources. Several optimizations are being done in Cosmos, Polkadot, and ARK, to increase the throughput. The AION The project looks deprecated and stalled. As stated, the "white paper is both ambitious and experimental" [201]. AION is now a part of a larger project called the *Open Application Network* (OAN).

Cosmos and Polkadot support smart contracts in languages compilable to WASM (Web Assembly), which means developers can write them in languages such as Go, C++, and JavaScript. AION would support domain-specific languages, Aion language. Blockchain of Blockchains instances achieve inter-chain interoperability by a common point of contact, the "connector", analogous with Hyperledger Fabric channels [9]. The connectors are the relay chain, the Cosmos Hub, the AION-1 blockchain, and the ARK main net if the technology is Polkadot, Cosmos Network, AION, or ARK, respectively. In Polkadot, the connector provides shared security. The relay-chain (the chain that coordinates consensus and communication between parachains and external blockchains) connects parachains and parachains to bridges. In Cosmos, the connector is loosely coupled to blockchains, providing greater flexibility than Polkadot. We could not extract meaningful considerations about AION's connector. In ARK, it looks like the connector is centralized at the expense of developability and ease of use. Concerning cross-blockchain interoperability, all solutions rely on *bridges* or *adapters* that route transaction from a particular blockchain type to another.

While the provided features can be desirable for end-users, blockchain-engines do not interoperate with each other. In light of this fact, end-users are obligated to choose between existing solutions, leading to sub-optimal leveraging of available resources. Therefore, participant networks have constraints on interoperability, ending at relying on a single blockchain engine solution. Some authors defend that blockchain engine approaches are not universally accepted and cannot eliminate fragmentation [1]. Some solutions are even centralized, in the sense that its code is not open-source, and the end-user needs to use an SDK to access core functionalities (e.g., [12, 201]). However, ongoing work on building a Tendermint light client for GRANDPA, which would allow Polkadot to interact with Cosmos may allow blockchain engine interoperability in the short-medium term. Thus, in theory, interoperability across Blockchain of Blockchains can also be achieved via the relay chain technique (i.e., a blockchain engine can be a sidechain of other blockchain engines; validation can happen via SPV).

Moreover, Blockchain of Blockchains requires transaction fees to keep the network operating. Given enterprise blockchain systems, a question could be posed: at which point shall an organization pay fees to sustain its business model across several blockchains? While Cosmos can provide flexibility configuring a zone, on Polkadot, this can be harder. Therefore, Blockchain of Blockchains can provide an optimal leveraging for public infrastructures, but that is not necessarily the case for private blockchains.

5.3 Hybrid Connectors

The *Hybrid Connector* category is composed of interoperability solutions that are not Public Connectors or Blockchain of Blockchains. Directed to both public and private blockchains, Hybrid Connectors attempt at delivering a "blockchain abstraction layer" [224], capable of exposing a set of uniform operations allowing a dApp to interact with blockchains without the need of using different APIs [83]. We derived a set of sub-categories from the studies available: *Trusted Relays*, *Blockchain Agnostic Protocols* (including *Blockchain of Blockchains*), and *Blockchain Migrators*. Trusted relays are directed to environments where a blockchain registry facilitates the discovery of the target blockchains. Typically, such a scheme appears in a permissioned blockchain environment, where trusted escrow parties route cross-blockchain transactions. As the name suggests, Blockchain-agnostic protocols provide technology-agnostic protocols for interoperation between distributed ledger systems but do not guarantee backward compatibility. In other words, to use such protocols, their source code has to be changed to existing blockchains to use such protocols. Solutions from the blockchain of blockchains category aim to provide mechanisms for developers to build cross-chain dApps. The blockchain migrators sub-category aggregates solutions that perform data migration across blockchains, which resemble the notary schemes discussed in Section 5.1.2 (as there is typically a centralized party mediating the migration process).

We introduce each sub-category, presenting only one illustrative example of each for the sake of space. Appendix E depicts a complete list of Hybrid Connectors. Evaluation tables for each sub-category are discussed in Section 5.3.4.

5.3.1 Trusted Relays. Trusted relays are trusted parties that redirect transactions from a source blockchain to a target blockchain, allowing end-users to define arbitrary business logic. These solutions imply managing different APIs, in which cross-chain consensus may be modular.

Hyperledger Cactus (Cactus), previously known as Blockchain Integration Framework, uses an interoperability validator network that validates cross-chain transactions, optionally using a trusted escrow party [155]. However, decentralized validators are implemented as well – making this project move towards a decentralized trusted relay. Cactus allows a party or a set of parties to issue transactions against several ledgers, similarly to some notary scheme solutions [106, 190]. The interoperability is enabled through a set of *interoperability validators*, which are participants from the source and target blockchains. Such validators collect cross-chain transaction requests, sign and deliver them. A CB-Tx is deemed valid, given that a quorum of validators signs them. It is then assumed that the blockchains participating in the network know how to address each other. However, trusted escrows can be replaced by decentralized parties. Currently, Hyperledger Cactus[11] supports Hyperledger technologies (e.g., Fabric, Besu), Corda, and Quorum. The roadmap predicts integration with public blockchains and blockchain migration capabilities.

5.3.2 Blockchain-Agnostic Protocols. Blockchain-agnostic protocols enable cross-blockchain or cross-chain communication between arbitrarily distributed ledger technologies by providing a blockchain abstraction layer. These solutions enable BoBs, "a system in which a consensus protocol organizes blocks that contain a set of transactions belonging

[11] https://github.com/hyperledger/cactus

to CC-dApps, spread across multiple blockchains. Such system should provide accountability for the parties issuing transactions on the various blockchains and providing a holistic, updated view of each underlying blockchain" (Section 2.3). Typically, the cross-chain consensus is fixed, and business logic is more restricted.

The *Interledger Protocol (ILP)* can be considered a decentralized, peer-to-peer payment network [209]. It firstly adopted a generalized hash locking scheme to enable asset transfers, and it was directed to cryptocurrency transfers. Nowadays, ILP is technology-agnostic, defining a "lowest unit common denominator" across distributed ledgers, blockchains, fiat payment networks, the ILP packet.

ILP sends payment information in packets by leveraging a network of connectors, which route such packets. At the core of Interledger is the Interledger Protocol (ILPv4) [115], which defines how senders, routers (or node, or connector), and receivers interact. Typically, the connector is a money packet router. The root of trust is then the connector, which has to be trusted: companies can settle payments via the routers, given that clearance of such payments is done afterward while being protected by the law. A sender is an entity that initiates a value transfer. A router applies currency exchange and forwards packets of value. The receiver obtains the value transmitted. ILPv4 is a request/response protocol enabled by ILPv4 packets. Each packet contains transaction information, and can be divided into *prepare*, *fulfill*, and *reject* packets. A sender node initiates an exchange of value by sending a *prepare* ILPv4 packet to a receiver. When a receiver obtains the prepared packet, it sends the response back to the sender via routers. The response may be a *fulfill* packet, whereby a transaction has been successfully executed, or a reject packet.

Several specifications for Interledger and related protocols are available[12]. The Interledger Protocol is discussed by a W3C community group[13] and has a proposal that "describes data structures and formats, and a simple processing model, to facilitate payments on the Web"[14]. The interledger protocol cannot integrate with existing blockchains: each one must be adapted to use ILP. A disadvantage is that Interledger does not support the transfer of non-fungible tokens (such as ERC-721[15] tokens).

5.3.3 Blockchain Migrators. Blockchain migrators allow an end-user to migrate the state of a blockchain to another. Currently, it is only possible to migrate data across blockchains, although moving smart contracts is also predicted [155].

Fynn et al. present an abstraction for smart contracts to switch to another blockchain consistently, moving the state required by the transaction to the target blockchain and execute it [92]. The authors call such abstraction the *Move* operation. The operation works as follows: first, it locks a smart contract on the source blockchain; next, the Move protocol recreates the smart contract in the target blockchain. This method allows arbitrary states to be transferred between blockchains. For example, it allows transferring cryptocurrencies by creating tokens on the target blockchain backed-up by locked tokens on the source blockchain (similarly to pegged sidechains). This method was tested on Ethereum and Hyperledger Burrow (based on Ethereum). The solution assumes the same cross-blockchain smart contracts utilize the same virtual machine, which can be limiting. Furthermore, for such a solution to be deployed, it requires Solidity changes and possibly a soft fork on Ethereum.

5.3.4 Discussion on Hybrid Connectors. This section defined the hybrid connector category and its sub-categories: trusted relays, blockchain-agnostic protocols, and blockchain migrators.

[12] https://github.com/interledger/rfcs
[13] https://www.w3.org/community/interledger/
[14] https://w3c.github.io/webpayments/proposals/interledger/
[15] http://erc721.org/

Regarding centralization, almost all adopt a decentralized model. Permissioned blockchain solutions are less flexible, as all involved participants are identified. In particular, trusted relays endorse connections made in a peer-to-peer fashion, upon previous agreement [1, 95]. However, Abebe et al.'s work pose some limitations: interoperating networks require a priori knowledge of each other's identities and configurations, hence being static. A discovery service could be implemented using a blockchain registry or a pub-sub mechanism [95], in which networks could be added and removed. In trusted relays, it is not completely clear the mechanisms to minimize malicious relay services, apart from replication (whereby the risk of a censorship attack is reduced but not erased). Hyperledger Cactus could be a true enabler of interoperability, given that a (decentralized) trusted blockchain registry would be deployed, and public escrow parties could replace the overlay of trusted parties. Cactus could, therefore, make the transition between a trusted relay to a semi-trusted relay or even a trustless relay.

Blockchain-agnostic protocols will be better positioned to offer interoperability to existing and yet to-exist blockchains, but most do not grant backward compatibility and lack the flexibility to define business logic. This inflexibility is inherent to the provided homogeneous interfaces (containing roles, methods, data, message formats, for instance, [83]); at least such solutions scale slowly, as adding methods compatible with all the supported blockchains incur in a polynomial effort. However, this category might resemble some of the trusted relay solutions. In particular, both Cactus [155], and SCIP [83] rely on connectors and validators and gateways, to access the underlying blockchains. The gateway paradigm implies a (semi) trusted gateway having read/write access to the shared ledger of the blockchain, and often they are expected to participate in the consensus mechanism of the blockchain [101]. While there is a higher trust requirement, gateway approaches might be the most suitable to solve interoperability across private blockchains if gateways are framed in a legal and regulatory framework. Proper solutions for enterprises, gateways need infrastructure comprising, for example, public identifiers, a set of connectors, and validators (which Cactus could provide), among others.

From the blockchain of the blockchains category, we highlight Hyperservice, a peer-reviewed paper, and Overledger. Hyperservice tries to achieve full dApp atomicity by introducing the concept of *stateless smart contracts*. Using a stateless smart contract, a CC-dApp can load a clean state for a contract, using a valid block. While it can partially solve forks in the underlying blockchains, a CC-dApp utilizes, the application of this concept paves a direction to decouple smart contract execution from the consensus layer [143]. Overledger is a sorted list of messages that are interpreted as the state of a cross-blockchain application. While this is an exciting approach to blockchain interoperability, the solution is proprietary, hindering community efforts for more complex solutions.

Blockchain migrators respond to an enterprise need: migration in case of disaster or performance issues [16, 21]. The two presented solutions can only provide data portability across a small set of public blockchains. It is currently impossible to reproduce the chain of events via smart contracts, as that requires a smart-contract translator functionality.

A limitation that we identified in the context of Hybrid Connectors is that most solutions do not support hard forks (i.e., the separation of a blockchain into two different blockchains) nor propose a solution for eventual forks, unlike some public connectors (most HTLCs and notary schemes). Forks do not happen regularly, and some solutions offer a quick analysis of the problem and acknowledge their importance [116, 155, 215]. However, this is still a problem that can affect the dependability of cross-chain dApps; dealing with forks is still an open issue. For instance, the protocol used in Hyperservice is unable to revert any state update to smart contracts when a dApp terminates prematurely, i.e., it does not grant atomicity. If one does not have atomicity guarantees, it forces the cross-blockchain application into an inconsistent state when a fork occurs. This can put at risk the purpose of the project: functional cross-blockchain applications. The same problem applies to, for instance, Overleder [177].

While one might be tempted to conclude that standardization could improve cross-blockchain API design, some argue that APIs are unlikely to generalize well across radically different technologies. Blockchain-agnostic protocols are more likely to be standardized than APIs, as shown historically by successful standards efforts such as HTTP or the TCP/IP family. Finally, solutions that prove cross-smart contract capabilities are emerging, but still in development [1, 116, 143, 189, 215].

6 DISCUSSION, USE CASES AND RESEARCH QUESTIONS

This section presents a comprehensive summary of each blockchain interoperability category we extracted and our considerations about blockchain interoperability. Then it presents use cases and finishes with answers to the research questions we proposed.

6.1 Discussion

Although blockchain interoperability is a complex technology, connecting blockchains ends up being a manageable approach, despite differences in, for example, data structures, digital signature schemes, transmission protocols, verification mechanisms, consensus mechanisms, token issue mechanisms, and smart contract language. However, "there is a scant effort today to address the standardization of the various infrastructure building blocks – messages, data formats, and flows – to support the interoperability across blockchains" [101].

Different categories of solutions approach the interoperability problem differently. Our paper firstly introduced Public Connectors in Section 5.1 and stressed their importance. Token exchange is arguably no longer the whole scope of blockchain interoperability [143]. Instead, various interoperability approaches emerged in the last years, whereby many of them aimed at generalizing blockchain interoperability. In particular, emerging solutions can be categorized as Hybrid Connectors, which provide cross-blockchain communication, and Blockchain of Blockchains, which allow an end-user to create customized, interoperable blockchains at the expense of vendor lock-in.

Public connectors are the most cited among industry and academia, as they provide practical solutions to real-world problems: asset transfers. As these were the first solutions to emerge, not surprisingly, some may not succeed. It seems that the merge of sidechain and protocols relying on an escrow party (enforced by smart contracts) are the most suitable solutions for interoperability among public blockchains. We argue that the flexibility, decentralization, and security of such proposals can be utilized for secure interoperability. However, creating and maintaining a decentralized application using several blockchains was difficult - and hence the Blockchain of Blockchains solutions appeared. Those can facilitate blockchain adoption while providing built-in interoperability among instances of the same platform, whereas variations of the solutions mentioned above can be used to bridge Blockchain of Blockchains to other blockchains.

While Blockchain of Blockchains, such as Cosmos or Polkadot provide a consensus engine and a security infrastructure to build blockchains, blockchain of blockchains aims at developing solutions using different infrastructures. In particular, Cosmos and Polkadot might progress towards *homogenity*, as they support only the creation of Tendermint-based blockchains and Substrate-based blockchains, respectively. While they provide interoperability capabilities, mainly on the chains relying on their technology and other desirable features (shared layer of security, decentralization, governance, better scalability), the end-users choice will be tied to specific implementations. Paradoxically, such solutions might contribute to data and value silos, as solutions built with them cannot connect with an arbitrary blockchain [1]. Despite this fact, one could argue that this problem can be alleviated by building bridges/adapters. These solutions are promising but are challenging to integrate with legacy systems and, generally, private blockchains - and hence the hybrid connectors started appearing.

Hybrid Connectors, specifically blockchain migrators and blockchain of blockchains, progress towards a user-centric, blockchain-agnostic view, enabling enterprise-connected CC-dApps. Arguably, the most suitable solution for connecting private blockchains is the usage of blockchain-agnostic protocols; however, they do not grant backward compatibility (as all previous solutions have to be adapted to integrate the adopted communication protocol). To overcome this fact, the short-medium-term solution would be using trusted relays. An interesting way for trusted relays to venture is by decentralizing the escrow party: from a set of trusted validators to a network of public nodes. It then follows from this survey that one could perceive trusted relays and blockchain-agnostic protocols to be good solutions to link private blockchains; and sidechain, smart-contract-based protocols suitable to solve interoperability among public blockchains.

A network of blockchain engine-powered blockchains can be leveraged using Hybrid Connectors. For instance, there is a possible synergy between Cosmos and the Interledger Protocol: when a user wants to make an in-app payment with fiat currency (e.g., dollars) within a Cosmos zone, he or she can rely on the interledger protocol as a payment rail. If using cryptocurrencies to pay (e.g., Bitcoin), the interledger router can route the transactions for a payment channel (e.g., Lightning Network), providing more trustful interaction. To connect this ecosystem to private blockchains, bridges have to be developed. To make such bridges trustable, a possible solution would be to elect a group of validator nodes, via an overlay network, that participates in the consensus of public blockchains and private blockchains. This way, cross-chain, and cross-blockchain transactions can be endorsed.

It is worth mentioning that several cross-chain programming languages are appearing, such as the Hyperservice Language [142] and DAML [72]. DAML provides a unified Blockchain programming model by abstracting the underlying blockchains and exposing a higher-level abstract ledger on top, similarly to HSL. DAML has different integration degrees: DAML as an application on the target platform; and integration where the DAML runtime engine validates transactions. Programs compiled on such languages can run on top of a BoB platform.

To conclude this discussion, we recall to the reader that blockchain development has been done in silos since its inception. New solutions for blockchain interoperability started emerging as of 2017, and, perhaps not surprisingly, such solutions are also being adopted in silos. While Public Connectors methods are commonly used nowadays, we focus on Blockchain of Blockchains and Hybrid Connectors. Blockchain of Blockchains and Hybrid Connectors allows interoperability between blockchains and other distributed ledger technologies and enterprise systems in the medium term. This promotes the development of blockchain interoperability standards. While blockchain matures, industries will tend to incorporate this technology into their business processes. Then, we predict that mass adoption will follow.

6.2 Supporting Technologies and Standards

Besides the presented solutions, there is work towards the support and standardization of blockchain interoperability. Blockchain interoperability standards attempt to create a "standardized transaction format and syntax", which is common to all blockchains, and secondly, a "standardized minimal operations set," common to all blockchains [103]. In particular, a standardized format is important because while fungible and non-fungible assets have a single, well-defined representation in each blockchain, arbitrary data can be manipulated freely. First, we introduce indirect contributions that promote blockchain interoperability and then the existing standards.

Recent efforts are visible in enabling heterogeneous smart contract integration through service-orientation [84], allowing external consumer applications to invoke smart contract functions uniformly. A cross-blockchain data storage solution becomes a feasible solution to achieve application interoperability, whereby applications rely on one blockchain.

Some dApps[16] already leverage the InterPlanetary File System (IPFS) [28] to create a common storage, adjacent to the blockchain. The InterPlanetary File System provides a peer-to-peer network for storing and delivering arbitrary data in a distributed file system, potentially facilitating the transfer of data across blockchains [23]. Organizations are working on standardizing digital assets. The Token Taxonomy Initiative[17] is a consortium dedicated to digital token standardization. It proposes a standard to identify tokens' behavior, properties, and control interfaces according to a token classification hierarchy. This project allows application developers to utilize a standard code set for interacting with tokens regardless of the blockchain platform, thus incentivizing blockchain interoperability. In the context of general interoperability, the Ethereum ERCs are a *de facto* standard[18].

Oracles are mechanisms that software systems provide an external source of truth for blockchains [156], and they can be centralized or decentralized [5]. Typically, centralized oracles are not as dependable as decentralized oracles, as they constitute a single point of failure.

Hyperledger Avalon [147] defers intensive processing from the main blockchain to an off-chain channel to support centralized yet trustable oracles (by using trusted execution environments). Since multiple blockchains can use the same data, it fosters interoperability.

Open source projects like Hyperledger Indy[19] and Hyperledger Aries[20] operate in the field of digital identity and self-sovereign identity. Central concepts of self-sovereign identity are decentralized identifiers (DIDs) [179] and verifiable credentials [50]. Decentralized Identifiers can be created, managed, and shared using Zero-Knowledge Proofs (ZKPs) mechanism, even allowing to create new access control models [22]. Such technologies allow for identity portability, enabling cross-blockchain identities [110].

So far, the presented standards are called DLT/Blockchain Enabling Technology Standards because they focus on standardizing elements that blockchains can use, as opposed to DLT/Blockchain Generic Framework Standards [141]. These refer to standardization of blockchain interoperability data and metadata formats, identity, and protocols, namely the IETF, ISO, Enterprise Ethereum Alliance, IEEE, The EU Blockchain Observatory & Forum, and W3C.

At the Internet Engineering Task Force (IETF), work is being done defining a set of drafts that guide the implementation of ODAP, a protocol using gateways [19, 102, 105]. The ISO Technical Committee 307 works towards the "standardization of blockchain and distributed ledger technologies"[21], but did not produce any standard yet. Subgroup 7 (ISO/TC/SG7) focuses specifically on interoperability. The Enterprise Ethereum Client Specification, currently on its seventh version, "defines the implementation requirements for Enterprise Ethereum clients, including the interfaces to external-facing components of Enterprise Ethereum and how they are intended to be used", including cross-chain interoperability [6]. The IEEE Blockchain Initiative[22] and the IEEE Standards Association[23], through the IEEE Standards P3203, P3204, and P3205 [24] work at providing "interfaces and protocols of data authentication and communication for homogeneous and heterogeneous blockchain interoperability". The EU Blockchain Observatory & Forum by the European Commission aims to 1) the monitoring of blockchain activities in Europe, 2) the management of the source of blockchain knowledge, 3) the creation of a forum for sharing information, and 4) the creation of recommendations on the role the EU could play

[16] https://ethlance.com/
[17] https://tokentaxonomy.org/
[18] https://eips.ethereum.org/erc
[19] https://www.hyperledger.org/projects/hyperledger-indy
[20] https://www.hyperledger.org/projects/hyperledger-aries
[21] https://www.iso.org/committee/6266604.html
[22] https://blockchain.ieee.org/standards
[23] https://standards.ieee.org/
[24] https://blockchain.ieee.org/standards

in blockchain [81]. The same entity points out the likelihood of an increasing number of standards and adoption within governments [64]. The W3C, via the Interledger Payments Community Group [25], is connecting payment networks, including decentralized ledgers. Other organizations working in the area include BIA, BiTA, BRIBA, BSI, CESI, DCSA, EBP, GS1, and MOBI [224].

Standardization efforts focused on a specific blockchain (DLT/Blockchain Platform-Specific Standards) are, for example, the 0302 Aries Interop Profile [26] and the Hyperledger Fabric Interoperability working group [27].

Multiple standards will likely arise and be used, for each vertical industry, as there is a lack of generalized interoperability standards. Standards are then reused across industries (e.g., IEEE P2418.5). Solving interoperability in a specific sector would then pave the way for standards in other industries because the main requirement is domain expertise (ontologies are good starting points for standardization) [141]. The heterogeneity created by standards will pose a regulation challenge, as blockchains may spread across different jurisdictions [25].

6.3 Use Cases with Multiple Blockchains

In this Section, we present use cases with multiple blockchains. More use cases can be found in Appendix F.

The industry is still applying blockchain to use cases using only one blockchain. Consequently, it is expected that use cases with multiple blockchains are rare. Notwithstanding, according to the existing resources, it seems that there is considerable interest in use cases using multiple blockchains. As long as the technologies mature, novel, disruptive use cases may be found. For the sake of space, we present some general use cases involving an IoB [188]. We refer readers to Appendix F for more use cases relative to Public Connectors, Hybrid Connectors, and Blockchain of Blockchains.

The first big IoB use case is asset transfers, where users can transfer assets from one blockchain to another. While some approaches implement this use case in an ad-hoc way, the emergence of central bank digital currencies (CBDCs) [150, 202], requires further efforts and standardization [57]. A CBDC is a digital version of a sovereign currency of a nation. A CBDC is issued by central banks, where each unit represents a claim on the value held by such central bank. Many blockchains features are appealing to implement CBDCs, particularly the offered immutability, transparency, and trust distribution. Some central banks are already experimenting with blockchain, including the Monetary Authority of Singapore and the Bank of Canada [188]. As each CBDC can be implemented with a blockchain, and each central bank might choose a different technology, interoperability between them is achieved using an IoB or even a BoB.

Another major use case is interoperability across supply chains [155, 188]. A supply chain is a chain of value transfer between parties, from the raw product (physical or intellectual) to its finalized version. Managing a supply chain is a complex process because it includes many non-trusting stakeholders (e.g., enterprises, regulators). As many markets are open and fluid, enterprises do not take the time to build trust - and instead, rely on a paper trail that logs the state of an object in the supply chain. This paper trail is needed for auditability and typically can be tampered with, leading to blockchain's suitability to address these problems [224]. A key challenge of blockchain-based supply chains is to interoperate with other DLT systems. Interoperability granted each participant of the supply chain (e.g., supplier, manufacturer, retailer) can participate at several supply chains (and thus several blockchains) using a single endpoint, simplifying the interaction process while reducing costs. Other use cases comprise connecting Hyperledger Fabric and Ethereum with Singapore Exchange and Monetary Authority of Singapore via node integration and EVRYTHNG, a product connecting multiple chains via API to digitize products [224].

[25] https://www.w3.org/community/interledger/
[26] https://github.com/hyperledger/aries-rfcs/tree/master/concepts/0302-aries-interop-profile
[27] https://wiki.hyperledger.org/display/fabric/Fabric+Interop+Working+Group

Finally, identity and data portability can be provided by an IoB approach. Identity paradigms like self-sovereign identity [22] can increase identity portability by providing users control of their identities. Typically, this is achieved by rooting user credentials in a blockchain. Hence, if blockchains can communicate with identity providers that are blockchains, one can use the same identity in different blockchains. Data portability complies with blockchains, allowing blockchain users to use their data outside of a blockchain without requiring significant effort.

6.4 Answers to the Research Questions

In this section, we provide answers to the presented research questions (further elaborated on Section A.1).

(1) **What is the current landscape concerning blockchain interoperability solutions, both from the industry and the academia? i.e., what is the current state of blockchain interoperability solutions?**
The first step towards blockchain interoperability has been creating mechanisms allowing the exchange of tokens (e.g., cryptocurrencies). We categorized such solutions as Public Connectors (Section 5.1). Such category comprises Sidechains and Relays (Section 5.1.1), Notary Schemes (Section 5.1.2), and Hash Time Lock Contracts (Section 5.1.3). This category provides an idea of the emergence of blockchain interoperability - but this area no longer applies solely to token exchanges between homogeneous blockchains.
Novel blockchain interoperability approaches are Blockchain of Blockchains (see Section 5.2) and Hybrid Connectors (Section 5.3). Hybrid Connectors fall into four sub-categories: trusted relays (Section 5.3.1), blockchain-agnostic protocols (Section E.2), and blockchain migrators (Section 5.3.3. We also analyzed related literature reviews on blockchain interoperability, in Section 3.

(2) **Is the set of technological requirements for blockchain interoperability currently satisfied?**
There are two requirements for realizing technical interoperability [52]: a pair of sufficiently mature blockchains to build artifacts that promote interoperability and "some application or need that cannot be served by implementing it on a single blockchain." There are several blockchains that can be considered mature enough to support applications built on top of them [9, 97, 130, 227]. On the other hand, interoperability regarding blockchain needs to have the necessary infrastructure and facilitating technologies. In particular, the production of standards [110, 212] technologies like decentralized identifiers [179], verifiable credentials [50], cross-blockchain communication protocols [45, 233, 234], and the representation of blockchain smart contracts [110] can foster the likelihood for blockchain interoperability standards and solutions, as they remove considerable barriers to blockchain interoperability.
On the other hand, there is a set of cross-blockchain use cases that validate the need for interoperability, which will inevitably foster it [25]. In conclusion, the set of critical requirements for blockchain interoperability is currently satisfied, but there is still work to be done at standardization and interoperability across public-private and private-private blockchains.

(3) **Are there real use cases enabling a value chain coming from blockchain interoperability?**
Regarding the third research question, some authors defend that blockchain interoperability is important and crucial for the survivability of this technology [103, 143**?**]. Standards are paving the way for blockchain adoption [71, 110]. It is likely that "forward-looking interoperability standards are most likely to result in successful standards creation and facilitate industry growth" [110]. Conversely, standardization is a requirement for mass adoption that is being developed. Given the multiple blockchain interoperability solutions, both Hybrid Connectors, and Blockchain of Blockchains, some of them with considerable weight on the industry, we believe

this is a very likely scenario. In Section 6.3, we expose multiple use-cases that may benefit from cross-blockchain technology, which can foster adoption by enthusiasts and enterprises. In conclusion, we envision reliable solutions and standards emerging in the following years and a steady increase in blockchain adoption by enterprises and individuals alike.

As a value enhancer and maturing key factor, interoperability will ultimately decide the survival of this technology. Based on the evidence collected and analyzed, we foresee increased attention to this research area, with blockchain interoperability gaining traction among the academia and the industry.

6.5 Open Issues and Challenges

In this section, we present open issues and challenges regarding blockchain interoperability and, in a more general sense, the adoption of blockchain.

Nowadays, solutions available to build decentralized applications lack interoperability, thwarting scalability [29]. As Liu et al. note, "it is very challenging to enforce correct executions in a full trust-free manner where no trusted authority is allowed to coordinate the executions on different blockchains" [143]. Although interesting and notorious recent advances in this area make interoperability a reality, there is still a gap between theory and practice, as much of the existing work is mostly conceptual.

Given the vast amount of blockchain platforms, fragmentation regarding solutions and their approach to interoperability is strongly present, for example, in IoT scenarios [239]. A combination of multiple platforms tailored for specific purposes, which can be public, private, or consortium, adds an overhead to manage workflows. In particular, this concern is intensified when multiple blockchains are serving a specific application.

Concerning blockchain scalability, the internet of blockchains can be realized upon improvements to current performance, both in public and private blockchains. Techniques such as implicit consensus and data sharding can improve transaction throughput and storage [128]. However, blockchain sharding requires solving cross-blockchain transaction routing and retrieval and asset referencing (also known as the discoverability problem).

It is challenging to coordinate transactions from different blockchains to support a cross-chain dApp, as different blockchains have different properties (e.g., architecture, protocols [1], service discovery, access control, between others). In particular, reverting a transaction that depended on another can be cumbersome, especially given different transaction finalities from different blockchains). Some solutions have proposed a mechanism to overcome such a challenge (blockchain of blockchains) [143, 215]. Although a promising approach, it is still unclear the applicability of these solutions to arbitrarily complex cross-blockchain dApp. More research is required to confirm the feasibility of this approach.

Some authors [206] highlight problems related to the GDPR, such as security, trust, confidentiality, and data privacy issues. In particular, security threats are exacerbated by the presence of multiple blockchains and possible multiple administrators. Regarding privacy, the authors underline problems with the right-to-forget, in which a user can ask his or her data to be deleted from the blockchain. Currently, most blockchains do not provide effective mechanisms that can respond to this request. Blockchain fine-grain access control is appointed as a key requirement to minimize information leakage and confidentiality risk.

Blockchain interoperability reduces dependencies on a single blockchain, and consequently, risk (e.g., the blockchain is attacked) [45], it does not eliminate the inherent risks. It is worth underscoring that the multiple blockchain approach is more complicated than the sum of its parts, as there is extra complexity underlying the cross-chain communication.

This adds challenges to governance: whereas a private consortia can use Hybrid Connectors at will to interoperate systems (decentralized and/or decentralized), the governance model is not straightforward within community projects, supported by public blockchains.

In short, the most relevant open issues towards blockchain interoperability are:

- The gap between theory and practice, including the lack of standardization and implementations [101, 239],
- Discoverability [1, 143, 215],
- Privacy and Security [206, 209, 227, 233],
- Governance [103, 104, 175, 224].

Notwithstanding, security [138, 178], privacy [55], and scalability (e.g., using sharding [232] or novel blockchain systems [162]) remain the most prominent areas to be improved in the blockchain space.

7 RESEARCH DIRECTIONS

New tools, frameworks, standard proposals, and even programming models are emerging and need further development. Programming models such as Polkadot and Cosmos offer developers a way to create their blockchains effectively and connect them to other blockchains. Protocols such as ILP and UIP allow cross-blockchain transactions. Programming languages such as HSL and DAML aim at embedding different blockchain models, providing an abstraction for cross-blockchain dApps.

Although one can have good reasons to utilize blockchain interoperability solutions for public or private blockchains, few solutions are available for connecting them. The problem of obtaining state from permissioned blockchains effectively [2] makes interoperating with private blockchains a challenge [114, 224]. Thus, connecting public and private blockchains bidirectionally remains an open problem.

One of the problems that bidirectional communication across permissioned and permissionless ledgers poses is semantic compatibility. Technical interoperability does provide the technical foundation that realizes interoperability but does not grant semantic interoperability per se [103]. There is, therefore, a gap: how can we effectively combine both blockchain types to enable new use cases? How to make sure a solution complies with the goals of all involved stakeholders? Disciplines as view integration can help to provide an answer [21]. View integration is the process that combines views of the same business process into a consolidated one by combining the different views of the stakeholders participating in different blockchains.

Another considerable obstacle for blockchain adoption is its fast-paced development. The development of blockchain interoperability standards may provide a way for more flexibility regarding backward compatibility.

In the light of the present study and the identified open issues and challenges, we propose research directions based on some sections of our survey: research on architecture for enabling blockchain interoperability, Public Connectors, Blockchain of Blockchains, Hybrid Connectors, and supporting technologies, standards, use cases, and others.

Architecture for Blockchain Interoperability (Section B):

- Define a blockchain interoperability maturity model, modeling interoperability at its various layers (e.g., technological, semantic, organizational).
- Model the different views on the various types of interoperability, according to different stakeholders (e.g., the provider's technical view on a cross-blockchain dApp *versus* the semantic view of the end-user on the same cross-blockchain dApp).
- Study blockchain interoperability semantics by exploring, for example, the research area of view integration [59].

Public Connectors (Section 5.1):

- Research on how permissioned blockchains can benefit from sidechains to improve scalability and privacy.
- Develop protocols to allow fiat money exchange, higher liquidity on decentralized exchanges. Conversely, improve the level of privacy and security of centralized exchanges.

Blockchain of Blockchains (Section 5.2):

- Integration of existing blockchain systems with Blockchain of Blockchains.
- Study how Blockchain of Blockchains can provide a reliable interoperability scheme bridging permissioned blockchains and permissionless blockchains.
- Connect Blockchain of Blockchains to both centralized systems and decentralized ledger systems (e.g., connect Polkadot to Visa).

Hybrid Connectors (Section 5.3):

- Decentralize the trust of trusted relays by integrating them with public blockchains (e.g., by submitting the state periodically to a public blockchain);
- Study how blockchain-agnostic protocols can be easily adapted to existing ledgers.
- Explore the blockchain of blockchains approach as an advance in dependable blockchain-based applications.
- Improve atomicity and consistency guarantees on cross-blockchain decentralized applications.
- Explore blockchain migration across public and permissioned ledgers. Such migration schemes can be decentralized and adapt to functional and non-functional requirements imposed by stakeholders.
- Explore blockchain migration via non-trusted relays (e.g., using a set of public escrow nodes following a protocol).
- Develop frameworks for multiple blockchain management. Such frameworks should respond to multiple stakeholder needs, decentralizing trust.
- Model integration abstraction layers that enable the development of universally connected contracts.
- Research on the visualization of CC-Txs.

Supporting technologies and standards, use cases, and others (Section 6.1):

- Work along with regulators and standardizing bodies to come with blockchain interoperability standards across industries
- Research on blockchain interoperability programming languages, supporting tools, and standards, including but not limited to cross-blockchain programming languages and frameworks, decentralized identifiers and verifiable credentials, and blockchain interoperability standards for enterprise blockchains;
- Explore new use cases using multiple blockchains, the "value-level" interoperability [155].
- Research synergies between cryptocurrency-based interoperability approaches, Blockchain of Blockchains, and Hybrid Connectors.
- Study security aspects of blockchain interoperability.
- Understand the implications of the different interoperability layers (value, semantic, organizational, among others).

8 CONCLUSION

In this paper, we performed a systematic literature review on blockchain interoperability. We systematically analyzed, compared, and discussed 80 documents, corresponding to 45 blockchain interoperability solutions. By including grey

literature, we expect to thwart intrinsic limitations regarding the blockchain interoperability research area, such as a considerable presence of the industry. By exploring each solution methodologically, this study provides interesting insights, distributed across three categories: Public Connectors, Blockchain of Blockchains, and Hybrid Connectors. Despite sidechain and HLTC solutions are gaining traction in the industry, blockchain interoperability are not solely Public Connectors solutions. New approaches started emerging since 2017. Hybrid Connectors provide a varied landscape of solutions, adapted for the majority of the use cases. They are likely to be used to produce cross-blockchain dApps. Blockchain of Blockchains are likely to be adopted by the industry in the short-medium term, by leveraging easy-to-produce, customizable blockchains.

Our findings allow us to conclude that conditions to research on blockchain interoperability are fulfilled, allowing a multitude of new use cases. Thus, we expect interest in this research area to raise considerably.

This work is towards making the blockchain ecosystem more practical, by easing work for developers and researchers. We expect that this study provides a robust and dependable starting point whereby developers and researchers can work in the blockchain interoperability research area.

ACKNOWLEDGMENTS

The authors would like to thank to the anonymous reviewers that constructively provided suggestions that significantly improved this paper. Thanks to Peter Somogyvari, Paul DiMarzio, Jag Sidhu, Sergio Lerner, Andy Leung, Travis Walker, Bill Laboon, Josh Lee, Austin King, Oliver Birch, Thomas Hardjono, and Miguel Matos for fruitful discussions regarding blockchain interoperability. We thank Daniel Hardman and Ken Elbert for constructive discussions about DIDs and verifiable credentials. Special thanks go to Iulia Mihaiu, Cláudio Correia, Benedikt Putz, Francisco Braga, Gavin Wood, João Ferreira, Miguel Pardal, Jonas Gehrlein, and Dilum Bandara for constructive comments and suggestions that greatly contributed to improving the paper. The authors express their gratitude to the Linux Foundation for providing the Diversity & Inclusion scholarship. This work was partially supported by the EC through project 822404 (QualiChain), and by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UIDB/50021/2020 (INESC-ID).

REFERENCES

[1] Ermyas Abebe, Dushyant Behl, Chander Govindarajan, Yining Hu, Dileban Karunamoorthy, Petr Novotny, Vinayaka Pandit, Venkatraman Ramakrishna, and Christian Vecchiola. 2019. Enabling Enterprise Blockchain Interoperability with Trusted Data Transfer. In *Proceedings of the 20th International Middleware Conference Industrial Track*. Association for Computing Machinery, 29–35.

[2] Ermyas Abebe, Dileban Karunamoorthy, Jiangshan Yu, Yining Hu, Vinayaka Pandit, Allison Irvin, and Venkatraman Ramakrishna. 2021. Verifiable Observation of Permissioned Ledgers. *arXiv* (2021).

[3] Abhishta Abhishta, Reinoud Joosten, Sergey Dragomiretskiy, and Lambert J.M. Nieuwenhuis. 2019. Impact of Successful DDoS Attacks on a Major Crypto-Currency Exchange. In *Proceedings - 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing*. Institute of Electrical and Electronics Engineers Inc., 379–384.

[4] Hayden Adams, Noah Zinsmeister, and Dan Robinson. 2020. *Uniswap v2 Core*. Technical Report.

[5] John Adler, Ryan Berryhill, Andreas Veneris, Zissis Poulos, Neil Veira, and Anastasia Kastania. 2018. ASTRAEA: A Decentralized Blockchain Oracle. *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (2018), 1349–1354.

[6] Enterprise Ethereum Alliance. 2021. *Enterprise Ethereum Alliance Client Specification v7*. Technical Report. Ethereum. https://entethalliance.github.io/client-spec/spec.html

[7] Mohammad Amiri, Divyakant Agrawal, and Mohammad Amr El Abbadi. 2019. CAPER: A Cross-Application Permissioned Blockchain. In *International Conference on Very Large Data Bases*, Vol. 12. 1385–1398.

[8] Emmanuelle Anceaume, Antonella Del Pozzo, Romaric Ludinard, Maria Potop-Butucaru, and Sara Tucci-Piergiovanni. 2018. Blockchain Abstract Data Type. http://arxiv.org/abs/1802.09877

[9] Elli Androulaki, Artem Barger, Vita Bortnikov, Srinivasan Muralidharan, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Chet Murthy, Christopher Ferris, Gennady Laventman, Yacov Manevich, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti,

Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick. 2018. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In *Proceedings of the 13th EuroSys Conference, EuroSys 2018*, Vol. 2018-Janua. Association for Computing Machinery, Inc, New York, New York, USA, 1–15.

[10] Andreas M Antonopoulos and Gavin Wood. 2018. *Mastering {Ethereum}: building smart contracts and dapps*. O'Reilly Media.

[11] V Arasev. 2017. *POA Network Whitepaper*. Technical Report. POA Network. https://www.poa.network/for-users/whitepaper

[12] ARK. 2019. ARK Whitepaper Version 2.1.0. https://whitepaper.ark.io/prologue

[13] N. Asokan, Victor Shoup, and Michael Waidner. 1998. Optimistic fair exchange of digital signatures. In *International Conference on the Theory and Applications of Cryptographic Techniques*, Vol. 1403. Springer Verlag, 591–606.

[14] Nicola Atzei, Massimo Bartoletti, and Tiziana Cimoli. 2017. A Survey of Attacks on Ethereum Smart Contracts. In *International Conference on Principles of Security and Trust*. Association for Computing Machinery, 164–186.

[15] Adam Back, Matt Corallo, Luke Dashjr, Mark Friedenbach, Gregory Maxwell, Andrew Miller, Andrew Poelstra, Jorge Timón, and Pieter Wuille. 2014. *Enabling Blockchain Innovations with Pegged Sidechains*. Technical Report. Blockstream.

[16] HMN Dilum Bandara, Xiwei Xu, and Ingo Weber. 2019. Patterns for Blockchain Data Migration. (6 2019). http://arxiv.org/abs/1906.00239

[17] Tal Baneth. 2019. Waterloo — a Decentralized Practical Bridge between EOS and Ethereum. https://blog.kyber.network/waterloo-a-decentralized-practical-bridge-between-eos-and-ethereum-1c230ac65524

[18] Richard Barnes. 2020. Factors in the Portability of Tokenized Assets on Distributed Ledgers. *arXiv pre-prints* (5 2020). http://arxiv.org/abs/2005.07461

[19] Rafael Belchior, Miguel Correia, and Thomas Hardjono. 2021. *DLT Gateway Crash Recovery Mechanism (draft-belchior-gateway-recovery-00)*. Technical Report. IETF. https://datatracker.ietf.org/doc/draft-belchior-gateway-recovery/

[20] Rafael Belchior, Miguel Correia, and André Vasconcelos. 2019. JusticeChain: Using Blockchain To Protect Justice Logs. In *27th International Conference on Cooperative Information Systems*. Springer, Cham.

[21] Rafael Belchior, Sérgio Guerreiro, André Vasconcelos, and Miguel Correia. 2020. A Survey on Business Process View Integration. (11 2020). http://arxiv.org/abs/2011.14465

[22] Rafael Belchior, Benedikt Putz, Guenther Pernul, Miguel Correia, André Vasconcelos, and Sérgio Guerreiro. 2020. SSIBAC : Self-Sovereign Identity Based Access Control. In *The 3rd International Workshop on Blockchain Systems and Applications*. IEEE.

[23] Rafael Belchior, André Vasconcelos, and André Correia. 2021. BUNGEE: Visualizing, Merging, and Processing Blockchain Views. *to appear* (2021).

[24] Rafael Belchior, André Vasconcelos, and Miguel Correia. 2020. Towards Secure, Decentralized, and Automatic Audits with Blockchain. In *European Conference on Information Systems*. Association for Information Systems.

[25] Rafael Belchior, André Vasconcelos, Miguel Correia, and Thomas Hardjono. 2021. HERMES: Fault-Tolerant Middleware for Blockchain Interoperability. *TechrXiv 14120291/1* (3 2021). https://doi.org/10.36227/TECHRXIV.14120291.V1

[26] Marianna Belotti, Stefano Moretti, Maria Potop-Butucaru, and Stefano Secci. 2020. *Game Theoretical Analysis of Atomic Cross-Chain Swaps*. Technical Report. https://hal.archives-ouvertes.fr/hal-02414356

[27] Eli Ben-Sasson Technion Alessandro Chiesa, Eran Tromer, and Madars Virza MIT. 2014. Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture. In *USENIX Security*. USENIX Association.

[28] Juan Benet. 2014. *IPFS - Content Addressed, Versioned, P2P File System*. Technical Report Draft 3. IPFS.

[29] L Besançon, Catarina Silva, and Parisa Ghodous. 2019. Towards Blockchain Interoperability: Improving Video Games Data Exchange. In *2019 IEEE International Conference on Blockchain and Cryptocurrency*.

[30] Alysson Bessani, Miguel Correia, Bruno Quaresma, Fernando André, and Paulo Sousa. 2011. DepSky: Dependable and secure storage in a cloud-of-clouds. *ACM Transactions on Storage* 9, 4 (2011), 31–45.

[31] Binance. 2020. *Binance Smart Chain: A Parallel Binance Chain to Enable Smart Contracts version 0.1*. Technical Report. Binance.

[32] Garrett Birkhoff. 1940. *Lattice Theory, Volume 25, Part 2*. American Mathematical Soc. 418 pages. https://books.google.com/books?id=0Y8d-MdtVwkC&pgis=1

[33] Monika Bishnoi and Rajesh Bhatia. 2020. Interoperability solutions for blockchain. *Proceedings of the International Conference on Smart Technologies in Computing, Electrical and Electronics, ICSTCEE 2020* (2020), 381–385.

[34] Bitcoin Wiki. 2016. Hash Time Locked Contracts. https://en.bitcoin.it/wiki/Hash_Time_Locked_Contracts

[35] Bitcoin Wiki. 2016. Hashlock. https://en.bitcoin.it/wiki/Hashlock

[36] Bitcoin Wiki. 2016. Timelock. https://en.bitcoin.it/wiki/Timelock

[37] Bitcoin Wiki. 2017. Simplified Payment Verification (SPV). https://en.bitcoinwiki.org/wiki/Simplified_Payment_Verification

[38] Matthew Black, Tingwei Liu, and Tony Cai. 2019. Atomic Loans: Cryptocurrency Debt Instruments. http://arxiv.org/abs/1901.05117

[39] Blocknet. 2019. Blocknet Documentation. https://docs.blocknet.co/#technical-overview

[40] Blocknet. 2019. Blocknet Protocol - XBridge Asset Compatibility. https://docs.blocknet.co/protocol/xbridge/compatibility/

[41] Michael Borkowski, Philipp Frauenthaler, Marten Sigwart, Taneli Hukkinen, Oskar Hladky, and Stefan Schulte. 2019. Cross-Blockchain Technologies: Review, State of the Art, and Outlook. https://dsg.tuwien.ac.at/projects/tast/pub/tast-white-paper-4.pdf

[42] Michael Borkowski, Daniel Mcdonald, Christoph Ritzer, and Stefan Schulte. 2018. Towards Atomic Cross-Chain Token Transfers: State of the Art and Open Questions within TAST. , 10 pages. http://www.infosys.tuwien.ac.at/tast/

[43] Michael Borkowski, Christoph Ritzer, Daniel Mcdonald, and Stefan Schulte. 2018. Caught in Chains: Claim-First Transactions for Cross-Blockchain Asset Transfers. http://www.infosys.tuwien.ac.at/tast/

[44] Michael Borkowski, Christoph Ritzer, and Stefan Schulte. 2018. Deterministic Witnesses for Claim-First Transactions. https://dsg.tuwien.ac.at/projects/tast/pub/

[45] Michael Borkowski, Marten Sigwart, Philipp Frauenthaler, Taneli Hukkinen, and Stefan Schulte. 2019. DeXTT: Deterministic Cross-Blockchain Token Transfers. *IEEE Access* 7 (8 2019), 111030–111042.

[46] Pearl Brereton, Barbara A. Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. 2007. Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software* 80, 4 (4 2007), 571–583. https://doi.org/10.1016/j.jss.2006.07.009

[47] Richard Brown. 2018. The Corda Platform: An Introduction White Paper. https://www.r3.com/reports/the-corda-platform-an-introduction-whitepaper/

[48] Gewu Bu, Riane Haouara, Thanh Son Lam Nguyen, and Maria Potop-Butucaru. 2020. Cross hyperledger fabric transactions. In *CRYBLOCK 2020 - Proceedings of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems, Part of MobiCom 2020*. Association for Computing Machinery, 35–40. https://doi.org/10.1145/3410699.3413796

[49] Jeff Burdges, Alfonso Cevallos, Peter Czaban, Rob Habermeier, Syed Hosseini, Fabio Lama, Handan Kılınç Alper, Ximin Luo, Fatemeh Shirazi, Alistair Stewart, and Gavin Wood. 2020. Overview of polkadot and its design considerations. *arXiv 2005.13456* (2020), 1–40.

[50] Daniel Burnett and Matt Stone. 2017. *Verifiable Credentials Working Group*. Technical Report. W3C. https://www.w3.org/2017/vc/WG/

[51] Vitalik Buterin. 2014. Ethereum: A next-generation smart contract and decentralized application platform. https://github.com/ethereum/wiki/wiki/White-Paper

[52] Vitalik Buterin. 2016. *R3 Report - Chain Interoperability*. Technical Report. R3 Corda. https://www.r3.com/wp-content/uploads/2017/06/chain_interoperability_r3.pdf

[53] Vitalik Buterin. 2021. An Incomplete Guide to Rollups. https://vitalik.ca/general/2021/01/05/rollup.html

[54] Christian Cachin and Marko Vukolić. 2017. Blockchain Consensus Protocols in the Wild. *arXiv e-prints* 91 (7 2017). http://arxiv.org/abs/1707.01873

[55] Fran Casino, Thomas Dasaklis, and Constantinos Patsakis. 2019. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telematics and Informatics* 36 (3 2019), 55–81.

[56] Jing Chen and Silvio Micali. 2016. Algorand. (7 2016), 51–68. http://arxiv.org/abs/1607.01341

[57] Mihai Christodorescu, Catherine Gu, Ranjit Kumaresan, Mohsen Minaei, Mustafa Ozdayi, Benjamin Price, Srinivasan Raghuraman, Muhammad Saad, Cuy Sheeeld, Minghua Xu, and Mahdi Zamani. 2020. Towards a Two-Tier Hierarchical Infrastructure: An OOine Payment System for Central Bank Digital Currencies. *arXiv* (2020).

[58] Clearmatics. 2018. Ion Interoperability Framework v2. https://github.com/clearmatics/ion

[59] João Colaço and Pedro Sousa. 2017. View integration of business process models. In *Lecture Notes in Business Information Processing*, Vol. 299. Springer Verlag, 619–632.

[60] COMIT. 2020. COMIT Protocol. DisincentivizingDoubleSpendAttacksAcrossInteroperableBlockchains.

[61] Mauro Conti, Kumar E. Sandeep, Chhagan Lal, and Sushmita Ruj. 2018. A survey on security and privacy issues of bitcoin. *IEEE Communications Surveys and Tutorials* 20, 4 (10 2018), 3416–3452.

[62] Miguel Correia. 2013. Clouds-of-Clouds for Dependability and Security: Geo-replication Meets the Cloud. In *Euro-Par 2013: Parallel Processing Workshops*. Springer Berlin Heidelberg, Berlin, Heidelberg, 95–104.

[63] Miguel Correia. 2019. From Byzantine Consensus to Blockchain Consensus. *Essentials of Blockchain Technology* (2019), 41.

[64] Ludovic Courcelas, Tom Lyons, and Ken Timsit. 2020. *European Union Blockchain Observatory and Forum 2018-2020 Conclusions and Reflections*. Technical Report. European Union Blockchain Observatory and Forum. https://www.eublockchainforum.eu/sites/default/files/reports/report_conclusion_book_v1.0.pdf?width=1024&height=800&iframe=true

[65] Arylyn Culwick and Dan Metcalf. 2018. *Blocknet design specification v1.0*. Technical Report. Blocknet. https://www.blocknet.co/wp-content/uploads/whitepaper/Blocknet_Whitepaper.pdf

[66] Bingrong Dai, Shengming Jiang, Menglu Zhu, Ming Lu, Dunwei Li, and Chao Li. 2020. Research and Implementation of Cross-Chain Transaction Model Based on Improved Hash-Locking. In *Communications in Computer and Information Science*, Vol. 1267. Springer Science and Business Media Deutschland GmbH, 218–230. https://doi.org/10.1007/978-981-15-9213-3{_}17

[67] Ratul Antik Das, Md Muhaimin Shah Pahalovi, and Muhammad Nur Yanhaona. 2019. Transaction finality through ledger checkpoints. In *Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS*, Vol. 2019-Decem. IEEE Computer Society, 183–192.

[68] Christian Decker and Roger Wattenhofer. 2015. A fast and scalable payment network with bitcoin duplex micropayment channels. In *Lecture Notes in Computer Science*, Vol. 9212. Springer Verlag, 3–18.

[69] Liping Deng, Huan Chen, Jing Zeng, and Liang Jie Zhang. 2018. Research on cross-chain technology based on sidechain and hash-locking. In *International Conference on Edge Computing*, Vol. 10973 LNCS. Springer Verlag, 144–151.

[70] Apoorvaa Deshpande and Maurice Herlihy. 2020. Privacy-preserving cross-chain atomic swaps. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 12063 LNCS. Springer, 540–549. https://doi.org/10.1007/978-3-030-54455-3{_}38

[71] Advait Deshpande, Katherine Stewart, Louise Lepetit, and Salil Gunashekar. 2017. *Distributed Ledger Technologies/Blockchain: Challenges, opportunities and the prospects for standards*. Technical Report. British Standards Institution.

[72] Digital Asset. 2019. DAML SDK 1.1.1 documentation. https://docs.daml.com/getting-started/installation.html

[73] Digital Asset. 2020. Canton : A Private , Scalable , and Composable Smart Contract Platform. (2020), 1–15.

[74] Johnny Dilley, Andrew Poelstra, Jonathan Wilkins, Marta Piekarska, Ben Gorlick, and Mark Friedenbach. 2016. *Strong Federations: An Interoperable Blockchain Solution to Centralized Third-Party Risks.* Technical Report. BlockStream. http://arxiv.org/abs/1612.05491

[75] Thomas Durieux, Joao F. Ferreira, Rui Abreu, and Pedro Cruz. 2020. Empirical review of automated analysis tools on 47,587 ethereum smart contracts. In *Proceedings - International Conference on Software Engineering*. IEEE Computer Society, 530–541.

[76] Ben Edgington. 2020. What's New in Ethereum 2. https://notes.ethereum.org/@ChihChengLiang/Sk8Zs--CQ/https%3A%2F%2Fhackmd.io%2F%40benjaminion%2Fwnie2_200320?type=book

[77] Ethereum Foundation. 2019. ETH 2 Phase 2 WIKI. https://hackmd.io/UzysWse1Th240HELswKqVA?view

[78] Ethereum Foundation. 2019. Ethereum 2.0 Specifications. https://github.com/ethereum/eth2.0-specs

[79] Ethereum Foundation and Consensys. 2015. BTC-relay: Ethereum contract for Bitcoin SPV. https://github.com/ethereum/btcrelay

[80] EthHub. 2020. Ethereum 2.0 Phases - EthHub. https://docs.ethhub.io/ethereum-roadmap/ethereum-2.0/eth-2.0-phases/#eth2-the-new-ether

[81] EU Blockchain. 2020. Scalability, interoperability and sustainability of blockchains. https://www.eublockchainforum.eu/reports

[82] Jody Condit Fagan. 2017. An Evidence-Based Review of Academic Web Search Engines, 2014-2016: Implications for Librarians' Practice and Research Agenda. *Information Technology and Libraries* 36, 2 (6 2017), 7–47. https://ejournals.bc.edu/index.php/ital/article/view/9718

[83] Ghareeb Falazi, Uwe Breitenbücher, Florian Daniel, Andrea Lamparelli, Frank Leymann, and Vladimir Yussupov. 2020. Smart Contract Invocation Protocol (SCIP): A Protocol for the Uniform Integration of Heterogeneous Blockchain Smart Contracts. In *International Conference on Advanced Information Systems Engineering*, Vol. 12127 LNCS. 134–149.

[84] Ghareeb Falazi, Andrea Lamparelli, Uwe Breitenbuecher, Florian Daniel, and Frank Leymann. 2020. Unified Integration of Smart Contracts through Service Orientation. *IEEE Software* 37, 5 (2020). https://doi.org/10.1109/MS.2020.2994040

[85] Wanchain Foundation. 2019. Wanchain Roadmap. https://www.wanchain.org/learn/

[86] Philipp Frauenthaler, Michael Borkowski, and Stefan Schulte. 2019. A Framework for Blockchain Interoperability and Runtime Selection. *arXiv preprint* (5 2019). https://arxiv.org/abs/1905.07014

[87] Philipp Frauenthaler, Marten Sigwart, Michael Borkowski, Taneli Hukkinen, and Stefan Schulte. 2019. *Towards Efficient Cross-Blockchain Token Transfers.* Technical Report. http://www.infosys.tuwien.ac.at/tast/

[88] Philipp Frauenthaler, Marten Sigwart, Christof Spanring, and Stefan Schulte. 2020. *Leveraging Blockchain Relays for Cross-Chain Token Transfers.* Technical Report.

[89] Philipp Frauenthaler, Marten Sigwart, Christof Spanring, and Stefan Schulte. 2020. Testimonium : A Cost-Efficient Blockchain Relay. *arXiv Preprints* (2020).

[90] Philipp Frauenthaler, Marten Sigwart, Christof Spanring, Michael Sober, and Stefan Schulte. 2020. ETH Relay: A Cost-efficient Relay for Ethereum-based Blockchains. In *2020 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 204–213. https://doi.org/10.1109/Blockchain50366.2020.00032

[91] Fusion Foundation. 2017. *An Inclusive Cryptofinance Platform Based on Blockchain.* Technical Report. Fusion Foundation.

[92] Enrique Fynn, Pedone, Fernando, and Bessani Alysson. 2020. Smart Contracts on the Move. In *Dependable Systems and Networks*.

[93] Alberto Garoffolo, Dmytro Kaidalov, and Roman Oliynykov. 2020. *Zendoo: a zk-SNARK Verifiable Cross-Chain Transfer Protocol Enabling Decoupled and Decentralized Sidechains.* Technical Report. V.N.Karazin Kharkiv National University.

[94] Peter Gaži, Aggelos Kiayias, and Dionysis Zindros. 2019. Proof-of-stake sidechains. *IEEE Symposium on Security and Privacy* (2019).

[95] Sara Ghaemi, Sara Rouhani, Rafael Belchior, Rui S. Cruz, Hamzeh Khazaei, and Petr Musilek. 2021. A Pub-Sub Architecture to Promote Blockchain Interoperability. (1 2021). http://arxiv.org/abs/2101.12331

[96] Christian Gorenflo. 2020. *Towards a New Generation of Permissioned Blockchain Systems.* Ph.D. Dissertation. University of Waterloo. https://uwspace.uwaterloo.ca/bitstream/handle/10012/15860/Gorenflo_Christian.pdf?sequence=3

[97] Christian Gorenflo, Stephen Lee, Lukasz Golab, and Srinivasan Keshav. 2019. FastFabric: Scaling Hyperledger Fabric to 20,000 Transactions per Second. *ICBC 2019 - IEEE International Conference on Blockchain and Cryptocurrency* (2019), 455–463.

[98] Gideon Greenspan. 2015. MultiChain White Paper. https://www.multichain.com/white-paper/

[99] Lewis Gudgeon, Pedro Moreno-Sanchez, Stefanie Roos, Patrick Mccorry, and Arthur Gervais. 2020. SoK: Layer-Two Blockchain Protocols. In *International Conference on Financial Cryptography and Data Security*.

[100] Joel Gugger. 2020. Bitcoin-Monero Cross-chain Atomic Swap. *Cryptology ePrint Archive* (2020). https://eprint.iacr.org/2020/1126

[101] Thomas Hardjono. 2021. Blockchain Gateways, Bridges and Delegated Hash-Locks. *arXiv* (2 2021). http://arxiv.org/abs/2102.03933

[102] Thomas Hardjono, Martin Hargreaves, and Ned Smith. 2020. An Interoperability Architecture for Blockchain Gateways. https://datatracker.ietf.org/doc/draft-hardjono-blockchain-interop-arch/

[103] Thomas Hardjono, Alexander Lipton, and Alex Pentland. 2019. Toward an interoperability architecture for blockchain autonomous systems. *IEEE Transactions on Engineering Management* (2019).

[104] Thomas Hardjono, Alexander Lipton, and Alex Pentland. 2019. Towards a Public Key Management Framework for Virtual Assets and Virtual Asset Service Providers. http://arxiv.org/abs/1909.08607

[105] Martin Hargreaves and Thomas Hardjono. 2020. Open Digital Asset Protocol (draft-hargreaves-odap-01). https://datatracker.ietf.org/doc/draft-hargreaves-odap/

[106] Timo Hegnauer Zürich, Eder Scheid, Bruno Rodrigues, Timo Hegnauer, Eder Scheid, and Bruno Rodrigues. 2019. *Design and Development of a Blockchain Interoperability API.* Ph.D. Dissertation. University of Zürich. http://www.csg.uzh.ch/

[107] Maurice Herlihy. 2018. Atomic cross-chain swaps. In *Proceedings of the Annual ACM Symposium on Principles of Distributed Computing*. Association for Computing Machinery, New York, New York, USA, 245–254.

[108] Maurice Herlihy. 2018. Atomic Cross-Chain Swaps. (2018). https://doi.org/10.1145/3212734.3212736

[109] Garrick Hileman and Michel Rauchs. 2017. Global Blockchain Benchmarking Study. *SSRN Electronic Journal* (4 2017).

[110] David Hyland-Wood and Shahan Khatchadourian. 2018. A Future History of International Blockchain Standards. *The Journal of the British Blockchain Association* 1, 1 (2018), 1–10.

[111] Hyperledger. 2015. Docs | Hyperledger Sawtooth. https://sawtooth.hyperledger.org/docs/

[112] Hyperledger. 2019. Hyperledger Quilt Documentation. https://wiki.hyperledger.org/display/quilt/Hyperledger+Quilt

[113] IBC Ecosystem Working Group. 2020. Inter-blockchain Communication Protocol (IBC). https://github.com/cosmos/ics/tree/master/ibc

[114] iiconsortium. 2020. *Distributed Ledgers in IIoT.* Technical Report. Industrial Internet Consortium. https://www.iiconsortium.org/pdf/Distributed_Ledgers_in_IIoT_White_Paper_2020-07-22.pdf

[115] Interledger. 2020. Interledger Protocol V4 (ILPv4) | Interledger. https://interledger.org/rfcs/0027-interledger-protocol-4/

[116] Arjun Jain and Patrick Schilz. 2017. *Block Collider Whitepaper.* Technical Report. https://www.blockcollider.org/whitepaper

[117] H Jin and X Dai. 2018. Towards a novel architecture for enabling interoperability amongst multiple blockchains. In *IEEE 38th International Conference on Distributed Computing Systems*.

[118] Sandra Johnson, Peter Robinson, and John Brainard. 2019. Sidechains and interoperability. *arXiv e-prints* (3 2019). http://arxiv.org/abs/1903.04077

[119] JP Morgan. 2017. Quorum White Paper. https://github.com/jpmorganchase/quorum/blob/master/docs/QuorumWhitepaperv0.2.pdf

[120] Luo Kan, Yu Wei, Amjad Hafiz Muhammad, Wang Siyuan, Gao Linchao, and Hu Kai. 2018. A Multiple Blockchains Architecture on Inter-Blockchain Communication. *Proceedings - 2018 IEEE 18th International Conference on Software Quality, Reliability, and Security Companion, QRS-C 2018* (2018), 139–145.

[121] Niclas Kannengießer, Michelle Pfister, Malte Greulich, Sebastian Lins, and Ali Sunyaev. 2020. Bridges Between Islands: Cross-Chain Technology for Distributed Ledger Technology. *Hawaii International Conference on System Sciences* (2020).

[122] Harleen Kaur, M. Afshar Alam, Roshan Jameel, Ashish Kumar Mourya, and Victor Chang. 2018. A Proposed Solution and Future Direction for Blockchain-Based Heterogeneous Medicare Data in Cloud Environment. *Journal of Medical Systems* 42, 8 (8 2018).

[123] Kiranbir Kaur, Sandeep Sharma, and Karanjeet Singh Kahlon. 2017. Interoperability and portability approaches in inter-connected clouds: A review. *Comput. Surveys* 50, 4 (2017).

[124] Rami Khalil, Pedro Moreno-Sanchez, Alexei Zamyatin, Arthur Gervais, and Guillaume Felley. [n.d.]. *Commit-Chains: Secure, Scalable Off-Chain Payments.* Technical Report. https://github.com/liquidity-network/nocust-contracts-solidity

[125] Aggelos Kiayias and Dionysis Zindros. 2018. *Proof-of-Work Sidechains.* Technical Report. IOHK.

[126] Barbara Kitchenham and Stuart Charters. 2007. *Guidelines for performing Systematic Literature Reviews in Software Engineering Version 2.3.* Technical Report. Keele University and University of Durham. https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf

[127] T. Koens and E. Poll. 2019. Assessing interoperability solutions for distributed ledgers. *Pervasive and Mobile Computing* 59 (2019), 101079.

[128] Eleftherios Kokoris-Kogias, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ewa Syta, and Bryan Ford. 2018. OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding. In *Proceedings - IEEE Symposium on Security and Privacy*, Vol. 2018-May. Institute of Electrical and Electronics Engineers Inc., 583–598.

[129] Komodo. 2018. *Komodo Whitepaper.* Technical Report. Komodo. https://komodoplatform.com/wp-content/uploads/2018/06/Komodo-Whitepaper-June-3.pdf

[130] Jae Kwon and Ethan Buchman. 2016. *Cosmos Whitepaper.* Technical Report. Cosmos Foundation.

[131] Kyber Network. 2018. Peace Relay. https://github.com/KyberNetwork/peace-relay

[132] KyberNetwork. 2018. Waterloo Bridge. https://github.com/KyberNetwork/bridge_eth_smart_contracts

[133] Pascal Lafourcade and Marius Lombard-Platet. 2020. About blockchain interoperability. *Inform. Process. Lett.* 161 (2020), 105976.

[134] Rongjian Lan, Ganesha Upadhyaya, Stephen Tse, and Mahdi Zamani. 2021. Horizon: A Gas-Efficient, Trustless Bridge for Cross-Chain Transactions. (1 2021). http://arxiv.org/abs/2101.06000

[135] Sergio Lerner. 2015. *RSK Whitepaper.* Technical Report. RSK. https://docs.rsk.co/RSK_White_Paper-Overview.pdf

[136] Dawei Li, Jianwei Liu, Zongxun Tang, Qianhong Wu, and Zhenyu Guan. 2019. AgentChain: A Decentralized Cross-Chain Exchange System. In *18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE).*

[137] Wenting Li, Alessandro Sforzin, Sergey Fedorov, and Ghassan O. Karame. 2017. Towards scalable and private industrial blockchains. In *ACM Workshop on Blockchain, Cryptocurrencies and Contracts*. Association for Computing Machinery, Inc, 9–14.

[138] Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo, and Qiaoyan Wen. 2020. A survey on the security of blockchain systems. *Future Generation Computer Systems* 107 (6 2020), 841–853.

[139] Alessandro Liberati, Douglas Altman, Jennifer Tetzlaff, Cynthia Mulrow, Peter Gøtzsche, John Ioannidis, Mike Clarke, Devereaux, Jos Kleijnen, and David Moher. 2009. *The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: Explanation and elaboration.* Technical Report 7.

[140] Libra Association. 2019. The Libra Blockchain. (2019). https://libra.org/en-US/wp-content/uploads/sites/23/2019/06/LibraWhitePaper_en_US.pdf
[141] Claudio Lima. 2018. Developing open and interoperable DLT/Blockchain Standards [Standards]. *IEEE Computer* 51, 11 (2018). https://doi.org/10.1109/MC.2018.2876184
[142] Zhuotao Liu, Yangxi Xiang, Jian Shi, Peng Gao, Haoyu Wang, Xusheng Xiao, Bihan Wen, and Yih-Chun Hu. 2019. HyperService. Association for Computing Machinery (ACM), 549–566. https://doi.org/10.1145/3319535.3355503
[143] Zhuotao Liu, Yangxi Xiang, Jian Shi, Peng Gao, Haoyu Wang, Xusheng Xiao, Bihan Wen, and Yih-Chun Hu. 2019. Hyperservice: Interoperability and programmability across heterogeneous blockchains. In *ACM SIGSAC Conference on Computer and Communications*.
[144] Loom. 2016. Intro to Loom Network | Loom SDK. https://loomx.io/developers/en/intro-to-loom.html
[145] Jack Lu, Boris Yang, Zane Liang, Ying Zhang, Shi Demmon, Eric Swartz, and Lizzie Lu. 2017. Wanchain: Building Super Financial Markets for the New Digital Economy. , 34 pages. https://wanchain.org/files/Wanchain-Whitepaper-EN-version.pdf
[146] Quenby Mahood, Dwayne Van Eerd, and Emma Irvin. 2014. Searching for grey literature for systematic reviews: challenges and benefits. *Research Synthesis Methods* 5, 3 (9 2014), 221–234.
[147] C Manjunath, Daniel Anderson, Thomas Barnes, Srinath Duraisamy, Manoj Gopalakrishnan, Karthika Murthy, Ramakrishna Srinivasamurthy, and Yevgeniy Yarmosh. 2019. Hyperledger Avalon. https://github.com/hyperledger/avalon/blob/master/docs/avalon-arch.pdf
[148] Likoebe M. Maruping, Viswanath Venkatesh, and Ritu Agarwal. 2009. A control theory perspective on agile methodology use and changing user requirements. *Information Systems Research* 20, 3 (9 2009), 377–399.
[149] Mayer Christoph, Mai Jesse N, and Tom M. 2017. *Tokrex Whitepaper.* Technical Report. Tokrex. www.tokrex.org
[150] Greg Medcraft. 2021. *Regulatory Approaches to the Tokenisation of Assets.* Technical Report. OECD. https://www.oecd.org/finance/regulatory-approaches-to-the-tokenisation-of-assets.htm
[151] Metronome. 2019. Metronome documentation, FAQ, July 2019. https://github.com/autonomoussoftware/documentation/blob/master/FAQ.md
[152] Metronome. 2019. Metronome documentation v0.99. https://github.com/autonomoussoftware/documentation/blob/master/owners_manual/owners_manual.md
[153] Paul V. Mockapetris and Kevin J. Dunlap. 1988. Development of the Domain Name System. In *Symposium Proceedings on Communications Architectures and Protocols, SIGCOMM 1988.* Association for Computing Machinery, Inc, New York, New York, USA, 123–133.
[154] Hart Montgomery, Hugo Borne-Pons, Jonathan Hamilton, Mic Bowman, Peter Somogyvari, Shingo Fujimoto, Takuma Takeuchi, and Tracy Kuhrt. 2020. Blockchain Integration Framework Whitepaper v 0.1. https://github.com/hyperledger/cactus/blob/master/docs/whitepaper/whitepaper.md
[155] Hart Montgomery, Hugo Borne-Pons, Jonathan Hamilton, Mic Bowman, Peter Somogyvari, Shingo Fujimoto, Takuma Takeuchi, Tracy Kuhrt, and Rafael Belchior. 2020. Hyperledger Cactus Whitepaper. https://github.com/hyperledger/cactus/blob/master/docs/whitepaper/whitepaper.md
[156] Roman Mühlberger, Stefan Bachhofner, Eduardo Castelló Ferrer, Claudio Di Ciccio, Ingo Weber, Maximilian Wöhrer, and Uwe Zdun. 2020. Foundational Oracle Patterns: Connecting Blockchain to the Off-Chain World. In *Lecture Notes in Business Information Processing*, Vol. 393 LNBIP. Springer Science and Business Media Deutschland GmbH, 35–51. https://doi.org/10.1007/978-3-030-58779-6{_}3
[157] Glenford Myers, Tom Badgett, and Corey Sandler. 2012. Test-Case Design. In *The Art of Software Testing*. John Wiley \& Sons, Ltd, Chapter 4, 41–84. https://doi.org/10.1002/9781119202486.ch4
[158] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. http://bitcoin.org/bitcoin.pdf
[159] National Interoperability Framework Observatory. 2020. European Interoperability Framework. https://joinup.ec.europa.eu/collection/nifo-national-interoperability-framework-observatory/3-interoperability-layers#3.6
[160] Cosmos Network. 2014. Cosmos Blog. https://blog.cosmos.network/
[161] Markus Nissl, Emanuel Sallinger, Stefan Schulte, and Michael Borkowski. 2020. Towards Cross-Blockchain Smart Contracts. (10 2020). http://arxiv.org/abs/2010.07352
[162] Henry C. Nunes, Roben C. Lunardi, Avelin F. Zorzo, Regio A. Michelin, and Salil S. Kanhere. 2020. Context-based smart contracts for appendable-block blockchains. In *IEEE International Conference on Blockchain and Cryptocurrency*.
[163] OASIS. 2010. Extensible Access Control Markup Language Version 3.0. https://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cd-03-en.html
[164] Yan Pang. 2020. A New Consensus Protocol for Blockchain Interoperability Architecture. *IEEE Access* 8 (2020), 153719–153730. https://doi.org/10.1109/ACCESS.2020.3017549
[165] Pantos Team. 2020. *Pantos Vision Paper.* Technical Report. Pantos.
[166] Joe Petrowski. 2020. Polkadot and Ethereum 2.0. https://wiki.polkadot.network/docs/en/learn-comparisons-ethereum-2
[167] Babu Pillai and Kamanashis Biswas. 2019. Blockchain Interoperable Digital Objects. In *ICBC2019 International Conference on Blockchain*. https://doi.org/10.1007/978-3-030-23404-1_6
[168] Babu Pillai, Kamanashis Biswas, and Vallipuram Muthukkumarasamy. 2020. Cross-chain interoperability among blockchain-based systems using transactions. *Knowledge Engineering Review* 35 (2020), 1–18. https://doi.org/10.1017/S0269888920000314
[169] Andrew Poelstra, Adam Back, Mark Friedenbach, Gregory Maxwell, and Pieter Wuille Blockstream. 2017. *Blockstream: Confidential Assets.* Technical Report. https://blockstream.com/bitcoin17-final41.pdf
[170] Polkadot. 2019. Cross-chain Message Passing (XCMP) · Polkadot Wiki. https://wiki.polkadot.network/docs/en/learn-crosschain
[171] Polkadot. 2019. Kusama Network. https://wiki.polkadot.network/docs/en/kusama-index
[172] Polkadot. 2019. Polkadot Consensus · Polkadot Wiki. https://wiki.polkadot.network/docs/en/learn-consensus
[173] Joseph Poon and Vitalik Buterin. 2017. *Plasma: Scalable Autonomous Smart Contracts.* Technical Report. Plasma. https://plasma.io/

[174] Joseph Poon and Thaddeus Dryja. 2016. *The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments.* Technical Report. Lightning Network. https://lightning.network/lightning-network-paper.pdf

[175] Ilham A. Qasse, Manar Abu Talib, and Qassim Nasir. 2019. Inter blockchain communication: A survey. In *Arab WIC 6th Annual International Conference Research Track*. Association for Computing Machinery.

[176] Minfeng Qi, Ziyuan Wang, Donghai Liu, Yang Xiang, Butian Huang, and Feng Zhou. 2020. ACCTP: Cross Chain Transaction Platform for High-Value Assets. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 12404 LNCS. Springer Science and Business Media Deutschland GmbH, 154–168. https://doi.org/10.1007/978-3-030-59638-5{_}11

[177] Quant Foundation. 2019. *Overledger Network Whitepaper v0.3.* Technical Report. Quant.

[178] Mayank Raikwar, Danilo Gligoroski, and Katina Kralevska. 2019. SoK of Used Cryptography in Blockchain. *IEEE Access* 7 (2019), 148550–148575.

[179] Drummond Reed, Manu Sporny, Markus Sabadello, Dave Longley, Christopher Allen, and Ryan Grant. 2018. *Decentralized Identifiers.* Technical Report. https://w3c.github.io/did-core/

[180] Marten Risius and Kai Spohrer. 2017. A Blockchain Research Framework. *Business and Information Systems Engineering* 59, 6 (12 2017), 385–409.

[181] Peter Robinson, David Hyland-Wood, Roberto Saltini, Sandra Johnson, and John Brainard. 2019. *Atomic Crosschain Transactions for Ethereum Private Sidechains.* Technical Report.

[182] Peter Robinson and Raghavendra Ramesh. 2020. General Purpose Atomic Crosschain Transactions. *arXiv* (11 2020). http://arxiv.org/abs/2011.12783

[183] Masoud Rostami, Farinaz Koushanfar, and Ramesh Karri. 2014. A primer on hardware security: Models, methods, and metrics. *Proc. IEEE* 102, 8 (2014), 1283–1295.

[184] Sara Rouhani, Rafael Belchior, Rui S Cruz, and Ralph Deters. 2021. Distributed Attribute-Based Access Control System Using a Permissioned Blockchain. *WWW* (2021).

[185] Frantz Rowe. 2014. What literature review is not: Diversity, boundaries and recommendations. *European Journal of Information Systems* 23, 3 (2014), 241–255.

[186] Janick Rueegger and Guilherme Sperb MacHado. 2020. Rational Exchange: Incentives in Atomic Cross Chain Swaps. In *IEEE International Conference on Blockchain and Cryptocurrency, ICBC 2020*. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ICBC48266.2020.9169408

[187] Kuheli Sai and David Tipper. 2019. Disincentivizing Double Spend Attacks Across Interoperable Blockchains. In *First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications*.

[188] Aetienne Sardon and Thomas Hardjono. 2020. Blockchain Gateways: Use-Cases (draft-sardon-blockchain-gateways-usecases-00). https://datatracker.ietf.org/doc/draft-sardon-blockchain-gateways-usecases/

[189] Eder Scheid, Bruno Rodrigues, and Burkhard Stiller. 2019. Toward a policy-based blockchain agnostic framework. *16th IFIP/IEEE International Symposium on Integrated Network Management* (2019).

[190] Eder J. Scheid, Timo Hegnauer, Bruno Rodrigues, and Burkhard Stiller. 2019. Bifröst: a Modular Blockchain Interoperability API. In *IEEE 44th Conference on Local Computer Networks*. Institute of Electrical and Electronics Engineers (IEEE), 332–339.

[191] Stefan Schulte, Marten Sigwart, Philipp Frauenthaler, and Michael Borkowski. 2019. Towards Blockchain Interoperability. In *International Conference on Business Process Management: BPM 2019: Business Process Management: Blockchain and Central and Eastern Europe Forum*, Vol. 361. Springer Verlag, 3–10. https://doi.org/10.1007/978-3-030-30429-4{_}1

[192] Security Token Standard. 2019. Security Token Standard - ERC 1400. https://thesecuritytokenstandard.org/

[193] Security Token Standard. 2019. SecurityTokenStandard/EIP-Spec. https://github.com/SecurityTokenStandard/EIP-Spec

[194] Narges Shadab, Farzin Hooshmand, and Mohsen Lesani. 2020. Cross-chain Transactions. In *IEEE International Conference on Blockchain and Cryptocurrency*.

[195] Omer Shlomovits and Oded Leiba. 2020. JugglingSwap: Scriptless Atomic Cross-Chain Swaps. *arXiv* (7 2020). http://arxiv.org/abs/2007.14423

[196] Jagdeep Sidhu, Eliot Scott, and Alexander Gabriel. 2018. *Z-DAG: An interactive DAG protocol for real-time crypto payments with Nakamoto consensus security parameters.* Technical Report. Syscoin. https://syscoin.org/zdag_syscoin_whitepaper.pdf

[197] Marten Sigwart, Philipp Frauenthaler, Taneli Hukkinen, and Stefan Schulte. 2019. *Towards Cross-Blockchain Transaction Verifications.* Technical Report. http://www.infosys.tuwien.ac.at/tast/

[198] Marten Sigwart, Philipp Frauenthaler, Christof Spanring, and Stefan Schulte. 2019. *Preparing Simplified Payment Verifications for Cross-Blockchain Token Transfers.* Technical Report. https://dsg.tuwien.ac.at/projects/tast/

[199] Amritraj Singh, Kelly Click, Reza M. Parizi, Qi Zhang, Ali Dehghantanha, and Kim Kwang Raymond Choo. 2020. Sidechain technologies in blockchain networks: An examination and state-of-the-art review. *Journal of Network and Computer Applications* 149 (2020).

[200] Vasilios A. Siris, Pekka Nikander, Spyros Voulgaris, Nikos Fotiou, Dmitrij Lagutin, and George C. Polyzos. 2019. Interledger Approaches. *IEEE Access* 7 (2019), 89948–89966.

[201] Matthew Spoke. 2017. *Aion: Enabling the decentralized Internet.* Technical Report. https://whitepaper.io/document/31/aion-whitepaper

[202] He Sun, Hongliang Mao, Xiaomin Bai, Zhidong Chen, Kai Hu, and Wei Yu. 2018. Multi-blockchain model for central bank digital currency. In *Parallel and Distributed Computing, Applications and Technologies, PDCAT Proceedings*, Vol. 2017-Decem. IEEE Computer Society, 360–367. https://doi.org/10.1109/PDCAT.2017.00066

[203] Swiss Financial Market Supervisory Authority. 2018. FINMA publishes ICO guidelines. https://www.finma.ch/en/news/2018/02/20180216-mm-ico-wegleitung/

[204] Nick Szabo. 1997. Formalizing and securing relationships on public networks. *First Monday* 2, 9 (1997).

[205] Michael Szydlo. 2004. Merkle Tree Traversal in Log Space and Time. In *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 541–554.

[206] H Tam Vo, Z Wang, D Karunamoorthy, J Wagner, E Abebe, and M Mohania. 2018. Internet of Blockchains: Techniques and Challenges Ahead. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 1574–1581.

[207] Paolo Tasca and Thayabaran Thanabalasingham. 2017. Taxonomy of Blockchain Technologies. Principles of Identification and Classification. *SSRN Electronic Journal* (6 2017).

[208] Tendermint. 2016. Tendermint BFT. https://github.com/tendermint/tendermint

[209] Stefan Thomas and Evan Schwartz. 2015. A Protocol for Interledger Payments. , 25 pages. https://interledger.org/interledger.pdf

[210] Hangyu Tian, Kaiping Xue, Shaohua Li, Jie Xu, Jianqing Liu, and Jun Zhao. 2020. Enabling Cross-chain Transactions: A Decentralized Cryptocurrency Exchange Protocol. *arXiv* (5 2020). http://arxiv.org/abs/2005.03199

[211] TO Group. 2016. *ArchiMate®3.0 Specification*. Van Haren Publishing.

[212] Token Taxonomy Consortium. 2019. *Token Specification Summary*. Technical Report. Token Taxonomy Initiative. https://tokentaxonomy.org/wp-content/uploads/2019/11/TTF-Overview.pdf

[213] Fakhar Ul Hassan, Anwaar Ali, Siddique Latif, Junaid Qadir, Salil Kanhere, Jatinder Singh, and Jon Crowcroft. 2019. Blockchain And The Future of the Internet: A Comprehensive Review. *arXiv e-prints* (2019). https://arxiv.org/abs/1904.00733

[214] U.S. Securities and Exchange Commission. 2019. *Framework for "Investment Contract" Analysis of Digital Assets 1*. Technical Report. https://www.sec.gov/files/dlt-framework.pdf

[215] Gilbert Verdian, Paolo Tasca, Colin Paterson, and Gaetano Mondelli. 2018. *Quant Overledger Whitepaper v0.1*. Technical Report. Quant. 1–48 pages. http://objects-us-west-1.dream.io/files.quant.network/Quant_Overledger_Whitepaper_v0.1.pdf

[216] F. B. Vernadat. 2006. Interoperable enterprise systems: Architectures and methods. In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, Vol. 12. Elsevier, 13–20.

[217] Fabian Vogelsteller and Vitalik Buterin. 2015. EIP 20: ERC-20 Token Standard. https://eips.ethereum.org/EIPS/eip-20

[218] W3F. 2020. Research at W3F. https://research.web3.foundation/en/latest/polkadot/

[219] Gang Wang, Zhijie Jerry, and Mark Nixon. 2019. SoK : Sharding on Blockchain. In *ACM Conference on Advances in Financial Technologies*.

[220] Hongkai Wang, Dong He, Xiaoyi Wang, Caichao Xu, Weiwei Qiu, Yiyang Yao, and Qiang Wang. 2020. An Electricity Cross-Chain Platform Based on Sidechain Relay. In *Journal of Physics: Conference Series*, Vol. 1631. IOP Publishing Ltd, 12189. https://doi.org/10.1088/1742-6596/1631/1/012189

[221] Xinying Wang, Timothy Tawose, Feng Yan, and Dongfang Zhao. 2020. *Distributed Nonblocking Commit Protocols for Many-Party Cross-Blockchain Transactions*. Technical Report. https://arxiv.org/pdf/2001.01174.pdf

[222] Sheila Warren and David Treat. 2019. Building Value with Blockchain Technology : How to Evaluate Blockchain' s Benefits. *White Paper in Word Economic Forum* (2019).

[223] Will Warren and Amir Bandeali. 2017. *0x: An open protocol for decentralized exchange on the Ethereum blockchain*. Technical Report.

[224] WEF. 2020. *Bridging the Governance Gap: Interoperability for blockchain and legacy systems*. Technical Report.

[225] Peter Wegner. 1996. Interoperability. *Comput. Surveys* 28, 1 (1996).

[226] Martin Westerkamp. 2019. Verifiable Smart Contract Portability. *ICBC 2019 - IEEE International Conference on Blockchain and Cryptocurrency* (2 2019), 413–421. http://arxiv.org/abs/1902.03868

[227] Gavin Wood. 2016. *Polkadot: Vision for a Heterogeneous Multi-Chain Framework*. Technical Report. 1–21 pages. https://github.com/w3f/polkadot-white-paper/raw/master/PolkaDotPaper.pdf

[228] Gavin Wood. 2019. *Ethereum: A secure decentralised generalised transaction ledger. Byzantium version 7e819ec*. Technical Report. https://ethereum.github.io/yellowpaper/paper.pdf

[229] Xingtang Xiao, Zhuo Yu, Ke Xie, Shaoyong Guo, Ao Xiong, and Yong Yan. 2020. A Multi-blockchain Architecture Supporting Cross-Blockchain Communication. In *Communications in Computer and Information Science*, Vol. 1253 CCIS. Springer Science and Business Media Deutschland GmbH, 592–603. https://doi.org/10.1007/978-981-15-8086-4{_}56

[230] Xiwei Xu, Ingo Weber, Mark Staples, Liming Zhu, Jan Bosch, Len Bass, Cesare Pautasso, and Paul Rimba. 2017. A Taxonomy of Blockchain-Based Systems for Architecture Design. In *Proceedings - 2017 IEEE International Conference on Software Architecture, ICSA 2017*. Institute of Electrical and Electronics Engineers Inc., 243–252.

[231] Dylan Yaga, Peter Mell, Nik Roby, and Karen Scarfone. 2018. *Blockchain Technology Overview*. Technical Report. NISTIR. https://doi.org/02

[232] Guangsheng Yu, Xu Wang, Kan Yu, Wei Ni, J. Andrew Zhang, and Ren Ping Liu. 2020. Survey: Sharding in Blockchains. *IEEE Access* 8 (2020), 14155–14181. https://doi.org/10.1109/ACCESS.2020.2965147

[233] Alexei Zamyatin, Mustafa Al-Bassam, Dionysis Zindros, Eleftherios Kokoris-Kogias, Pedro Moreno-Sanchez, Aggelos Kiayias, and William J Knottenbelt. 2019. *SoK: Communication Across Distributed Ledgers*. Technical Report. https://eprint.iacr.org/2019/1128.pdf

[234] Alexei Zamyatin, Dominik Harz, Joshua Lind, Panayiotis Panayiotou, Arthur Gervais, and William J Knottenbelt. 2019. XCLAIM: A Framework for Blockchain Interoperability. In *IEEE Symposium on Security & Privacy*.

[235] A. Zamyatin, N. Stifter, A. Judmayer, P. Schindler, E. Weippl, and W. J. Knottenbelt. 2019. A wild velvet fork appears! Inclusive blockchain protocol changes in practice. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 10958. Springer Verlag, 31–42.

[236] P Zappalà, M Belotti, M Potop-Butucaru, and S Secci. 2020. *Game theoretical framework for analyzing Blockchains Robustness.* Technical Report. https://eprint.iacr.org/2020/626.pdf

[237] Dongfang Zhao and Tonglin Li. 2020. Distributed Cross-Blockchain Transactions. *arXiv* (2020).

[238] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. 2017. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In *Proceedings - 2017 IEEE 6th International Congress on Big Data, BigData Congress 2017.* Institute of Electrical and Electronics Engineers Inc., 557–564.

[239] Qingyi Zhu, Seng W Loke, Rolando Trujillo-Rasua, Frank Jiang, and Yong Xiang. 2019. Applications of Distributed Ledger Technologies to the Internet of Things: A Survey. *Comput. Surveys* 52, 6 (2019), 120:1–120:34.

[240] Aviv Zohar. 2015. Bitcoin: Under the Hood. *Commun. ACM* 58, 9 (8 2015), 104–113.

[241] Guy Zyskind Oz, Nathan Alex ', and Sandy ' Pentland. 2015. *Enigma: Decentralized Computation Platform with Guaranteed Privacy.* Technical Report.

A METHODOLOGY

This section presents the methodology we followed in conducting the systematic literature review about blockchain interoperability. Our methodology follows several phases, as advised by several authors, [126, 185]. In the planning phase, we select the research questions, the data sources, the search terms, the practical screening criteria, and the methodological screening criteria. In the review phase, we abstract data from selected papers, identifying the underlying conceptual mechanisms for interoperability. We then correlate approaches *intra-category* and *inter-category* (via the discussion subsections). Finally, we report the review and synthesize the findings.

We give special attention to grey literature, as some authors defend that it includes "a broader scope of literature, providing a more comprehensive view of the available evidence" [126, 146]. In particular, we analyze grey literature as a way to include recent endeavors. In particular, we argue that including grey literature is relevant, as: (i) blockchain interoperability is in active development, and there is still a reduced number of academic studies, (ii) some research is concentrated on the industry, and (iii) grey literature reduces the publication bias [126].

Notwithstanding, grey literature is not often updated (e.g., whitepapers [11, 116, 201, 227]). To the best of our knowledge, we picked the most recent whitepaper versions and made the effort of looking through the documentation for updates. Nonetheless, it is possible that a newer version is available, or that we missed out on relevant information. That is why we systematically contacted the authors of the projects (see Section A.3). This methodology allows us to validate or view of the project at hand while addressing some shortcomings of researching grey literature. Hence, we built a list of references and contacts, which we engaged during our research. We indicate when we obtained feedback from authors on their projects, using the "checkmark" sign (✓). More specifically, the checkmark typically indicates that we have taken the authors or their respective team's feedback into consideration, regarding a specific project. Exceptions occur whenever the legend of a table indicates so (for example, in Table 2, the checkmark indicates that an author discusses the referenced criteria. A caveat of our approach is that grey literature is not, necessarily, quality scientific work, as it is not peer-reviewed [180].

Moreover, in order for our grey literature search to be "systematic, transparent, and reproducible," we adopt recommendations from Mahood et al. [146]. In particular, they recommend "that searches include online databases, web search engines and websites, university, and institutional repositories, library catalogs, as well as contacting subject specialists, hand-searching and consulting reference lists of relevant documents". We then include grey literature, as the result of retrieving references from scientific articles, and consultation with both academics and professionals in the area of blockchain interoperability. We, therefore, define grey literature as: Github documentation, whitepapers, technical and institutional reports, initial coin offer plans, magazine articles, academic dissertations, consultant reports,

book chapters, and blog posts. With such sources, we believe that it is possible to construct a reliable, updated, and extensive understanding of blockchain interoperability.

We believe this approach leads to adequate coverage and transparency in blockchain interoperability research and, consequently, provides accurate information to the reader in a research area evolving so quickly. In a research area on its inception, and given its fragmentation, we acknowledge that we may have missed some advances in this field. We commit to updating our knowledge base in the light of the new information being produced, to yield the most comprehensive results possible.

A.1 Research Questions

Taken into account the different stakeholders of the blockchain technology, and the previous literature reviews limitations, we propose the following research questions, addressed by this paper:

(1) **What is the current landscape concerning blockchain interoperability, both from the industry and the academia?** Bitcoin and Ethereum fostered hundreds of cryptocurrencies and use cases, shortly after their inception. Heterogeneous solutions appeared to further deliver customization, tailored for enterprise use-case scenarios that benefit with blockchain technology. Soon after this solution proliferation, and in particular, with the vast number of platforms emerging, the blockchain interoperability problem started to be tackled by industry and academia [52, 120, 122, 206]. Although some attempts of classifying blockchain interoperability solutions have been made [41, 52, 175], they are either outdated, or not capturing the whole interoperability spectrum.

(2) **Is the set of technological requirements for blockchain interoperability currently satisfied?** According to several authors, the prerequisites for blockchain interoperability are: (i) the existence of a cross-blockchain communication protocol that can transfer arbitrary data in a trustless and decentralized way, comparable to the transport layer of the Internet [103], (ii) a pair of sufficiently mature blockchains that can be bridged through such protocol, and (iii) the need for applications benefiting from a multiple-blockchain approach [52], i.e., IoB-powered BoB applications. This research question is particularly important since it gives a perspective if research and focus should be put in the direction of blockchain interoperability.

(3) **Are there real use cases enabling a value chain coming from blockchain interoperability?** According to some authors [103, 109, 143, 167**?**], blockchain interoperability is a core requirement for the survival of the technology. Given stable, matured blockchain interoperability mechanisms, one needs to explore which solutions can be built, which sectors it may benefit, and what are the use cases foreseeable in the short and medium-term.

A.2 Data Sources

The online repository using for the majority of the research is Google Scholar. Google Scholar is a modern search engine owned by Google, which indexes most major digital libraries, including but not limited to IEEE Xplore, ACM Digital Library, Science Direct (another major search engine for digital libraries), ASCE, Scopus, Web of Science, SpringerLink, and arXiv (known for containing grey literature). According to Google's documentation[28], "Google Scholar includes journal and conference papers, theses and dissertations, academic books, pre-prints, abstracts, technical reports, and other scholarly literature from all broad areas of research". It includes "academic publishers, professional societies, and university repositories, as well as scholarly articles available anywhere across the web. Google Scholar also includes court opinions and patents". It covers grey literature, making it a suitable option to reduce the publication bias [126].

[28] https://scholar.google.com/intl/en/scholar/help.html#coverage

Google Scholar's coverage is arguably the biggest across other academic search engines for Computer Science [82], and it meets the criteria recommended in guidelines for conducting systematic literature reviews [46, 82]. Fagan critiques Google Scholar for giving too much importance to the citation count and therefore suggests the usage of additional search tools to conduct the review [82]. However, as we are aiming for a bigger coverage, by studying most work concerning blockchain interoperability up to this date, the bias introduced by the citation count does not significantly impair our study. Hence, and to simplify our research process, we rely on Google Scholar.

Furthermore, in order to add resiliency to our study, we compiled a list of appropriate search terms from our knowledge of the literature – previous searches on this topic, well-known projects on the community and suggestions from other researchers, to identify additional references not previously captured. Such references were included in the review.

A.3 Search Process

We divided the search process into three phases: searching for related literature reviews, searching for relevant peer-reviewed scientific papers, and searching for relevant grey literature.

We aim to find relevant literature directed to blockchain interoperability, which can be synonyms with *chain interoperability*, *interconnected blockchain*, *multiple blockchains*, and *internet of blockchains*. One could consider the concept of *blockchain sharding* a novel solution to address blockchain scalability, which can ultimately foster blockchain interoperability since shards need to communicate with each other. However, due to the extension of the blockchain sharding research area, and because of space constraints, we purposely leave it out of the scope of this research.

In the first phase of the search process, *identification*, we queried *"interblockchain survey"* OR "blockchain interoperability survey" OR *"IoB"*, where we obtained 86 results. From those 86 results, only one was explicitly a literature review concerning blockchain interoperability (i.e., contained the term "survey" at the title).

Next, we performed a keyword-based search. We limited the scope of queries until the present date of writing, i.e., the 14th February 2020, thus covering literature up to the present day. Notwithstanding, we updated this paper with both academic literature and grey literature dated up to the end of May 2020. Google Scholar treats all terms specified in the search query as an *AND* operator: it yields search results for all the terms. Henceforth, all queries presented in this document assume such quotes. Therefore, we opt by restricting this feature, as querying *blockchain interoperability* yields more than 9,000 results. By using quotes in the search, we limited its range. Hence, a query with the keywords *blockchain* and *interoperability* yields results only if both terms are present. We then searched the terms *interchain communication*, *interconnected blockchain*, and *blockchain interoperability*, as they semantically seem the most suitable terms for our search. We obtain 262 results: and chose not to include terms as *multiple blockchains* or *chain interoperability*, because although related, those terms are too vague and yield too many results not directly related to this study, respectively 494 and 665 results.

In the third phase, we collected relevant work classified as grey literature. We retrieved the collected reference list and used techniques as snowballing to expand our document repository further. We obtain an additional 69 documents.

A.4 Screening and Eligibility Processes

In this section, we define our methodology for the eligibility criteria. Figure 5 represents an adapted *Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA)* diagram [139], considering all steps of our literature research methodology.

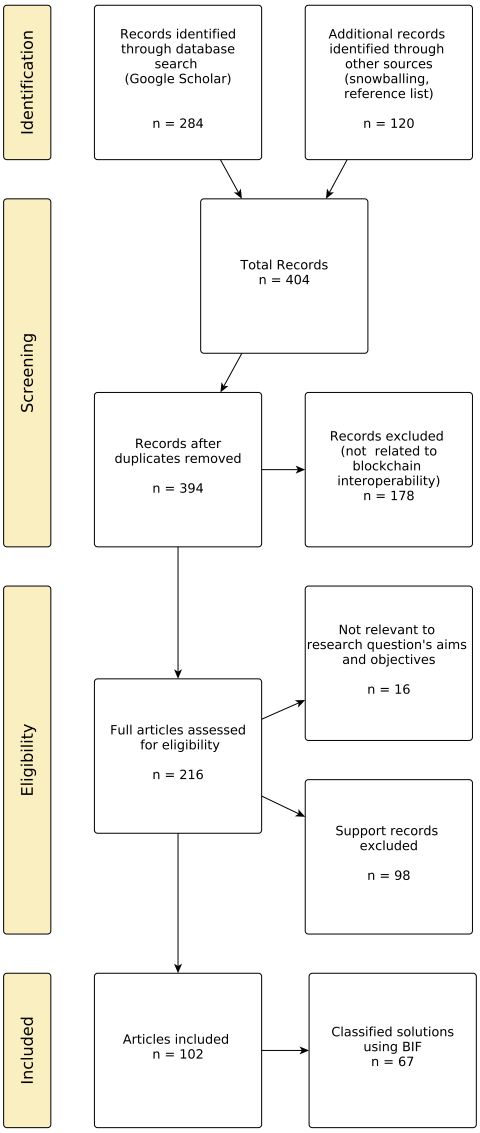

Fig. 5. PRISMA diagram specifying our literature research methodology.

In terms of the included documents (papers, grey literature), we first examined the title, abstract, and keywords. When these three elements do not provide enough insights to decide on whether include the document on this study, we examined the full-text body of the documents. This first screening aims to conclude about the feasibility of a given document to answer the proposed research questions.

Due to the small number of available papers, we had a lenient approach regarding the exclusion criteria: we only excluded papers that do not comprehensively tackle blockchain interoperability. For example, papers which focus is

state of the art on blockchain applications, security, scalability, consensus mechanisms, and economic models, even if they tackle blockchain interoperability, are excluded. In contrast, papers with at least a section dedicated to blockchain interoperability are taken into consideration. The process above leads to a total number of 404 documents. After excluding 178 non-related papers, 10 duplicates, 16 not relevant papers, and 98 support papers (papers that, although crucial for the understanding of this topic, they are not included in the comparison of solutions), we achieve a total of 102 documents, from which 67 were systematically compared.

B AN ARCHITECTURE FOR BLOCKCHAIN INTEROPERABILITY

This section discusses existing architectures for interoperable blockchains, the "internet of blockchains" approach. We then present a consolidated architecture.

Zhu et al. define several layers for a blockchain [239]. The *data layer* defines the representation of data in the blockchain (e.g., transactions aggregated into blocks vs transactions represented in a directed acyclic graph). The *network layer* defines the type of nodes in the peer-to-peer network (e.g., full nodes and light nodes [158]). The *consensus layer* represents the consensus algorithm the network uses and its security assumptions. The *contract layer* represents the execution environment for smart contracts, which provide the foundation for the application layer, which include the blockchain-enabled business logic.

Other authors proposed architectures for blockchain interoperability composed of several layers: Jin et al. proposed the data, network, consensus, contract, and application layers [117], while Kan et al. proposed the basic, blockchain, multi-chain communication, and application layers [120].

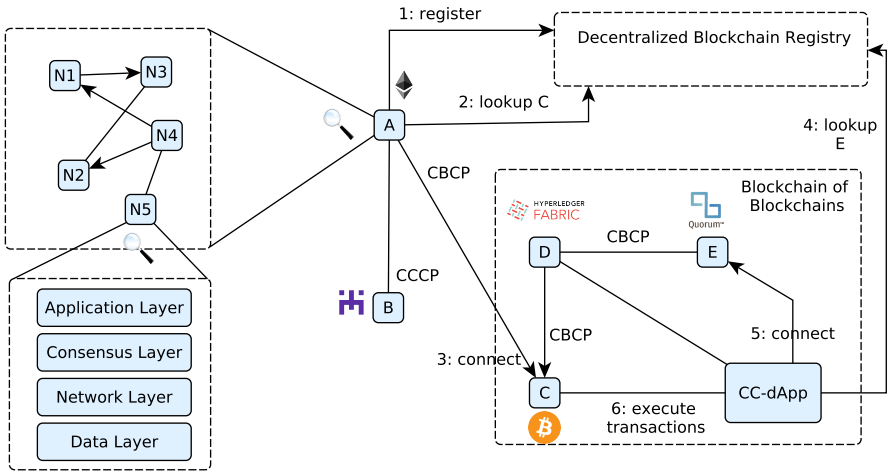

Fig. 6. Architecture for Interoperable Blockchains: a network comprised of five blockchains (A to E) and a cross-chain decentralized application (CC-dApp).

Hardjono et al. proposed an architecture inspired by the architecture of the Internet [103]. The proposed architecture has as central concepts the Autonomous System (AS) (or *routing domain*) and gateway. A routing domain is a network ecosystem operating with specific rules, under an administrative domain. An AS is a set of IP networks that form a single administrative domain, which maps to a blockchain network. A gateway supports cross-domain routing in order

to allow communication among networks in different ASs. Gateways are nodes that support interoperability, such as smart contracts or trusted third parties.

Our proposal is influenced by previous work: in particular, we envision each blockchain as an autonomous system, which communicates to others via a cross-blockchain protocol. Most nodes on public and private blockchains can serve as interoperability gateways. To facilitate communication among blockchains, one can rely on decentralized blockchain registries, that can identify and address oracles, blockchains, and their components (e.g., smart contracts, and certificate authorities) [206]. A registry for both public and private blockchains could be written in a public blockchain with strong security assumptions (e.g., a high degree of decentralization). Alternatively, the contents of the registry can be recorded in a custom public blockchain maintained by the stakeholders of major blockchains, or enforced by trusted hardware [103]. The decentralized registry would act as a (preferably) decentralized domain name system [153], but for blockchains instead of domains. A simple implementation would be leveraging a multi-signature Ethereum smart contract where a consortium could manage a registry of gateway nodes.

We leave further discussions on a decentralized blockchain registry for future work. Note that this registry is optional, and it is not essential for enabling an IoB.

Figure 6 illustrates our proposal for an architecture for the IoB, the enabler of technical interoperability. Although we represent a BoB in the figure, we do not detail its architecture at this stage. Blockchain$_A$ (A) and Blockchain$_B$ (B) are both public, EVM-based blockchains, namely Ethereum and POA Network. Blockchain$_D$ (D) and Blockchain$_E$ (E) are private blockchains, namely Hyperledger Fabric and Quorum. A blockchain node belonging to the Ethereum network, Blockchain$_A$, registers its communication endpoint (i.e., IP address) on the blockchain registry (step 1). After that, it looks up for the address of a node belonging to Blockchain$_C$ (C), Bitcoin (step 2). CCCP and CBCP protocols can provide unilateral or bidirectional interoperability. In step 3, a CBCP establishes communication between the Ethereum node and the Bitcoin node, unilaterally, since the Ethereum node can read Bitcoin's blocks headers (e.g., via [79]), but not the other way around. Blockchain$_D$ and Blockchain$_E$ are heterogeneous, thus connected by a CBCP. A CC-dApp is already connected to blockchain$_C$ and blockchain$_D$, and further connects with blockchain$_E$, after fetching its address on the blockchain registry (steps 4 and 5). Step 4 assumes the necessary credentials to access the private blockchain are held by the CC-dApp user(s) (e.g., private keys, X.509 certificates). A CC dApp protocol allows an end-user to realize the semantic interoperability, by leveraging blockchain$_C$, blockchain$_D$, and blockchain$_E$ (step 6). These steps accomplish connectivity among blockchains, thus forming an IoB, and therefore enabling a BoB.

CCCPs (e.g., XClaim [234]) and CBCPs (e.g., inter-blockchain protocol [113] or the Interledger Protocol [115]) can be employed to manage the end-to-end communications between blockchain networks, addressable by the blockchain registry. While such protocols can provide seamless interoperability for future blockchains, via standardization, they are not compatible with existing blockchains. Existing blockchains would require to refactor several layers: the network, consensus, contract, and application layers [239], would need to be changed.

In Figure 7, we model the layers of blockchain interoperability that correspond to the proposed architecture, using the Archimate modeling language [211], a standard for enterprise architecture modeling. Blockchain interoperability, technical interoperability and semantic interoperability are capabilities, abilities that the business processes "Internet of Blockchains" and "Blockchain of blockchains" possesses (as they enable interoperability at different levels). "Cross-chain protocols" and "cross-chain dApp protocols" are applicational components that realize the "cross-chain transaction" function. Other interoperability layers are left for future work.

Regardless of the interoperability solution employed, it is likely that the network layer has to suffer refactoring, and consequently the consensus layer since there are blockchains with different transaction finalities [67]. Transaction

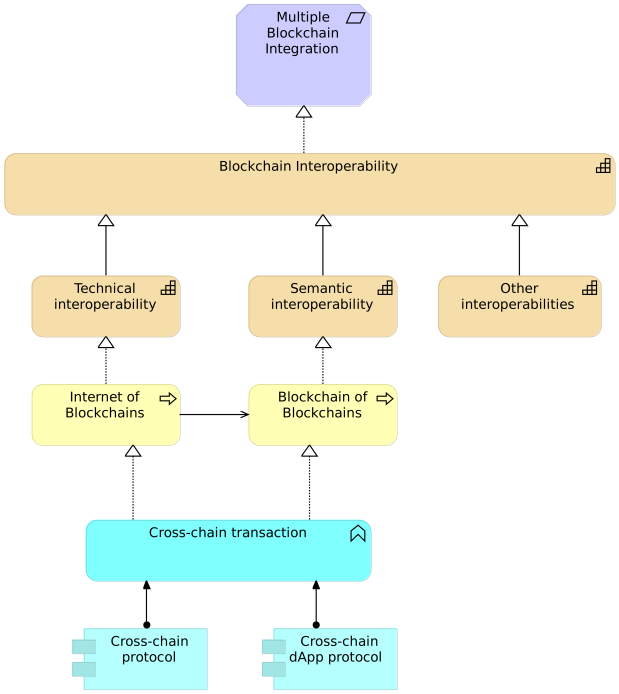

Fig. 7. Simplified blockchain interoperability model, represented in Archimate

finality can be probabilistic or deterministic, and refers to when parties involved in a transaction can consider it committed to the blockchain. For example, Bitcoin needs around 6 confirmed blocks to consider a transaction final with a high probability (probabilistic), whereas Tendermint transactions are final right after their execution (deterministic). Several abstractions that include transactions from other blockchains can be implemented on the contract layer. These changes have repercussions on the application layer, as now it can handle more complex operations. The application can now expose APIs to dispatch cross-blockchain transactions, as illustrated in some works [143, 155, 215]. The data layer would not necessarily have to be changed.

Although this could be a viable solution, it is logistically cumbersome to adjust all blockchains in production to use a specific set of inter-blockchain protocols and to adapt their different layers. As this solution is not feasible in practice, at least in the short term, blockchain interoperability solutions are typically tailored for a specific blockchain or a set of specific blockchains. Nevertheless, we believe that as the technology matures blockchain interoperability standards will guide technical efforts, leading to convergence towards interoperability within the blockchain space.

Throughout this paper, blockchain-agnostic solutions, as well as specific solutions will be presented and discussed.

Table 5. Comparison of *Sidechains* solutions

Reference	Mainchain	Sidechain consensus	Summary	Strong points	Weak points	Roadmap
BTC Relay [79] ✓	Ethereum	×	Ethereum smart contract reading Bitcoin's blockchain	Simple solution relying on verifying block headers	Limited functionality	None
Peace Rekay [131]	Ethereum	×	SPV on EVM-based blockchains	Allows two way pegs	It is expensive to verify Ethereum block headers	None
Testimonium [89]	Ethereum	×	EVM-based blockchains SPV	Effiecient validation	Mainly support EVM-based blockchains	Batch submission of block headers
POA Network [11] ✓	Ethereum	Proof of authority	Applicational interoperability to EVM-based dApps	Inexpensive consensus	Validators confined to one country (geographic concentration)	POA-based stable token
Liquid [15**?**] ✓	Bitcoin	Strong federations	Strong federation-based settlement network	Strong federation of functionaries maintain the network	Consensus secured by specialized hardware	Wallet and mining services
Loom Network [144] ✓	Ethereum	Delegated proof of stake	dApp platform with interoperability capabilities	Support for a high number of tokens	Closed source solution	Integrations with major blockchains
Zendoo [93]	Bitcoin	Proof of stake	Sidechain creation platform	zk-Snark solution allows the mainchain to verify the sidechain without disclosing sensitive information	zk-Snarks are computationally expensive	Further specification of the protocol
RSK [135] ✓	Bitcoin	DECOR+	Federated sidechain, in which RBTC is tethered to BTC	Merge mining allows reutilization of work	Relies on PoW, energetically inneficient	Decentralized bridge with Ethereum
Blocknet [65] ✓	Ethereum	Proof of stake	EVM-based blockchain with interoperability capabilities	Blocknet protocol allows trustless blockchain interoperability	Currently limited to digital assets	EOS/NEO/other integrations

✓our description was endorsed by the authors/team
× not specified
∗ although zk-Snarks are not a consensus algorithm, consensus on which operations were performed at each sidechain is obtained through a process that uses zk-Snarks to generate proofs of sidechain state that, on its turn, generate certificate proofs for the mainchain

C PUBLIC CONNECTORS

C.1 Sidechains

We now describe some of sidechain solutions we identified in the literature. Table 5 summarizes these solutions. An analysis of this table is conducted in the discussion.

The *Peace Relay* is inspired by BTC Relay, allowing communication between EVM-based blockchains [131]. Peace allows Ethereum contracts to verify account states and transactions from Ethereum Classic, and vice-versa, allowing a two-way peg (given that the Peace relay smart contract is deployed on both chains).

Testimonium is a relay solution that follows a validation-on-demand pattern, validating blockchain block headers on-chain [89]. As block headers are accepted optimistically, validation-on-demand locks block headers for a specific lock time, where off-chain clients (disputers) can challenge their validity.

POA Network encompasses an EVM-based blockchain as well as the POA Bridge [11]. The POA Bridge is a component that enables cross-application transactions with Ethereum, providing support for ERC-20 tokens. For instance, the POA20 token represents the POA token available to use on the Ethereum main network. The sidechain achieves consensus through proof of authority.

A newer feature from POA, *Arbitrary Message Bridge*,[29] allows transferring arbitrary data between EVM-based chains (e.g., POA, Loom, Ethereum Classic). This feature can be used for cross-chain smart contract invocations. POA implemented a POA-based stable token, through the xDai chain.[30] POA is an open-source project.[31]

Elements[32] is a sidechain-capable blockchain platform. *Liquid* is a federated pegged sidechain [15**?**], based on Elements, relying on the concept of *strong federations* [74]. Strong federations introduce the concepts of a federated two-way peg, in which entities move assets between two chains. In strong federations, a role called block-signers maintains the consensus of the blockchain, while the watchmen realize cross-chain transactions. Software running on hardware security modules achieve consensus. Hardware security modules (HSMs) are physical computing devices that actively hides and protects cryptographic material, e.g., via limited network access and features that provide tamper evidence [183]. Moreover, a *k-of-n* multi-signature scheme is also used to endorse block creation.

Liquid supports several assets, including fiat currencies and cryptocurrencies, such as Bitcoin. When Bitcoins are pegged to the Liquid sidechain, they are backed by an L-BTC token, which represents one Bitcoin. The roadmap predicts updates to wallet and mining services[33]. Liquid is an open-source project[34].

Loom Network is a dApp platform, which relies on sidechains connected to Ethereum, Binance Chain, and Tron [144]. Loom is a federated two-way peg, whereby a set of 21 validators and token delegators validate cross-asset transactions. Loom uses *Delegated Proof of Stake* (DPoS) as the consensus mechanism for transactions happening on the sidechain.

Proof of Stake (PoS) is an alternative to PoW that aims to reduce energy consumption [63]. In PoS, the ability for nodes to append blocks to the ledger depends on their stake, that often depends on the amount of currency they own. In DPoS only a subset of the nodes participate in the consensus, which is based on PoS.

The roadmap predicts integration with more blockchain networks[35]. Loom is open-source components[36].

RSK is a general-purpose smart contract platform pegged to the Bitcoin network that offers improvements in security and scalability of the latter [135], and the first sidechain solution in production (January 2018). It relies on a combination of a federated sidechain with an SPV. Each smart Bitcoin (RBTC), the native token of RSK, is tethered to one Bitcoin.

In order to get RBTCs, a user has to send Bitcoin to a specific multi-signature address (an address controlled by several parties, through the several signatures) located at the Bitcoin network. That address is controlled by the RSK Federation, which is composed of several stakeholders. The federation members use hardware security modules. By leveraging HSMs, each validator can protect its private keys, and enforce the transaction validation protocol [135]. Moreover, an additional layer of security that prevents any corrupt collaborator from forcing the HSM from each stakeholder to sign a fake peg-out transaction: nodes automatically follow the blockchain with the highest cumulative proof of work.

After the transaction is finished, a proof of transfer (via SPV) is generated and given as an input to a smart contract on the RSK network, called the bridge contract. The bridge contract then sends a corresponding amount of RBTC tokens to the address present at the RSK network that corresponds to the Bitcoin address sending Bitcoin to the RSK address. RSK has a virtual machine that executes smart contracts in the Bitcoin network.

[29] https://docs.tokenbridge.net/amb-bridge/about-amb-bridge
[30] https://www.poa.network/roadmap
[31] https://github.com/poanetwork
[32] https://elementsproject.org
[33] https://blockstream.com/2020/02/10/en-blockstream-2019-review-building-foundations/
[34] https://github.com/Blockstream?q=liquid&type=&language=
[35] https://medium.com/loom-network/5183ce02267
[36] https://github.com/loomnetwork

RSK uses consensus mechanism designated DECOR+ and a technique called merge-mining, which allows users to mine in both the RSK and Bitcoin networks without performance penalties. RSK introduces shrinking-chain scaling, a technique to compress blocks after they are mined.

The RSK roadmap predicts the development of a decentralized bridge between RSK and Ethereum[37]. RSK is an open-source project[38].

Blocknet is blockchain based on PoS that includes a protocol for interoperability among public and private blockchains [65]. At its core, Blocknet has several components: the XBridge, XRouter, and XCloud [39, 40]. XBridge allows exchanging digital assets, powered by a set of APIs, and relying on SPV. XRouter actuates as an inter-chain address system, providing lookup capabilities to the network. XCloud, relying on XRouter, provides a decentralized oracle network, that can be used to obtain trusted data.

C.2 Notary Schemes

Despite this evolution, commonly used notary schemes are centralized cryptocurrency Exchanges (e.g., Binance, Coinbase, BKEX, LBank, Bilaxy, BitForex). Most exchanges are centralized (237), against 22 decentralized exchanges listed by CryptoCompare, at the time of writing.[39]

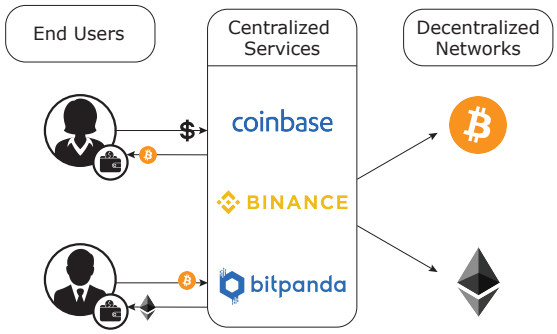

Fig. 8. Alice and Bob buy cryptocurrencies via a centralized exchange. The assets are held by a custodial wallet.

Figure 8 represents the task of a user acquiring cryptocurrencies via centralized exchanges. Users buy cryptocurrencies with fiat currencies, and are credited the bought assets on their respective wallets, owned by the exchange, i.e., the exchange also known as *custodial wallets*. Exchanges acquire such cryptocurrencies directly on the network, or via an intermediary, and provide arbitrage services.

Although a simple way to obtain cryptocurrencies, some attacks have been conducted to exchanges, leading to loss of very large cryptocurrency sums [3].

Decentralized exchanges can be implemented with hashed timelocks (see Section 5.1.3), or other technologies (see Section 6.3). Figure 9 depicts users exchanging assets via a decentralized exchange (e.g., Nash, AtomicDEX, IDEX). When trading via a decentralized exchange, users typically do not disclose their private keys, eliminating the single point of failure inherent with centralized exchanges.

[37] https://blog.rsk.co/noticia/hawkclient-building-a-fully-decentralized-bridge-between-rsk-and-ethereum/
[38] https://github.com/rsksmart
[39] https://www.cryptocompare.com/exchanges/#/overview

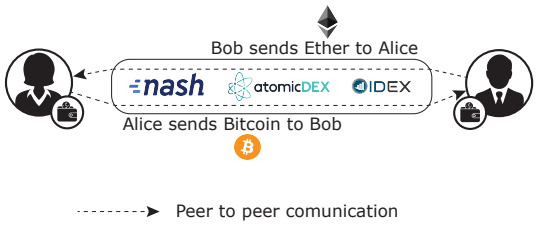

Fig. 9. Alice can send cryptocurrencies directly to Bob, and vice-versa. Each user holds their private keys. The exchange is a facilitator of the transactions.

Table 6. Comparison of *Hash Lock Time Contract* solutions

Reference	Supported Chains	Architecture	Summary	Strong points	Weak points	Roadmap
Black et al. [38]	×	Lender, borrower	Leverage HTLC to provide fiat/stablecoinaccess for cryptocurrency holders	Decentralized solutions	Inneficient (atomic swaps); requires over-collateralization	×
Wanchain [145] ✓	Bitcoin, Ethereum	Vouchers, validators, storemen (Wan protocol)	Connects major currency exchanges	Cross-Chain Bridge Node Staking Rewards	Storemen are not completely decentralized	General interoperability
LN [174]	Bitcoin	Relies on multi-signature channel addresses	High volume, low latency micropayment enabler	Increases Bitcoin performance, solution in production	Timelock expiration exploits	×
Komodo [129] ✓	Bitcoin, Ethereum	Liquidity provider nodes, buyers, sellers	Atomic swap decentralized exchange	Provides a framework for cross-chain atomic swaps	All products are "highly experimental"	Derivative tokens on the decentralized exchange
COMIT [60]	Bitcoin, Ethereum	Traders, COMIT protocol	Open protocol facilitating trustless cross-blockchain applications	Adds negotiation phase to the atomic swap	Does not support negotiation protocols	Protocol for privacy preserving swaps

✓our description was endorsed by the authors/team
× not specified

Agent Chain is a project aiming to exchange assets between blockchains using a multi-signature scheme [136]. A trader maps the possessed assets to AgentChain, which combines several trading operators in a trading group. Members of that group generate an account using a multi-signature, to serve as a deposit pool, containing the assets. Tokens are then locked. An arbitration mechanism is introduced in case of a malicious trading group.

C.3 Hashed Time-Locks

Black et al. propose the concept of *atomic loans*, based on atomic swaps [38]. Atomic loans allow market participants to create loans in a trustless manner, enabling liquidity. The process of atomic loans is rooted in the foundations of HTLCs and has several phases: the loan period, in which the loan withdrawal and repayment process is handled; the bidding period; the seizure period; and the refund period. The last four phases happen in case the loan is not repaid in due time during the bidding period phase.

Wanchain aims to provide deposit and loan services with cryptocurrencies [145]. When a transfer request is sent to Wanchain, it issues the corresponding tokens in the existing smart contract that locks them on the target blockchain. Wanchain's validator nodes receive such request, verify that a transaction has been placed into the target blockchain, and creates a representation of the tokens to be transferred (a new smart contract token, analogous to the original currency).

When a party that has a representation of the original tokens wants to send them to a third party, the locked assets in a smart contract are released to the beneficiary of the transaction. As Wanchain creates a representation of tokens as a means of exchanging assets, we can consider that such a solution is a notary scheme, although decentralized (several validator nodes operate the network). Wanchain's architecture includes the following nodes: vouchers, the cross-chain transaction proof nodes; validators, the verification nodes; and storeman, the locked account management nodes. Vouchers check whether a transaction has been confirmed on a source blockchain. Validators verify the asset registry from the source blockchain: in case it is a new asset, it is registered and added into the registry. Storeman manages locked accounts, facilitating cross-chain transactions. An incentive mechanism rewards the participants to perform their functions. More recently, Wanchain is working towards more general interoperability, by promoting cross-chain integration with enterprise blockchains and supporting Web Assembly (WASM) smart contracts [85].

COMIT is a protocol allowing for atomic swaps, based on HLTCs [60]. COMIT defines several atomic swap protocols that support different cryptocurrencies and tokens, such as HAN (HTLCs for Assets that are Native to the ledger), HErc20 (HTLCs for the Erc20 asset), and HALight (HTLCs for Assets on the Lightning ledger). COMIT nodes can trade Bitcoin for Ether or ERC-20 tokens. The COMIT protocol[40] allows one to exchange assets directly with another user (e.g., Bitcoin for Ether).

Apart from HLTCs and sidenchains, there is a set of approaches that share characteristics from several subcategories presented, for instnace, using distributed private key schemes or collateralization with HLTCs. *Distribute private key approaches* rely on the distribution of users' and organizations' private keys, i.e., in splitting each private key in a set of parts [69]. This leads to distributing the control of assets among several parties. Such schemes can be used to implement decentralized two-way pegs, as well as decentralized notaries. Other approaches combine sidechains and protocols based on escrow parties, relying on smart contracts. An *escrow* is an arrangement in which a third party regulates a transaction or group of transactions between two parties. An escrow typically holds assets (e.g., cryptocurrency) from one of the parties that serves as the collateral of a transaction (assets pledged by a borrower to protect the interests of the lender). Some of those solutions include:

Tokrex enables the exchange of cryptocurrencies between different blockchains in a decentralized way, by leveraging the concept of *meta-swap* [**?**]. A meta swap happens when a sender transmits his private key instead of signing an on-chain transaction. For that, a domain-specific language, Tokrex TLQ, allows developers to write cross-chain applications that run on a decentralized network infrastructure. Tokrex relies on escrow nodes distributing the generated keys, a modularized distributed key generator, cross-chain swaps, and an Incentivization scheme to keep the escrow and validator nodes honest.

Fusion is an interoperable blockchain, focused on financial use cases [91]. Fusion owns a proprietary technology, DCRMS (Distributed Control Rights Management System), which allows users to lock-in and lock-out assets across blockchains. DCRMS is a decentralized custodian model, which tries to prevent private keys from being a single point of failure: asset control is decentralized along network nodes, instead of them relying on individuals and centralized organizations. The distributed storage and generation of a private key keeps a single entity of obtaining full control of an asset. Fusion supports any chain that uses EcDSA signatures, which includes Bitcoin, Ethereum and other EVM-based blockchains.

TAST (Token Atomic Swap Technology) is a project[41] that aims to create the first multi-blockchain token system [165]. TAST includes several components explained in a set of documents.

[40] https://github.com/comit-network/comit-rs/
[41] https://dsg.tuwien.ac.at/projects/tast/

Table 7. Comparison of *Alternative* solutions

Reference	Main Supported Chains	Architecture	Summary	Strong points	Weak points	Roadmap
Tokrex [?] ✓	×	Validation and escrow nodes, distributed key generation	Cryptocurrency exchange enabling meta-swaps	Allows "real time" value exchange	Both sender and receiver know the private key used for asset transfer	×
Fusion [91] ✓	Ethereum	FUSION distributed control rights services	Distributed storage of a private key and cryptoasset mapping	Distributes trust and responsability of managing private keys	Does not provide instant atomic swaps	Decentralized oracle services
Sai et al. [187]	Ethereum	Neutral observers	Neutral observers monitor transactions to avoid double spending	Trustees can choose any node to be an observer	Trustees that choose observers are assumed to be honest	Behaviour of malicious trustee
XClaim [234]	Bitcoin, Ethereum	Requester, sender, receiver, redeemer, the backing vault, issuing smart contract	HTLC-based trustless protocol that manages crpytocurrency-backed assets	Good performance compared to traditional HLTCs	Over-collateralization can lead to locked funds	Asymmetric and non-fungible cryptocurrency-backed assets
DeXTT [43–45, 198]	Ethereum	PBTs, claim-first transactions, deterministic witnesses	A protocol implementing eventual consistency for cross-blockchain token transfers.	Ensures eventual consistency of balances across blockchains	Veto contest poses strict requirements towards signed PoIs	DeXTT implementation on OmniLayer
XChain [194]	Ethereum	Directed graph, 3PP: contract creation, secret release, and secret relay	A 3PP for general cross-chain transactions	Generates custom smart contracts for performing cross atomic swaps	Only applicable to Ethereum	×

✓our description was endorsed
× not defined

In one of these documents, the authors present *claim-first transactions*, a protocol for decentralized blockchain asset transfers. [43]. The protocol includes the role of witness, who verifies cross-blockchain transactions and is rewarded for that. Another document presents the notion of *Proof of Intent* (PoI) [44], a cryptographic construction that implements claim first transactions. The notion of deterministic witnesses is introduced as the mechanism for assigning rewards to parties observing claim-first transactions.

In [41], the authors present the design of a blockchain interoperability solution based on an atomic cross-chain token transfer protocol. Other documents summarize the work developed [87, 197] and discuss the requirements for more efficient cross-blockchain token transfers.

In [198], the authors propose an incentive structure for blockchain relays, presenting an enhanced prototype based on SPV. The presented solution showed that the solution incurred in high operation costs. The most recent whitepaper, [88], introduces optimizations that reduce such costs. This paper shows the applicability of a cross-blockchain token, relying on token incentives and simplified payment verification.

DeXTT is an atomic cross-chain token transfer protocol that migrates assets – Pan-Blockchain Tokens (PBTs) – that can exist in different blockchains simultaneously [45]. DeXTT is part of the TAST project.

DeXTT provides eventual consistency of asset balances across blockchains. Eventual consistency, guarantees that eventually all accesses to an item that has not been updated after the access request will return the latest value. To achieve eventual consistency, the authors use a technique called *claim first transactions* [43], and observers. The *claim transaction*, immediately claims the asset before it is marked as spent, through a *SPEND transaction*. The party creating a SPEND transaction is called a *witness*, the rewarded party. Observers observe a transfer and propagate such information across blockchains. As several observers might compete for a reward, a solution called *deterministic witnesses* is proposed [44, 45]. Deterministic witnesses solve the problem of assigning witness awards by defining a witness context, whereby observers participate.

A cross-blockchain asset transfer starts with a *transfer initiation*. In a transfer initiation, a wallet$_a$ expresses the intent of transferring an asset to a wallet$_b$, by signing a transaction with its private key. Wallet$_b$ then countersigns the

transaction, using its private key, (creating a PoI). A PoI proves that a transfer is authorized by both the sender and the receiver. After that, the receiver can then publish the PoI using a *CLAIM* transaction, used to redeem the assets. Only one PoI from a source wallet is valid at each time, eliminating double-spends.

Right after a PoI is published on a blockchain$_a$, the balance of both wallets has not been updated. In order to propagate this information to the other blockchains, in particular blockchain$_b$, the protocol follows the *witness contest* phase. Here, observers become contestants that propagate the PoI to other blockchains, through a *CONTEST* transaction. After that, in the *deterministic witness selection* phase, the destination wallet, wallet$_b$, posts a *FINALIZE* transaction on each blockchain, finalizing the contest and awarding an observer. The double-spending problem is eliminated via *VETO* transactions, which can be called by any party, and discloses conflicting PoI (e.g., a source wallet tries to send more assets than it owns to several destination wallets).

DeXTT tolerates blockchain failures, as long as at least one blockchain remains functional. It is meant to be a blockchain agnostic solution, but the most straightforward framing is within public blockchains. The authors presented a proof of concept using Solidity[42].

XChain includes a three-phase-protocol that generalizes atomic cross-chain swaps, in which two entities, the leaders and the followers exchange assets [194]. Hashed timelock contracts are leveraged to resolve the order of issuing contracts and reedeming locked funds from smart contracts. Nodes that create the HLTCs are called leaders, which first release the secrets; followers execute transactions that react to the leaders' actions (i.e., when a leader shares the secret of the HTLC to a follower, the follower unlocks its smart contract, and receives funds from other entity, by sharing the received secret). This solution is based on HLTCs and a protocol that guarantees end-to-end and uniformity properties.

D BLOCKCHAIN OF BLOCKCHAINS

We now describe some of sidechain solutions we identified in the literature. Research on Blockchain of Blockchains required substantial *ad-hoc* research, including blog posts, roadmaps, and update announcements, for us to build an updated understanding regarding the latest capabilities of each blockchain engine.

The *Polkadot* network has several entities engaged in handling transactions: *collator*, *validator*, *nominator*, and *fisherman*. Collators produce proofs for the validators. Transactions are then executed and aggregated in blocks. There is the possibility of collators *to pool*, to coordinate and share the rewards coming from creating blocks on the parachains they actuate. Validators produce and finalize blocks on the relay chain. The validator role is contingent on a stake that is put on hold to foment good behavior. Validators who misbehave can have their block rewards denied or, in case of recurrence, have their security bond confiscated. Validators are the equivalent to groups of cooperating miners that share block rewards proportionally to their contribution (mining pools) on PoW systems (e.g., Bitcoin). Nominators provide their own stake to validators, whereby sharing the rewards and incurring in potential slashing, in case of misbehaving.Fishermen get bounties for reporting validators' misbehavior, such as helping to ratify an invalid block.

Figure 10 depicts the several components constituting Polkadot. Polkadot's relay chain uses Substrate. Polkadot's state machine is compiled to WASM, a virtual environment that can execute the state transition functions [218]. Libp2p is a network library for peer-to-peer applications, written in the Rust programming language. Parachains run the application logic, creating transactions as needed. Collators group those transactions and redirect them to Validators, who then deem blocks as valid or invalid. After that, the valid ones are added to the relay-chain.

[42] https://github.com/pantos-io/dextt-prototype

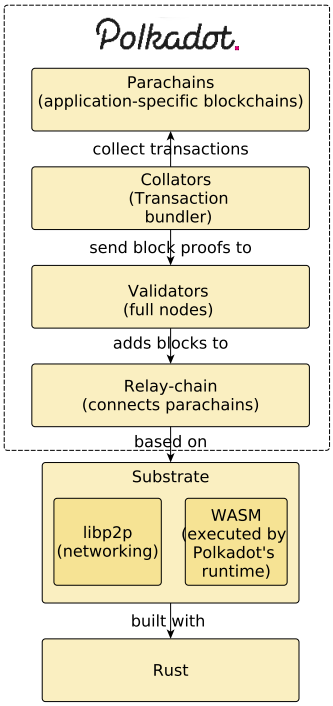

Fig. 10. Polkadot's stack [218, 227]

Polkadot uses the DOT token as an incentive for nodes to behave correctly. DOT has several purposes: (i) decentralize governance (i.e., protocol updates), (ii) operation (i.e., rewarding good actors), and (iii) bonding (i.e., adding new parachains).

Polkadot's relay chain achieves consensus using BABE and GRANDPA [172]. BABE is the block production algorithm, and GRANDPA is the finalizing algorithm. To determine a set of validators, Polkadot uses selection based on PoS, designated Nominated Proof-of-Stake (*NPoS*). Allying NPoS with the rewarding mechanism helps to diminish the impact of attacks such as short-range attack (when a validator attempts to ratify both branches of a fork) or the nothing-at-stake attack (where the risk of simultaneously validating several forks is exploited). The roadmap comprises the launch of the main network[43].

Cosmos is another popular Blockchain of Blockchains. Figure 11 gives a general overview on the Cosmos Network stack. Wrappers can be developed to allow the usage of other programming languages. The applicational layer can be developed with the Cosmos SDK, a framework. This layer connects to the Tendermint BFT Engine (the component responsible for consensus).

Cosmos was limited to asset token on its original inception, now it supports arbitrary data transfers. For CC-Txs, the *relayer* pays a transaction fee on behalf of the transaction sender. The relayer can whitelist any type of financial incentives to keep CC-Txs free.

[43] https://wiki.polkadot.network/docs/en/learn-roadmap

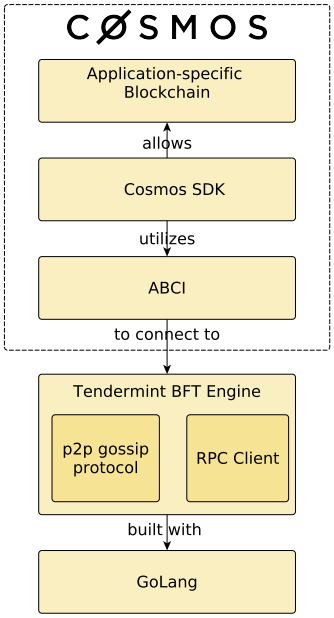

Fig. 11. Cosmos Network's stack [130]

In Cosmos, validators process blocks of transactions. Validators need to stake ATOM tokens to process blocks and earn transaction fees. Delegators can offload transaction processing to validators, and earn transaction fees. As a way to promote an open-governance model, participants (e.g., validators and delegators) can hold the ATOM token and vote on proposals that can change the parameters of the system. Decisions about the network governance, to vote, validate, or delegate transaction validation to other validators are made as a function of how many Atoms are held, similarly to a PoS view. Atoms can also be used to pay transaction fees.

In Cosmos each zone is sovereign, i.e., it can define, for instance, authentication of accounts and transactions, on-chain governance proposals and voting, validator punishment mechanisms, fee distribution and staking token provision distribution, and creation of new units of staking token.

ARK utilizes smart bridges to make instances of its platform interoperable [12]. A smart bridge has two components. The first, *Protocol-Specific SmartBridge* (or *bridgechain*), achieves inter-blockchain communication, by interconnecting the various chains based on ARK. The *Protocol-Agnostic SmartBridge* achieves communication between blockchains that use different consensus mechanisms.

ARK's public network (or the ARK main blockchain) provides the foundation for other blockchains to issue and read transactions. Forging delegates are the entities that create blocks of transactions, analogous to miners in the Bitcoin blockchain.

The consensus mechanism is a modified version of *Delegated Proof-of-Stake* (DPoS). Holders of the ARK token vote to elect the top 51 delegates, who are randomly chosen to secure the network by validating transactions. By fixing the number of delegators (or forging nodes) at 51, the "ARK main net strikes a balance between decentralization and

performance". The ARK token is also used to pay cross-chain transaction fees, which can be triggered by smart contracts, coded languages such as JavaScript, Go, Java, and C#.

The ARK Contract Execution Services (ACES) has "demonstrated two-way transfers between ARK and Bitcoin, Litecoin, and Ethereum, including issuing smart contracts from ARK to Ethereum, regardless of the underlying protocols". While the ARK project defends cross-blockchain interoperability, ACES is on its inception. ACES can only provide interoperability on an *ad hoc* basis. Connectors have to be implemented to connect ARK to other blockchains. Furthermore, ACES is that it is not entirely decentralized, as intermediary nodes are necessary to achieve interoperability. ARK plans to add several features to its platform[44], such as integrating HLTCs to provide ARK bridgechains atomic swap capabilities. ARK is a proprietary solution – it is not open-source. All ARK blockchains are powered by the ARK platform.

AION was originally an ERC-20 token implemented on Ethereum [201]. Later, it evolved to a PoS blockchain system designed to provide the foundation for "custom blockchain architectures". A token bridge was built to swap tokens from the Ethereum blockchain to the AION blockchain. AION-compliant blockchains communicate through CC-Txs, issued by participating networks and routed by connecting networks. CC-Txs are created and processed on a source blockchain and routed by bridges. Bridges connect participating networks with connecting networks.

Bridges would sign and broadcast CC-Txs upon payment of a fee and the validation by the source network. They would act as observers, reporting state changes via Merkle tree hashes to the communicating network.

AION's Transwarp Conduit[45] is a smart-contract based solution that enables developers to create interchain smart contracts, by listening to the source blockchain contract adapter, and calling the corresponding target blockchain. Users can call such contract, triggering a transwarp conduit node to validate the request. After that, the request is processed by the contract.

The AION project was divided into two distinct brands: the Open Application Network (The OAN)[46] and AION itself. The OAN network is no longer focusing on interoperability; it is an open source public infrastructure for the creation and hosting of "open apps". AION is now the digital asset powering such apps. AION plans to develop the OAN tech stack, as stated by the roadmap[47].

Komodo is a blockchain infrastructure that allows one to create chains pegged to the Komodo blockchain, which is pegged to Bitcoin. Komodo uses delayed Proof of Work to create checkpoints of the Komodo's state that are added to Bitcoin from time to time (a process called notarization). Among other use cases, Komodo-based infrastructure allows atomic swaps, via the AtomicDEX feature [129]. To foster adoption, Komodo promotes liquidity provider nodes, which are trading parties that act as market-makers, by buying and selling cryptocurrencies. Komodo is an open-source composable smart chain platform[48], built on top of Bitcoin and ZCash, which take Merkle tree roots from a smart chain set of blocks and merge them with other Merkle roots, that represent other smart chains. This generates a single Merkle root out of the various Merkle roots, referring to blocks of all smart chains. The mainchain, the KMD ledger, then synchronizes the state of each smart chain, providing interoperability capabilities. This mechanism works similarly to *Delayed Proof of Work* (dPoW). dPoW allows securing a chain with another chain by leveraging a high hash rate (like KMD or even Bitcoin itself). This way, the risk of 51% attacks is reduced.

[44] https://ark.io/roadmap
[45] https://github.com/aionnetwork/transwarp_conduittree/master/aion
[46] https://developer.theoan.com/community
[47] https://medium.com/theoan/2019-q4-foundation-report-b3a38a28d2b1
[48] https://github.com/KomodoPlatform/komodo

Table 8. Comparison between Polkadot, Ethereum 2.0, and Cosmos [78, 160, 218]

	Polkadot	Ethereum 2.0	Cosmos
Model	Sharded, pure-abstract STF	Sharded, fixed-function STF	Bridge-hub
Consensus protocol	GRANDPA/BABE	Serenity	Tendermint
Main Chain	Relay-chain	Beacon Chain	Cosmos Hub
Main Chain State Transition Function	Abstract meta-protocol	Fixed-function	Fixed-function
Finality fault tolerance	33%	33%	33%
Finalization expected latency	6-60 seconds	6-12 minutes	Instant
Horizontal Scaling (sharding)	Yes	Yes	Not available
Governance	Lock-vote; Committees; council	Forks	Coin-vote
BTC Token Support	Two-way peg	Not available	Two-way peg
ETH Token Support	Two-way peg	One-way-peg	Two-way peg
EVM Sidechain bridging	Parity PoA	Not available	Two-way peg

We now compare the Blockchain of Blockchains with highest adoption, Polkadot, and Cosmos. As a baseline, we use *Ethereum 2.0* [76–78], a major upgrade to the current Ethereum public mainnet, to be launched in three phases across 2020-2023. Ethereum 2.0 is an advance in blockchain inteoperability, as it will be composed by shards that interoperate with each other. It features a new execution environment for smart contracts, running on a new virtual machine, eWASM. We compare Polkadot, Ethereum 2.0, and Cosmos in Table 8.

In phase 0, the beacon chain of the Ethereum 2.0 network will be launched, implementing PoS and managing the validator registry. The beacon chain is meant for testing purposes and does not have functionality: Ethereum 1.0 will continue to operate. In phase 1, the old main chain and the beacon chain are merged, resulting in a single consolidated chain. Blockchain sharding techniques are used to raise Ethereum 2.0 throughput. Phase 2 focuses on enabling ether accounts, transactions, smart contract execution, and possibly further interoperability features [80].

Ethereum 2.0 is suitable to serve as a baseline, as its performance in terms of throughput will be close to Blockchain of Blockchains; and furthermore, Ethereum is one of the most popular blockchains regarding dApps and industrial use cases.

Polkadot and Ethereum 2.0 have a different approach to interoperability than Cosmos. Cosmos relies on a bridge-hub architecture, making it challenging to scale; Polkadot and Ethereum 2.0 have a shared-security/sharded approach, thus providing better scalability.

Polkadot and Ethereum 2.0 have block production protocols, BABE and RanDAO + LMD Casper, respectively. Moreover, Polkadot and Ethereum 2.0 have finality sub-protocols, GRANDPA, and Casper FFG. Those protocols have to be implemented to provide sharding functionalities. Polkadot can achieve up to 100 shards while Ethereum 2.0 can support 64 shards. Cosmos do not support horizontal scalability via sharding. However, a shared security layer, similar to Polkadot's, is being idealized. In particular, it would allow a zone to inherit the validator set from another zone, allowing for transaction offload.

On Polkadot, the main chain is the relay-chain, relying on the DOT token. Ethereum's 2.0 main chain is the Beacon chain, using Ether. Cosmos' main chain is the Cosmos Hub, and the token used is ATOM. The main chain state transition

function in Polkadot is an abstract meta protocol relying on web assembly. Cosmos and Ethereum 2.0 utilize fixed functions.

The finality fault tolerance, i.e., the minimum required number of faulty nodes to compromise the network, is one third of the nodes less one) for all solutions, with different latencies. Although those solutions have different finality times, one should note that Polkadot and Ethereum 2.0 rely on a sharding strategy.

Polkadot and Cosmos utilize smart contracts and state transaction functions (provide an interface for smart contract execution [218]). Ethereum 2.0 only supports smart contracts. All solutions have robust governance mechanisms, namely decision making and decision enactment mechanisms (e.g., multicameral governance mechanism with conviction voting in Polkadot, coin-vote signaling in Cosmos). Polkadot has enhanced governance with a tech committee and an on-chain treasury. In Cosmos, validators can vote on behalf of the ATOMs staked to them, although it is possible to ATOM holders directly vote, canceling the staked validators' vote.

Regarding compatibility and bridging, Polkadot and Cosmos have two-way pegs to the Bitcoin and Ethereum networks. Ethereum 2.0 has a one-way peg with Ethereum, in which only Ethereum users can send Ether to Ethereum 2.0. Both Polkadot and Cosmos can communicate with sidechains. Polkadot further implements bridging capabilities, by leveraging substrate, achieving shard compatibility.

E HYBRID CONNECTORS

We now describe some of sidechain solutions we identified in the literature. Table 9 summarizes each solution and aggregates them into the corresponding subcategory. One can assert that from the 14 solutions identified, 3 are trusted relays, 4 are blockchain-agnostic protocols, 4 blockchain of blockchains, and 3 blockchain migrators.

E.1 Trusted Relays

Trusted relays are trusted parties that redirect transactions from a source blockchain to a target blockchain.

Kan et al. introduce a protocol that delivers atomicity and consistency through asset escrow (third-party releasing locked assets under specific conditions) and a three-phase commit [120]. This scheme assumes a trusted party. The authors provide a superficial evaluation, consisting of custom-made blockchains.

Abebe et al. propose a generalized protocol for data transfer, with a particular focus on permissioned networks [1]. They introduce system contracts, a *relay service*, and a communication protocol.

The conceptual mechanisms that achieve interoperability are the relay service and system contracts. The relay service acts on behalf of each blockchain, serving requests from applications using the blockchains. Relay services communicate with each other using protocol buffers, a method of serializing structured data, and require *verification policies* to be satisfied by the requester (by verifying a proof). They are also responsible for translating the network-neutral protocol messages into blockchain-specific transactions on the target blockchain. Although the authors defend that relayers operate with "minimal trust" (as they require verifiable proofs coupled with every request), they are trusted in the sense that they follow the protocol, i.e., do not suffer from Byzantine faults.

System contracts are smart contracts that manage data exposure, such as identity and disclosure of network information. One can consider system contracts to be smart contracts handling infrastructural aspects, being an extension to the business logic encoded in most smart contracts. Moreover, such contracts use access control request policy rules against incoming cross-network requests, and if such information is valid (given an attached verifiable proof), according to a specific verification policy. The generation of proofs based on verification policies, and its subsequent validation, allow for trust distribution regarding cross-network transactions.

Table 9. Comparison of *hybrid connector* solutions

	Reference	Transaction validation	Protocol	Supported Blockchains	Public PoC
Trusted Relays	Montgomery et al., [154]* ✓	Trusted escrow party	Cross-blockchain transactions signed by validator quorum	Private	✓
	Kan et al., [120]	Trusted escrow party	3-phase-commit protocol	–	×
	Abebe et al., [1]	Relay service, verifiable proofs, system smart contracts	System contracts, communication protocol. protocol buffers	Private	×
	Falazi et al., [83]	Centralized Gateway	Smart Contract Invocation Protocol	Private, Public	×
Blockchain-Agnostic Protocols	Hardjono et al. [103]	Blockchain Gateways	–	–	×
	Vo et al., [206]	–	× - but Multi-Protocol Communication is referred	–	×
	Interledger Protocol [209]* ✓	(Trusted) Router	Packet Switching (ILPv4)	Private, Public	✓
	Hyperledger Quilt [112]*	(Trusted) Router	Packet Switching (ILPv4)	Private, Public	✓
Blockchain of Blockchains	Verdian et al, [215]* ✓	BPI, Messaging, Filetering and Ordering layers	Based on posets and order theory	Public	×
	Liu et al., [143]	NSB, ISC	UIP protocol	Public	✓
	Block Collider [116]* ✓	Base tuples	Proof of Distance (PoD)	Public	✓
	Amiri et al., [7]	Blockchain views, internal and external transactions	Hierarchical consensus and one-level consensus	–[1]	×
Blockchain Migrators	Frauenthaler et al., [86]	Enforced by smart contracts	Adapters	Public	✓
	Scheid et al., [189]	Enforced by smart contracts	Adapters	–	×
	Fynn et al., [92]	Enforced by smart contracts	Move Operation	Public	×

✓our description was endorsed
∗ considered grey literature
× lacks implementation or implementation is not public
– Not defined or not applicable
[1] CAPER instance enables cross-aplication transactions

Falazi et al. [83] propose an abstraction layer that provides a uniform interface for external client applications to communicate with blockchains and smart contracts. The proposed protocol, Smart Contract Invocation Protocol (SCIP), exposes a interface with several elements (roles, methods, data, and message format), which can be used by applications to issue transactions against different ledgers. The available request messages include (i) the invocation of a smart

contract function, (ii) the subscription to notifications regarding function invocations or event occurrences, (iii) the unsubscription from live monitoring, and (iv) the querying of past invocations or events.

E.2 Blockchain-Agnostic Protocols

Blockchain-agnostic protocols enable cross-blockchain or cross-chain communication between arbitrary distributed ledger technologies.

Hardjono et al. proposed a model for blockchain interoperability, in the context of the Tradecoin[49] project [103].

Each blockchain is seen as an autonomous system (or routing domain), as a connectivity unit that can scale. Such autonomous systems have a domain-centered control with distributed topology. Entities that execute and validate cross-blockchain transactions are called gateways.

Generally, the conceptual mechanism that underlies the interoperability scheme is the ability of gateways to be autonomous and discoverable. Gateways can then redirect transactions to the corresponding blockchain.

Kan et al. presented a theoretical work on how blockchains can execute cross-chain transactions, via several actors: *validators*, *nominators*, *surveillants*, and *connectors* [120]. Validators verify and forward blocks to the correct destination. Nominators elect validators. Surveillants monitor the blockchain router's behavior. The proposed protocol aims participants to achieve a dynamic equilibrium state, using incentivization (fees awarded to the parties following the protocol). No implementation details are provided.

Hyperledger Quilt is a Java implementation of the Interledger protocol [112]. While Interledger implements connectors, Quilt implements several primitives of the Interledger protocol, namely: interledger addresses, ILPv4[50], payment pointers, ILP-over-HTTP, simple payment setup protocol, and STREAM.

Quilt is an open-source project[51], and it is interoperable with other implementations, such as Interledger Rust[52] and InterledgerJS[53].

Other systems are focused on building cross-blockchain dApps, by organizing blocks that contain a set of transactions belonging to CC-dApps, spread across multiple blockchains. Such system should provide accountability for the parties issuing transactions on the various blockchains, as well as providing a holistic, updated view of each underlying blockchain" (Section 2.3).

Overledger aims to ease the development of decentralized apps on top of different blockchain infrastructures [177, 215]. Interoperability is achieved by using a common interface among ledgers.

Overledger proposes a four-layer approach. The *transaction layer* contains different blockchains, and stores transactions coming from them. While the *messaging layer* retrieves relevant information from the transaction layer, coming from heterogeneous blockchains: transactions from a pool of transactions, metadata, or smart contracts. The *filtering layer and the ordering layer* create connections between messages from the messaging layer. Messages are ordered and filtered according to a specific set of rules (e.g., respecting a schema, containing specific cryptographic signatures). In particular, the filtering layer requires knowledge about all the different blockchains included in Overledger.

Overledger requires a block ordering mechanism to ensure the total ordering of cross-blockchain transactions: the application scans the compatible ledgers' transaction hashes and places them into a *verification block*. Transactions

[49] https://tradecoin.mit.edu/
[50] https://github.com/interledger/rfcs/blob/master/0027-interledger-protocol-4/0027-interledger-protocol-4.md
[51] https://github.com/hyperledger/quilt
[52] http://interledger.rs/
[53] https://github.com/interledgerjs/ilp-connector

in a verification block are modeled as a total poset, in which a binary relationship is used to compare the order of transactions within a block [215].

Overledger achieves blockchain interoperability using a protocol for message-oriented middleware that implements a protocol similar to 2-phase-commit scheme, instead of relying on adapters between a central blockchain and external blockchains, but no details are given.

Block Collider enables smart contract communication among smart contracts located in different chains [116]. The goal is to alleviate the developer's work while building decentralized apps that use several blockchains.

Block Collider unifies the latest blocks on each bridged chain via blocks' *base tuples*: every block references the header of the block from each of the bridged chains. This allows Block Collider to be a decentralized *unifying chain*.

The consensus mechanism for determining the following block head is the proof of distance, a variation of proof of work. Proof of distance uses an algorithm in which a string edit distance scheme is used. In this scheme, the idea is to hash to be filtered within some distance of a reference set. Block Collider is an open-source project[54], and supports various cryptocurrencies, including BTC, ETH, USDT, WAV, LSK, NEO, DAI, and Tether Gold.

E.3 Blockchain Migrators

Blockchain migrators allow an end user to migrate the state of a blockchain to another. Currently, it is only possible to migrate data across blockchains, although moving smart contracts is also predicted [155].

Frauenthaler et al. propose a framework for blockchain interoperability and runtime selection [86]. The framework supports Bitcoin, Ethereum, Ethereum Classic, and Expanse. This framework is app-centric since the user can parameterize the app with functional and nonfunctional requirements. The framework can choose a blockchain at runtime, allowing a blockchain to route transactions to other blockchain, depending on weighted metrics.

Some metrics include the price of writing and reading from a blockchain, the exchange rate between the cryptocurrency supporting a blockchain and the dollar, the average time to mine a block and the degree of decentralization.

Based on such metrics, and their weight, specified by the end-user, the blockchain selection algorithm computes the most appropriate blockchain. According to the authors, switching to another blockchain can help users to save costs and make them benefit from a better infrastructure (e.g., better performance, higher decentralization, better reputation). This solution does not tackle the migration of smart contracts. However, data transfers are possible (i.e., data is copied from the source to the target blockchain). This project is a centralized application ran by the end user. It is open-source[55].

Scheid et al. propose a policy-based agnostic framework that connects, manages, and operates different blockchains [189, 190].

Policies can be defined to optimize costs or performance. If one chooses to minimize costs associated with data storing, the framework chooses the blockchain which has the cheapest cost of writing. Conversely, performance policies can configure the framework to minimize a transaction's confirmation waiting time. The authors include AAA access control, as defined by the OASIS consortium [163], to manage policies.

The platform is blockchain agnostic, but details on supported blockchains are not provided. Although this work is not a functional blockchain migration tool, it allows the flexibility needed for blockchain migrations (see Section 5.3.3).

[54] https://github.com/blockcollider
[55] https://github.com/pf92/blockchain-interop

F USE CASES

Example use cases related to cryptocurrency-related techniques are cross-chain payment channels [15, 116, 145, 196], efficient multi-party swaps [45, 89, 234], point of sales and utility tokens [196], and decentralized exchanges [52, 234]. As a notable use case, we highlight decentralized exchanges [**?**], leveraging HLTC techniques to allow users to exchange assets from different blockchains directly with other users.

Blockchain of Blockchains [12, 130, 201, 227] do implement decentralized exchanges, and predict decentralized banking as use cases. For example, the decentralized exchange Binance [31] utilizes the Cosmos SDK. Blockchain gaming platforms[56], and stablecoins[57] have been implemented with Polkadot. Moreover, Blockchain of Blockchains can stimulate blockchain adoption by enterprises. By using Cosmos, zones can serve as blockchain-backed versions of enterprise systems, whereby services that are traditionally run by an organization or a consortium are instead run as an application blockchain interface on a particular zone. Some authors proposed an IoB approach for a central bank digital currency [202], which could be realized with a blockchain engine solution.

Regarding Hybrid Connectors, we highlight blockchain migrators, as solutions that can reduce the risk for enterprises and individuals when investing in blockchain. By reducing risks, investors can expect a higher return on investment [222]. Hyperledger Cactus, a blockchain interoperability project includes a blockchain migration feature, which allows a consortium of stakeholders operating a blockchain to migrate their assets (data, smart contracts) to another blockchain [155]. Other use cases can be realized: cross-blockchain asset transfer, escrowed sale of data for coins, pegging stable coins to fiat currency or cryptocurrencies, healthcare data sharing with access control lists, integration of existing food traceability solutions, and end-user wallet access control.

More generally, a blockchain of blockchains approach can be leveraged to solve current problems. In [30, 62], the authors argue that accidental failures and security events (in particular internal data breaches) is a problem for the end-user. This problem can be alleviated by creating a "cloud-of-clouds" for extra security and dependability, on top of individual cloud providers that do not offer enough trust. One could argue that one can use a blockchain of blockchains approach to increase the dependability of services, as well as their security.

Collecting, storing, accessing, and processing data is not only a common practice across industries but also essential to their thriving. Often, a use-case has several stakeholders with different needs, who belong to different organizational boundaries. Those stakeholders might have different access rights to data [20, 24]. Thus, developers adapt the features of the blockchain they are using to the (sometimes conflicting) needs of their stakeholders. It is important to underline that developers want flexibility regarding their blockchain choice, as they might want to change it in the future [86]. This particular need is related to the possibility of vendor lock-in, which also happens in cloud environments [123]. The need for this flexibility can be achieved by leveraging blockchain migration or multiple blockchains.

[56] https://xaya.io/
[57] http://bandot.io/

Table 10. IoB and BoB use cases

Use Case \ Categories	Public Connectors	Blockchain of Blockchains	Hybrid Connectors
Decentralized Finance	+	+	+
Cross-blockchain dApps	-	±	+
Blockchain Migration	-	±	+
Enabling Enterprise Business Processes	+	±	±

+ Use case already implemented
± Use case being developed
– Use case not planned