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

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

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 [52, 79]. 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" [52, 63]. 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 [52].

Notwithstanding, grey literature is not often updated (e.g., whitepapers [4, 47, 87, 96]). 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. A caveat of our approach is that grey literature is not, necessarily, quality scientific work, as it is not peer-reviewed [77].

Moreover, in order for our grey literature search to be "systematic, transparent, and reproducible," we adopt recommendations from Mahood et al. [63]. 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,

0360-0300/2021/10-ART168 $15.00
https://doi.org/10.1145/3471140

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

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

(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 from 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 [19, 49, 50, 89]. Although some attempts of classifying blockchain interoperability solutions have been made [14, 19, 75], 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 as follows: (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 [42], (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 [19], i.e., IoB-powered BoB applications. This research question is particularly important since it gives a perspective on whether 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 [42, 43, 59, 72], 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 used 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,[1] "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

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

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 [52]. Google Scholar's coverage is arguably the biggest across other academic search engines for Computer Science [32], and it meets the criteria recommended in guidelines for conducting systematic literature reviews [18, 32]. 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 [32]. 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 in 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" in the title).

Next, we performed a keyword-based search. We limited the scope of queries until the present date of writing, i.e., February 14, 2020, thus covering literature up to the present day. Notwithstanding, we updated this article 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 to restrict 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 obtained 262 results: and chose not to include terms such 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; 494 and 665 results, respectively.

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

A.4 Screening and Eligibility Processes

In this section, we define our methodology for the eligibility criteria. Figure 1 represents an adapted *Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA)* diagram [58], considering all steps of our literature research methodology.

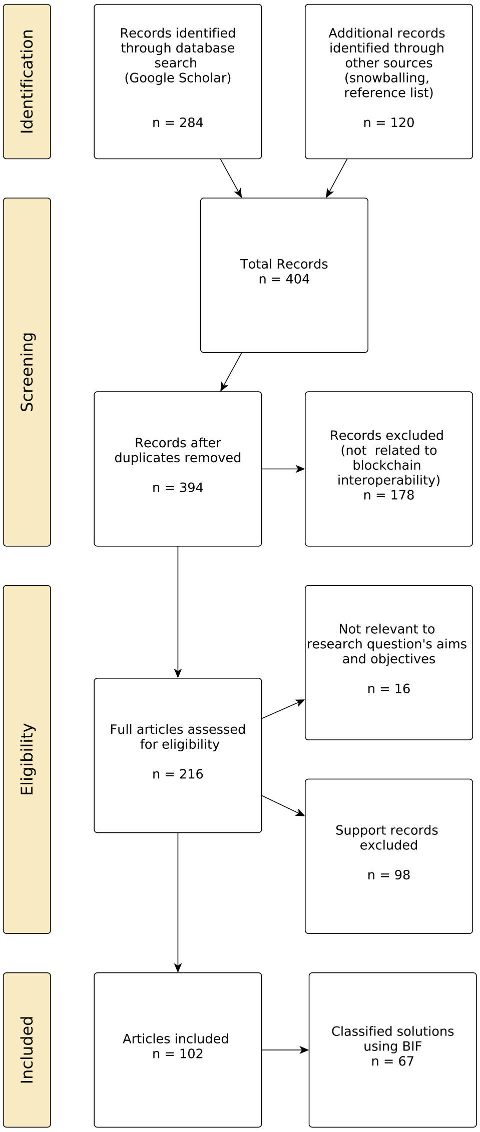
Identification
Screening
Eligibility
Included
Records identified through database search (Google Scholar)
n = 284
Additional records identified through other sources (snowballing, reference list)
n = 120
Total Records
n = 404
Records after duplicates removed
n = 394
Records excluded (not related to blockchain interoperability)
n = 178
Not relevant to research question's aims and objectives
n = 16
Full articles assessed for eligibility
n = 216
Support records excluded
n = 98
Articles included
n = 102
Classified solutions using BIF
n = 67

Fig. 1. 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 to include the document in 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 whose 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, 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 [98]. 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 [68]). 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 and Dai proposed the data, network, consensus, contract, and application layers [48], while Kan et al. proposed the basic, blockchain, multi-chain communication, and application layers [49].

Hardjono et al. proposed an architecture inspired by the architecture of the Internet [42]. 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) [89]. 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 [42]. The decentralized registry would act as a (preferably) decentralized domain name system [65], 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 2 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 the address of a node belonging to Blockchain$_C$ (C), Bitcoin (step 2). CCCP and CBCP protocols can provide unilateral or bidirectional interoperability. In

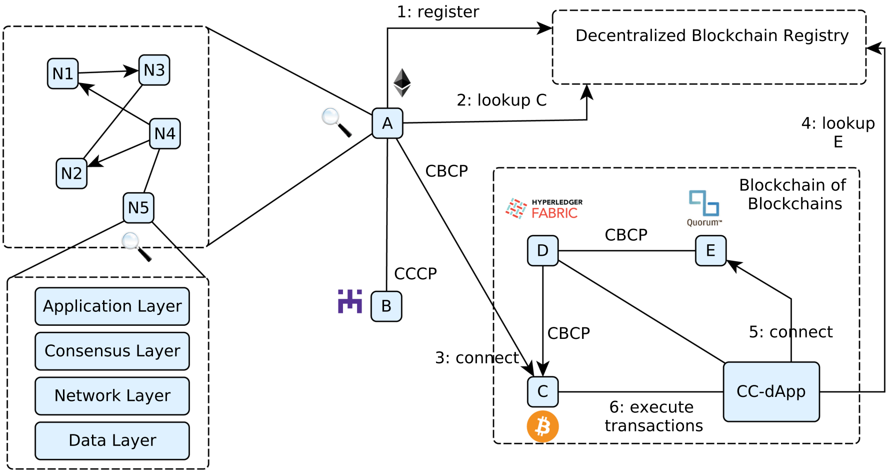

Fig. 2. Architecture for Interoperable Blockchains: a network comprised of five blockchains (A–E) and a **cross-chain decentralized application** (**CC-dApp**).

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 [30]), 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 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 [97]) and CBCPs (e.g., inter-blockchain protocol [45] or the Interledger Protocol [46]) 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 one to refactor several layers: the network, consensus, contract, and application layers [98] would need to be changed.

In Figure 3, we model the layers of blockchain interoperability that correspond to the proposed architecture, using the Archimate modeling language [91], 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" possess (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 [24]. Transaction *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 six 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,

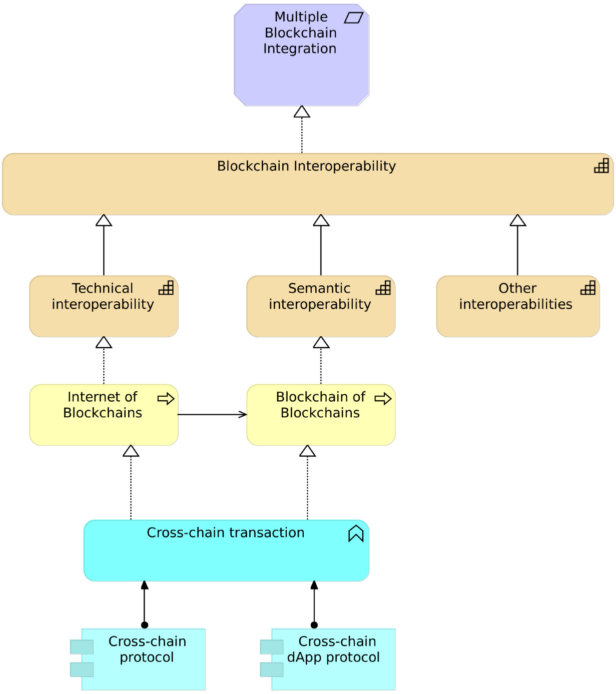
Multiple Blockchain Integration
Blockchain Interoperability
Technical interoperability
Semantic interoperability
Other interoperabilities
Internet of Blockchains
Blockchain of Blockchains
Cross-chain transaction
Cross-chain protocol
Cross-chain dApp protocol

Fig. 3. Simplified blockchain interoperability model, represented in Archimate.

as now it can handle more complex operations. The application can now expose APIs to dispatch cross-blockchain transactions, as illustrated in some works [60, 67, 92]. 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 toward interoperability within the blockchain space.

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

C PUBLIC CONNECTORS

C.1 Sidechains

We now describe some of sidechain solutions we identified in the literature. Table 1 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 [55]. 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 [38]. 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.

Table 1. Comparison of *Sidechains* Solutions

Reference	Mainchain	Sidechain consensus	Summary	Strong points	Weak points	Roadmap
BTC Relay [30] ✓	Ethereum	×	Ethereum smart contract reading Bitcoin's blockchain	Simple solution relying on verifying block headers	Limited functionality	None
Peace Relay [55]	Ethereum	×	SPV on EVM-based blockchains	Allows two-way pegs	It is expensive to verify Ethereum block headers	None
Testimonium [38]	Ethereum	×	EVM-based blockchains SPV	Effiecient validation	Mainly support EVM-based blockchains	Batch submission of block headers
POA Network [4] ✓	Ethereum	Proof of authority	Applicational interoperability to EVM-based dApps	Inexpensive consensus	Validators confined to one country (geographic concentration)	POA-based stable token
Liquid [6] ✓	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 [61] ✓	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 [41]	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 [56] ✓	Bitcoin	DECOR+	Federated sidechain, in which RBTC is tethered to BTC	Merge mining allows reutilization of work	Relies on PoW, energetically inefficient	Decentralized bridge with Ethereum
Blocknet [23] ✓	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, in its turn, generate certificate proofs for the mainchain.

POA Network encompasses an EVM-based blockchain as well as the POA Bridge [4]. 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*,[2] 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.[3] POA is an open source project.[4]

Elements[5] is a sidechain-capable blockchain platform. *Liquid* is a federated pegged sidechain [6], based on Elements, relying on the concept of *strong federations* [26]. 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 hide and protect cryptographic material, e.g., via limited network access and features that provide tamper evidence [78]. 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.[6] Liquid is an open source project.[7]

[2] https://docs.tokenbridge.net/amb-bridge/about-amb-bridge.
[3] https://www.poa.network/roadmap.
[4] https://github.com/poanetwork.
[5] https://elementsproject.org.
[6] https://blockstream.com/2020/02/10/en-blockstream-2019-review-building-foundations/.
[7] https://github.com/Blockstream?q=liquid&type=&language=.

Loom Network is a dApp platform, which relies on sidechains connected to Ethereum, Binance Chain, and Tron [61]. 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 **Proof of Work** (**PoW**) that aims to reduce energy consumption [22]. 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.[8] Loom is open source components.[9]

RSK is a general-purpose smart contract platform pegged to the Bitcoin network that offers improvements in security and scalability of the latter [56], 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 [56]. Moreover, an additional layer of security 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.

RSK uses a 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.[10] RSK is an open source project.[11]

Blocknet is blockchain based on PoS that includes a protocol for interoperability among public and private blockchains [23]. At its core, Blocknet has several components: the XBridge, XRouter, and XCloud [12, 13]. 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.[12]

[8] https://medium.com/loom-network/5183ce02267.

[9] https://github.com/loomnetwork.

[10] https://blog.rsk.co/noticia/hawkclient-building-a-fully-decentralized-bridge-between-rsk-and-ethereum/.

[11] https://github.com/rsksmart.

[12] https://www.cryptocompare.com/exchanges/#/overview.

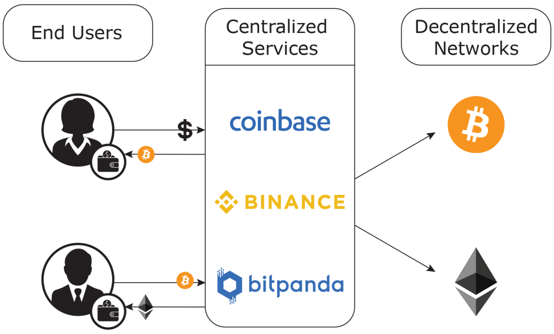

Fig. 4. Alice and Bob buy cryptocurrencies via a centralized exchange. The assets are held by a custodial wallet.

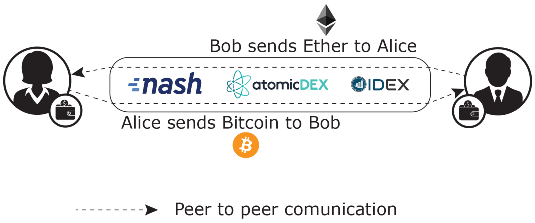

Fig. 5. Alice can send cryptocurrencies directly to Bob, and vice versa. Each user holds their private keys. The exchange is a facilitator of the transactions.

Figure 4 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 [2].

Decentralized exchanges can be implemented with hashed timelocks. Figure 5 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.

Agent Chain is a project aiming to exchange assets between blockchains using a multi-signature scheme [57]. 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

We now describe some of HTLC solutions we identified in the literature. Table 2 summarizes these solutions. Table 3 describes solutions that are based on HTLCs, but have differentiating features, as described on the table.

Black et al. propose the concept of *atomic loans*, based on atomic swaps [11]. 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

Table 2. Comparison of *Hash Lock Time Contract* Solutions

Reference	Supported Chains	Architecture	Summary	Strong points	Weak points	Roadmap
Black et al. [11]	×	Lender, borrower	Leverage HTLC to provide fiat/stablecoinaccess for cryptocurrency holders	Decentralized solutions	Inefficient (atomic swaps); requires over-collateralization	×
Wanchain [62] ✓	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 [74]	Bitcoin	Relies on multi-signature channel addresses	High volume, low latency micropayment enabler	Increases Bitcoin performance, solution in production	Timelock expiration exploits	×
Komodo [53] ✓	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 [20]	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.

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 [62]. 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 toward more general interoperability by promoting cross-chain integration with enterprise blockchains and supporting **Web Assembly** (**WASM**) smart contracts [34].

COMIT is a protocol allowing for atomic swaps, based on HLTCs [20]. 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[13] 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 instance, 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 [25]. 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

[13] https://github.com/comit-network/comit-rs/.

Table 3. Comparison of *Alternative* Solutions

Reference	Main Supported Chains	Architecture	Summary	Strong points	Weak points	Roadmap
Tokrex [64] ✓	×	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 [39] ✓	Ethereum	FUSION distributed control rights services	Distributed storage of a private key and cryptoasset mapping	Distributes trust and responsibility of managing private keys	Does not provide instant atomic swaps	Decentralized oracle services
Sai and Tipper [80]	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	Behavior of malicious trustee
XClaim [97]	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 [15–17, 86]	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 toward signed PoIs	DeXTT implementation on OmniLayer
XChain [83]	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 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 the following.

Tokrex enables the exchange of cryptocurrencies between different blockchains in a decentralized way, by leveraging the concept of *meta-swap* [64]. 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 [39]. 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 from 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[14] that aims to create the first token system natively supported by multiple blockchains [71]. TAST includes several components explained in a set of documents.

In one of these documents, the authors present *claim-first transactions*, a protocol for decentralized blockchain asset transfers. [15]. 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**) [16], 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 [14], 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 [36, 85] and discuss the requirements for more efficient cross-blockchain token transfers.

[14] https://dsg.tuwien.ac.at/projects/tast/.

In [86], 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, [37], introduces optimizations that reduce such costs. This article 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 [17]. 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* [15], 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 [16, 17]. 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.[15]

XChain includes a three-phase protocol that generalizes atomic cross-chain swaps, in which two entities, the leaders and the followers, exchange assets [83]. 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 the sidechain solutions we identified in the literature. Research on Blockchain of Blockchains required substantial *ad hoc* research, including blog posts, roadmaps, and

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

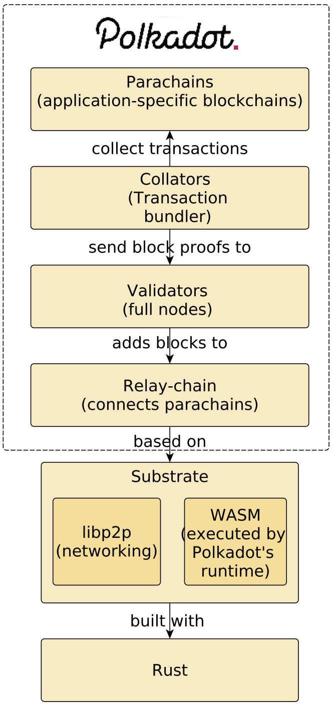

Fig. 6. Polkadot's stack [93, 96].

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 potential slashing, in case of misbehaving. Fishermen get bounties for reporting validators' misbehavior, such as helping to ratify an invalid block.

Figure 6 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 [93]. 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.

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).

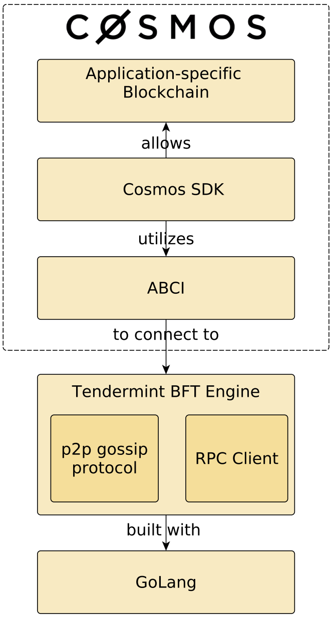

Fig. 7. Cosmos Network's stack [54].

Polkadot's relay chain achieves consensus using BABE and GRANDPA [73]. 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.[16]

Cosmos is another popular Blockchain of Blockchains. Figure 7 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.

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.

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

ARK utilizes smart bridges to make instances of its platform interoperable [5]. 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, and 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,[17] 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 [87]. 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[18] 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**)[19] 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.[20]

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).

[17] https://ark.io/roadmap.

[18] https://github.com/aionnetwork/transwarp_conduit\/tree/master/aion.

[19] https://developer.theoan.com/community.

[20] https://medium.com/theoan/2019-q4-foundation-report-b3a38a28d2b1.

Among other use cases, Komodo-based infrastructure allows atomic swaps, via the AtomicDEX feature [53]. 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,[21] built on top of Bitcoin and ZCash, which takes Merkle tree roots from a smart chain set of blocks and merges 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.

We now compare the Blockchain of Blockchains with highest adoption, Polkadot, and Cosmos. As a baseline, we use *Ethereum 2.0* [27–29], 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 4.

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 [31].

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 [93]). Ethereum 2.0 only supports smart contracts. All solutions have robust governance mechanisms, namely, decision making and decision enactment

[21] https://github.com/KomodoPlatform/komodo.

Table 4. Comparison between Polkadot, Ethereum 2.0, and Cosmos [29, 69, 93]

	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

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 for ATOM holders to 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 5 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 [49]. 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.

Table 5. Comparison of *Hybrid Connector* Solutions

	Reference	Transaction Validation	Protocol	Supported Blockchains	Public PoC
Trusted Relays	Montgomery et al. [66]* ✓	Trusted escrow party	Cross-blockchain transactions signed by validator quorum	Private	✓
	Kan et al. [49]	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., [33]	Centralized Gateway	Smart Contract Invocation Protocol	Private, Public	×
Blockchain-Agnostic Protocols	Hardjono et al. [42]	Blockchain Gateways	–	–	×
	Vo et al. [89]	–	×—but Multi-Protocol Communication is referred	–	×
	Interledger Protocol [90]* ✓	(Trusted) Router	Packet Switching (ILPv4)	Private, Public	✓
	Hyperledger Quilt [44]*	(Trusted) Router	Packet Switching (ILPv4)	Private, Public	✓
Blockchain of Blockchains	Verdian et al. [92]* ✓	BPI, Messaging, Filtering and Ordering layers	Based on posets and order theory	Public	×
	Liu et al. [60]	NSB, ISC	UIP protocol	Public	✓
	Block Collider [47]* ✓	Base tuples	Proof of Distance (PoD)	Public	✓
	Amiri et al. [3]	Blockchain views, internal and external transactions	Hierarchical consensus and one-level consensus	–[1]	×
Blockchain Migrators	Frauenthaler et al. [35]	Enforced by smart contracts	Adapters	Public	✓
	Scheid et al. [81]	Enforced by smart contracts	Adapters	–	×
	Fynn et al.[40]	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-application transactions.

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.

Falazi et al. [33] 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 an 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[22] project [42].

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* [49]. 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 [44]. While Interledger implements connectors, Quilt implements several primitives of the Interledger protocol, namely, interledger addresses, ILPv4[23], payment pointers, ILP-over-HTTP, simple payment setup protocol, and STREAM.

Quilt is an open source project,[24] and it is interoperable with other implementations, such as Interledger Rust[25] and InterledgerJS.[26]

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."

Overledger aims to ease the development of decentralized apps on top of different blockchain infrastructures [76, 92]. 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*

[22] https://tradecoin.mit.edu/.
[23] https://github.com/interledger/rfcs/blob/master/0027-interledger-protocol-4/0027-interledger-protocol-4.md.
[24] https://github.com/hyperledger/quilt.
[25] http://interledger.rs/.
[26] https://github.com/interledgerjs/ilp-connector.

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 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 [92].

Overledger achieves blockchain interoperability using a protocol for message-oriented middleware that implements a protocol similar to the 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 [47]. 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 PoW. 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,[27] 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 [67].

Frauenthaler et al. propose a framework for blockchain interoperability and runtime selection [35]. 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.[28]

Scheid et al. propose a policy-based agnostic framework that connects, manages, and operates different blockchains [81, 82].

[27] https://github.com/blockcollider.

[28] https://github.com/pf92/blockchain-interop.

Table 6. IoB and BoB Use Cases

Categories / Use Case	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.

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 [70], 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.

F USE CASES

Example use cases related to cryptocurrency-related techniques are cross-chain payment channels [6, 47, 62, 84], efficient multi-party swaps [17, 38, 97], point of sales and utility tokens [84], and decentralized exchanges [19, 97]. As a notable use case, we highlight decentralized exchanges [95], leveraging HLTC techniques to allow users to exchange assets from different blockchains directly with other users.

Blockchain of Blockchains [5, 54, 87, 96] do implement decentralized exchanges, and predict decentralized banking as use cases. For example, the decentralized exchange Binance [10] utilizes the Cosmos SDK. Blockchain gaming platforms[29] and stablecoins[30] 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 [88], which could be realized with a blockchain engine solution.

[29] https://xaya.io/.
[30] http://bandot.io/.

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 investments [94]. 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 [67]. 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 [9, 21], 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 [7, 8]. 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 [35]. This particular need is related to the possibility of vendor lock-in, which also happens in cloud environments [51]. The need for this flexibility can be achieved by leveraging blockchain migration or multiple blockchains.

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] 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 of the 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing*. Institute of Electrical and Electronics Engineers Inc., 379–384.

[3] 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.

[4] V. Arasev. 2017. *POA Network Whitepaper*. Technical Report. POA Network. Retrieved on 24 March, 2020 from https://www.poa.network/for-users/whitepaper.

[5] ARK. 2019. ARK Whitepaper Version 2.1.0. Retrieved on 24 March, 2020 from https://whitepaper.ark.io/prologue.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

[11] Matthew Black, Tingwei Liu, and Tony Cai. 2019. Atomic Loans: Cryptocurrency Debt Instruments. Retrieved on 24 March, 2020 from http://arxiv.org/abs/1901.05117.

[12] Blocknet. 2019. Blocknet Documentation. Retrieved on 24 March, 2020 from https://docs.blocknet.co/#technical-overview.

[13] Blocknet. 2019. Blocknet Protocol - XBridge Asset Compatibility. Retrieved on 24 March, 2020 from https://docs.blocknet.co/protocol/xbridge/compatibility/.

[14] Michael Borkowski, Philipp Frauenthaler, Marten Sigwart, Taneli Hukkinen, Oskar Hladky, and Stefan Schulte. 2019. Cross-Blockchain Technologies: Review, State of the Art, and Outlook. Retrieved on 24 March, 2020 from https://dsg.tuwien.ac.at/projects/tast/pub/tast-white-paper-4.pdf.

[15] Michael Borkowski, Christoph Ritzer, Daniel Mcdonald, and Stefan Schulte. 2018. Caught in Chains: Claim-First Transactions for Cross-Blockchain Asset Transfers. Retrieved on 24 March, 2020 from http://www.infosys.tuwien.ac.at/tast/.

[16] Michael Borkowski, Christoph Ritzer, and Stefan Schulte. 2018. Deterministic Witnesses for Claim-First Transactions. Retrieved on 24 March, 2020 from https://dsg.tuwien.ac.at/projects/tast/pub/.

[17] Michael Borkowski, Marten Sigwart, Philipp Frauenthaler, Taneli Hukkinen, and Stefan Schulte. 2019. DeXTT: Deterministic cross-blockchain token transfers. *IEEE Access* 7 (Aug. 2019), 111030–111042.

[18] 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 (April 2007), 571–583. https://doi.org/10.1016/j.jss.2006.07.009

[19] Vitalik Buterin. 2016. *R3 Report - Chain Interoperability.* Technical Report. R3 Corda. Retrieved on 24 March, 2020 from https://www.r3.com/wp-content/uploads/2017/06/chain_interoperability_r3.pdf.

[20] K. Sai and D. Tipper. 2019. Disincentivizing double spend attacks across interoperable blockchains. In *Proceedings of the First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA'19).* 36–45. DOI: 10.1109/TPS-ISA48467.2019.00014

[21] Miguel Correia. 2013. Clouds-of-clouds for dependability and security: Geo-replication meets the cloud. In *Euro-Par 2013: Parallel Processing Workshops*. Springer, Berlin, 95–104.

[22] Miguel Correia. 2019. From Byzantine consensus to blockchain consensus. *Essentials of Blockchain Technology* (2019), 41.

[23] Arylyn Culwick and Dan Metcalf. 2018. *Blocknet Design Specification v1.0.* Technical Report. Blocknet. Retrieved on 24 March, 2020 from https://www.blocknet.co/wp-content/uploads/whitepaper/Blocknet_Whitepaper.pdf.

[24] 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-Dec. IEEE Computer Society, 183–192.

[25] 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*, Lecture Notes in Computer Science, Vol. 10973. Springer-Verlag, 144–151.

[26] 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.

[27] Ben Edgington. 2020. What's New in Ethereum 2. Retrieved on 15 March, 2020 from https://notes.ethereum.org/@ChihChengLiang/Sk8Zs--CQ/https%3A%2F%2Fhackmd.io%2F%40benjaminion%2Fwnie2_200320?type=book.

[28] Ethereum Foundation. 2019. ETH 2 Phase 2 WIKI. Retrieved on 15 March, 2020 from https://hackmd.io/UzysWse1Th240HELswKqVA?view.

[29] Ethereum Foundation. 2019. Ethereum 2.0 Specifications. Retrieved on 15 March, 2020 from https://github.com/ethereum/eth2.0-specs.

[30] Ethereum Foundation and Consensys. 2015. BTC-relay: Ethereum Contract for Bitcoin SPV. Retrieved on 15 March, 2020 from https://github.com/ethereum/btcrelay.

[31] EthHub. 2020. Ethereum 2.0 Phases - EthHub. Retrieved on 15 March, 2020 from https://docs.ethhub.io/ethereum-roadmap/ethereum-2.0/eth-2.0-phases/#eth2-the-new-ether.

[32] 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 (June 2017), 7–47. Retrieved on 15 March, 2020 from https://ejournals.bc.edu/index.php/ital/article/view/9718.

[33] 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*, Lecture Notes in Computer Science, Vol. 12127, Springer-Verlag. 134–149.

[34] Wanchain Foundation. 2019. Wanchain Roadmap. Retrieved on 15 March, 2020 from https://www.wanchain.org/learn/.

[35] 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.

[36] Philipp Frauenthaler, Marten Sigwart, Michael Borkowski, Taneli Hukkinen, and Stefan Schulte. 2019. *Towards Efficient Cross-Blockchain Token Transfers*. Technical Report. https://www.dsg.tuwien.ac.at/tast/tast-white-paper-5.pdf.

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

[38] Philipp Frauenthaler, Marten Sigwart, Christof Spanring, and Stefan Schulte. 2020. Testimonium: A cost-efficient blockchain relay. *arXiv Preprints* arxiv id: 2002.12837. https://arxiv.org/pdf/2002.12837.pdf.

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

[40] Enrique Fynn, Pedone, Fernando, and Bessani Alysson. 2020. Smart contracts on the move. In *Dependable Systems and Networks*.

[41] 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.

[42] Thomas Hardjono, Alexander Lipton, and Alex Pentland. 2019. Toward an interoperability architecture for blockchain autonomous systems. *IEEE Transactions on Engineering Management* 67, 4 (2019), 1298–1309.

[43] Garrick Hileman and Michel Rauchs. 2017. Global blockchain benchmarking study. *SSRN Electronic Journal* (April 2017).

[44] Hyperledger. 2019. Hyperledger Quilt Documentation. Retrieved on 15 March, 2020 from https://wiki.hyperledger.org/display/quilt/Hyperledger+Quilt.

[45] IBC Ecosystem Working Group. 2020. Inter-blockchain Communication Protocol (IBC). Retrieved on 15 March, 2020 from https://github.com/cosmos/ics/tree/master/ibc.

[46] Interledger. 2020. Interledger Protocol V4 (ILPv4) | Interledger. Retrieved on 15 March, 2020 from https://interledger.org/rfcs/0027-interledger-protocol-4/.

[47] Arjun Jain and Patrick Schilz. 2017. *Block Collider Whitepaper*. Technical Report. Retrieved on 24 March, 2020 from https://www.blockcollider.org/whitepaper.

[48] 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*.

[49] Luo Kan, Yu Wei, Amjad Hafiz Muhammad, Wang Siyuan, Gao Linchao, and Hu Kai. 2018. A multiple blockchains architecture on inter-blockchain communication. *Proceedings of the 2018 IEEE 18th International Conference on Software Quality, Reliability, and Security Companion (QRS-C'18)*, 139–145.

[50] 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 (Aug. 2018).

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

[52] 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.

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

[54] Jae Kwon and Ethan Buchman. 2016. *Cosmos Whitepaper*. Technical Report. Cosmos Foundation. Retrieved on 26 Februay, 2020 from https://whitepaper.io/document/582/cosmos-whitepaper.

[55] Kyber Network. 2018. Peace Relay. Retrieved on 24 March, 2020 from https://github.com/KyberNetwork/peace-relay.

[56] Sergio Lerner. 2015. *RSK Whitepaper*. Technical Report. RSK. Retrieved on 24 March, 2020 from https://docs.rsk.co/RSK_White_Paper-Overview.pdf.

[57] 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'19)*.

[58] 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.

[59] 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

[60] 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*.

[61] Loom. 2016. Intro to Loom Network | Loom SDK. Retrieved on 24 March, 2020 from https://loomx.io/developers/en/intro-to-loom.html.

[62] 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. Retrieved on 24 March, 2020 from https://wanchain.org/files/Wanchain-Whitepaper-EN-version.pdf.

[63] Quenby Mahood, Dwayne Van Eerd, and Emma Irvin. 2014. Searching for grey literature for systematic reviews: Challenges and benefits. *Research Synthesis Methods* 5, 3 (Sept. 2014), 221–234.

[64] Mayer Christoph, Mai Jesse N., and Tom M. 2017. *Tokrex Whitepaper.* Technical Report. Tokrex. Retrieved on 24 March, 2020 from www.tokrex.org

[65] Paul V. Mockapetris and Kevin J. Dunlap. 1988. Development of the domain name system. In *Symposium Proceedings on Communications Architectures and Protocols (SIGCOMM'88).* Association for Computing Machinery, Inc., New York, New York, 123–133.

[66] 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. Retrieved on 24 March, 2020 from https://github.com/hyperledger/cactus/blob/master/docs/whitepaper/whitepaper.md.

[67] Hart Montgomery, Hugo Borne-Pons, Jonathan Hamilton, Mic Bowman, Peter Somogyvari, Shingo Fujimoto, Takuma Takeuchi, Tracy Kuhrt, and Rafael Belchior. 2020. Hyperledger Cactus Whitepaper. Retrieved on 24 March, 2020 from https://github.com/hyperledger/cactus/blob/master/docs/whitepaper/whitepaper.md.

[68] Satoshi Nakamoto. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved on 15 March, 2020 from http://bitcoin.org/bitcoin.pdf.

[69] Cosmos Network. 2014. Cosmos Blog. Retrieved on 15 March, 2020 from https://blog.cosmos.network/.

[70] OASIS. 2010. Extensible Access Control Markup Language Version 3.0. Retrieved on 15 March, 2020 from https://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cd-03-en.html.

[71] Pantos Team. 2020. *Pantos Vision Paper.* Technical Report. Pantos. Retrieved on 17 May, 2020 from https://pantos.io/pdf/pantos-visionpaper.pdf.

[72] 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

[73] Polkadot. 2019. Polkadot Consensus · Polkadot Wiki. Retrieved on 1 April, 2020 from https://wiki.polkadot.network/docs/en/learn-consensus.

[74] Joseph Poon and Thaddeus Dryja. 2016. *The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments.* Technical Report. Lightning Network. Retrieved on 31 January, 2020 from https://lightning.network/lightning-network-paper.pdf.

[75] 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.

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

[77] Marten Risius and Kai Spohrer. 2017. A blockchain research framework. *Business and Information Systems Engineering* 59, 6 (Dec. 2017), 385–409.

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

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

[80] 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.*

[81] Eder Scheid, Bruno Rodrigues, and Burkhard Stiller. 2019. Toward a policy-based blockchain agnostic framework. In *16th IFIP/IEEE International Symposium on Integrated Network Management (IM'19).*

[82] 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.

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

[84] 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. Retrieved on 31 January, 2020 from https://syscoin.org/zdag_syscoin_whitepaper.pdf.

[85] Marten Sigwart, Philipp Frauenthaler, Taneli Hukkinen, and Stefan Schulte. 2019. *Towards Cross-Blockchain Transaction Verifications.* Technical Report. Retrieved on 15 March, 2020 from http://www.infosys.tuwien.ac.at/tast/.

[86] Marten Sigwart, Philipp Frauenthaler, Christof Spanring, and Stefan Schulte. 2019. *Preparing Simplified Payment Verifications for Cross-Blockchain Token Transfers.* Technical Report. Retrieved on 15 March, 2020 from https://dsg.tuwien.ac.at/projects/tast/.

[87] Matthew Spoke. 2017. *Aion: Enabling the Decentralized Internet*. Technical Report. Retrieved on 15 March, 2020 from https://whitepaper.io/document/31/aion-whitepaper.

[88] He Sun, Hongliang Mao, Xiaomin Bai, Zhidong Chen, Kai Hu, and Wei Yu. 2018. Multi-blockchain model for central bank digital currency. In *2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'17)*, Vol. 2017-Dec. IEEE Computer Society, 360–367. https://doi.org/10.1109/PDCAT.2017.00066

[89] 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), IEEE Green Computing and Communications (GreenCom), IEEE Cyber, Physical and Social Computing (CPSCom), and IEEE Smart Data (SmartData)*. 1574–1581.

[90] Stefan Thomas and Evan Schwartz. 2015. A Protocol for Interledger Payments. 25 pages. Retrieved on 15 March, 2020 from https://interledger.org/interledger.pdf.

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

[92] Gilbert Verdian, Paolo Tasca, Colin Paterson, and Gaetano Mondelli. 2018. *Quant Overledger Whitepaper v0.1*. Technical Report. Quant. 1–48 pages. Retrieved on 12 February, 2020 from http://objects-us-west-1.dream.io/files.quant.network/Quant_Overledger_Whitepaper_v0.1.pdf.

[93] W3F. 2020. Research at W3F. Retrieved on 15 March, 2020 from https://research.web3.foundation/en/latest/polkadot/.

[94] Sheila Warren and David Treat. 2019. Building value with blockchain technology: How to evaluate blockchain's benefits. *White Paper in Word Economic Forum* (2019).

[95] Will Warren and Amir Bandeali. 2017. *0x: An Open Protocol for Decentralized Exchange on the Ethereum Blockchain*. Technical Report.

[96] Gavin Wood. 2016. *Polkadot: Vision for a Heterogeneous Multi-Chain Framework*. Technical Report. 21 pages. Retrieved on 15 March, 2020 from https://github.com/w3f/polkadot-white-paper/raw/master/PolkaDotPaper.pdf.

[97] 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*.

[98] 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. *ACM Computing Surveys* 52, 6 (2019), 120:1–120:34.