How to Design Autonomous Service Level Agreements for 6G

Tooba Faisal, Jose Antonio Ordoñez Lucena, Diego R. Lopez, Chonggang Wang, and Mischa Dohler

ABSTRACT

With the growing demand for network connectivity and diversity of network applications, one primary challenge that network service providers are facing is managing the commitments for Service Level Agreements (SLAs). Service providers typically monitor SLAs for management tasks such as improving their service quality, customer billing and future network planning. Network service customers, on their side, monitor services provided to them, to optimize their network usage and apply, when required, penalties related to service failures. In future 6G networks, critical network applications such as remote surgery and connected vehicles will require these SLAs to be more dynamic, flexible, and automated to match their diverse requirements on network services. Moreover, these SLAs should be transparent to all stakeholders to address the trustworthiness on network services and service providers required by critical applications. Currently, there is no standardized method to immutably record and audit SLAs, leading to challenges in aspects such as SLA enforcement and accountability — traits essential for future network applications. This work explores new requirements for future service contracts, that is, on the evolution of SLAs. Based on those new requirements, we propose an end to end layered SLA architecture leveraging Distributed Ledger Technology (DLT) and smart contracts. Our architecture is inheritable by an existing telco-application layered architectural frameworks to support future SLAs. We also discuss some limitations of DLT and smart contracts and provide several directions of future studies.

INTRODUCTION

A Service Level Agreement (SLA) is a contract or agreement between a service provider and a customer. The customer can be an organization or an individual. It is a legal and detailed document that typically includes roles and responsibilities of all parties involved such as service quality, duration, and penalties [1]. Typically, network service providers (e.g., mobile service provider or infrastructure providers) have set procedures in place to monitor their SLAs and generate periodic reports. Such mechanisms allow them to monitor their service quality and keep records in the situations of customer complaints or disputes. A generic and high level contractual system is depicted in Fig. 1.

In the 6G networks, many of the existing 5G networks’ applications will evolve to be applicable in much more demanding environments. Applications such as remote surgery and extreme uses of the Industry 4.0 paradigms will become prevalent. We argue that, due to the nature and the criticality of these services current SLA methods will not be viable. Currently, SLAs are typically based on long term commitments between the service providers. The contracts, that is, SLAs, are agreed in months in advance and may last for years. It is obviously difficult to promise on services and predict the service metrics (e.g., service quality) in advance for operators.

From the business perspective, in the current networks’ applications, typically, there are one or two key players involved. For example, in an autonomous car plant, private 5G will be provided by a vendor. This will change in 6G networks, when to meet the demand of services, several vendors and service providers will be involved in an end-to-end service provisioning. This means they will have several agreements in place to fulfill their dealings.

Indeed, this calls for an overhaul of the traditional, manual and opaque contract management systems to a reliable contractual mechanism for future network’s applications in 6G. We note two key requirements must be fulfilled for future SLAs. Firstly, the SLAs must be reliable and trustable. This means that all the stakeholders should trust the contractual mechanism, and that the SLAs shall be honored in all circumstances. Secondly, they must be automatically managed throughout their lifetime, with a special focus on SLA monitoring and enforcement stages.

To that end, we propose the use of DLT, particularly Permissioned Distributed Ledgers (PDLs), to design a contractual mechanism for SLAs in future networks. Distributed Ledgers are immutable and transparent network of nodes, in which all the participants keep an identical copy of the record. New transaction requests are accepted and included by the PDL participants (commonly known as nodes) to the ledger, after the consensus of the ledger is reached. Depending on the factors such as requirements and available resources, PDL consensus is chosen by the founding members of the PDL. The planned executions are recorded in the ledgers in the form of executable software code called “Smart Contracts.” Typically, smart contracts are installed on distribu-
ed ledgers and consequently inherit some of the distributed ledgers’ properties such as immutability and automated execution. Some properties of smart contracts are briefly described below:

**Immutable** — Smart contracts are immutable, when they are installed on a ledger, they cannot be amended or deleted.

**Auto-executable** — When a smart contract is installed on a ledger it becomes auto-executable.

**Transparent** — Typically, a smart contract is installed on all nodes of a distributed ledger system. Therefore, a smart contract becomes visible to all the nodes.

From lifecycle perspective, we believe smart contracts inherently resemble SLAs as illustrated in Fig. 2. Authors in [2] proposed an SLA processing lifecycle and proposed three high level stages: creation phase, an operation phase, and a termination phase. Smart contracts’ life cycle defined in [3] also divides smart contracts in three similar phases, that is, initialization/creation, execution/logic, and termination. Indeed, a smart contract is in fact a software code and can be applied to a number of use cases. However, the focus of this work is to exploit the properties of smart contracts and use them as SLAs for 6G.

Our contribution primarily includes:
1. Novel SLA requirements for future network
2. This new SLA architectural framework is proposed, which utilizes smart contracts to automate the entire multi-layer SLA process.
3. Several future directions are identified and detailed.

The rest of this article is organized as follows: Firstly, we review the existing work. Next, we highlight the recommendations about SLA for 6G. We propose a novel DLT-enabled SLA architecture. We discuss the considerations while designing a DLT enabled architecture. Then, a few future directions are discussed. Lastly, we conclude our work.

**RELATED WORK**

Smart contracts as SLAs have been in discussion for a while, and several applications have been proposed. For instance, in [4] authors propose the use of DLT in the design of an SLA management system for cloud-based services. This work discusses advantages and insights of using DLT in SLA process with a high level DLT-enabled architecture for cloud services with some details on the contents of SLAs. However, it is limited to smart contract modularization for SLA in the cloud scenario. Also, they do not discuss the pitfalls of the distributed ledgers and the considerations for their adoption in the architecture.

A general discussion on requirements and security challenges in 6G networks and potential of blockchain/smart contracts in future networks is presented by [5]. The authors highlight some of the points discussed in our article, such as auditability and SLA management through blockchain. However, this article is limited to discussion on overall 6G requirements and challenges with the scope of blockchain in future networks and does not present a DLT-enabled SLA architecture.

Another approach for SLA management in the context of Quality of Experience (QoE) in 6G is presented by [6]. In this work, authors propose a blockchain-based platform for SLA management between the users and the service providers along with a QoE model and dynamic resource selection in 6G. This work is focused on the overall architectures and methodologies and does not discuss SLA architecture in detail nor provide any insights of smart contracts.

A smart contract together with an SLA model is represented by [7] where the authors propose a witness committee based model to monitor the SLA; the system incentivizes committee members for being honest. However, they neither discuss the structure of SLA nor challenges related to adopting a smart contract as SLA.

Authors in [8] propose a model for translating SLAs into a smart contract; the proposed model relies on a third party to collect the monitoring data and adjust the metrics (e.g., billing and payment) accordingly. Like other existing works, the work from [8] is also focused on smart contract implementation for SLAs, but neither discusses the limitations of smart contracts nor the requirements for future networks.

**SLAs in 6G**

**Requirements**

6G networks will introduce a new wave of devices, applications and use cases. In this section, we identify the key requirements for future SLAs. Even though we aim to address the requirements of 6G, the requirements presented in this section, apply to current 5G applications as well.

**Monitorable** — The future SLAs must be monitorable during their lifetime, that is, from its initialization to completion; Fig. 2a.

**Transparent** — A future network SLA will need to be transparently available to all the contract stakeholders.

**Auto-Executable** — Future network SLAs will need to be auto-executable. That is, they must be executable with the pre-programmed conditions without human intervention.

Indeed, humans will still play key roles in the contracts, but their role will be shifted to contract management tasks (e.g., planning, designing and maintenance) of the SLA software.

**Dynamic** — SLAs will need to be dynamically generated and in real-time as per user demand.
Zero Touch Network and Service Management
— SLAs should be actively monitoring the system and report any anomalies in the system autonomously.

AI-Enabled — Future generation of contractual systems must have the cognitive ability to make human-like decisions in unforeseen situations like network anomalies and participants' disputes.

Intent-Enabled — As per Internet Engineering Task Force (IETF) definition: “An intent specifies the goals and outcomes of the network without specifying how to achieve them.” Intent-Enabled SLAs would allow customers to specify their connection requirements without any network-specific terms. Therefore, customers can request services tailored for their requirements.

Cross-Provider Collaboration — Cross-providers collaboration is already a common practice (e.g., RAN sharing, roaming) to allow operators to extend their service footprint at reasonable costs.

RELIABLE AND TRUSTABLE SLAS
Considering the requirements highlighted in the earlier section, we argue that a reliable and trustable contractual mechanism is essential for the viability of 6G. We note that 6G requirements defined earlier align with the properties of DLT. Contracts (i.e., SLAs) managed and maintained with DLT will enable reliability and trust in the contractual process in the future networks of 6G.

ARCHITECTURE
Based on the requirements identified earlier, in this section, we present our contractual system for future networks of 6G. Our architecture exploits PDL to enable an automated and transparent contractual tool. As a use case, we position the modularized SLA components in a typical service providers’ network slicing architecture [9]. The components (Fig. 3) are placed in a complete telco stack, that is, in the BSS (Business Support System), OSS (Operation Support System) and network layers. The aim of our architecture is to modularize the SLA architecture to enable portability and decentralization. As seen in Fig. 4, the SLA components are modularized into sub-functions; which organizes the contract development process and makes it ready for the DLT-based contractual solutions. In Fig. 3, we place these components in an example of network slicing [9], to explain the notion that these components will be executed at different layers of the service provisioning. However, network slicing is a use case only and our architecture is applicable to a wide scale of telco-applications.

As a use case, we position the modularized SLA components in a typical service providers’ network slicing architecture [9]. The components are placed in a complete telco stack, that is, in the BSS (Business Support System), OSS (Operation Support System) and network layers.

FIGURE 3. Our novel DLT-Enabled contract architecture. The processes are modularised to enable portability in contractual process.
In some cases, it may be required to negotiate the SLAs, for example, if the service provider wishes to provision their services through an auction or when a customer needs unique services. P-CNf will provide negotiation functionalities such as bids and customized pricing in such a situation. The governance of the PDL controls P-CNf, but the service provider will negotiate the SLA.

**PDL Access Control Function (P-ACF):** In the next step, internal PDL governance will assign access credentials to the customers through P-ACF and as per the agreed SLA. The access control will be recorded to the ledger through AC-SC for a limited and SLA-agreed time.

**PDL Service Orchestration Function (P-SOF):** Allocates the resources as per the agreed SLA. Once the services are orchestrated, the P-CIF initializes the smart contract (i.e., SLA), and the services are provided. P-CIF also initializes a timer within the contract to keep track of service provisioning duration.

**PDL Inter-Operability Function (P-IoF):** We advocated a single multi-operator and shared architecture and argued on the viability of service providers’ collaboration. However, several networks are likely to be involved in some service provisioning, for example, in a roaming scenario. Yet, distributed ledgers have a significant consideration of interoperability. That is, not every DLT protocol is compatible with other DLT protocols. To this end, P-IoF will ensure that appropriate functionalities are implemented to enable interoperability between two service providers. The PDL interoperability strategies and algorithms are a research area on their own, and we leave this for future work.

**Functional Stratum**

**PDL Function (PDLF):** Provides the PDL functionalities, that is, recording the data to the PDL and running the consensus mechanism. Every operator-managed resource, including network functions and OSS assets, is equipped with a PDLF (Fig. 4). However, the PDL functionalities are not always required to be active and are only activated when needed.

A PDLF will have two major functions 1) record the data to the PDL, and 2) generate and send periodic reports to the intra-PDL governance for future auditability. Note that PDLF can be installed on any layer, where PDL functionality is required and will perform these two functions regardless.

**PDL SLA Management Function (P-SMF):** Monitors the SLA and collects the insights of SLA execution and checks for anomalies. In case any misbehavior (i.e., SLA violation) is identified, the P-SMF will immediately send a report to the intra-PDL governance, and the smart contract will be interrupted immediately through control instructions [3].

**Termination Stratum**

Once the service contract is completed, and SLA is finalized, the smart contract must be deactivated properly [3]. The complete and safe termination of a contract is the responsibility of the Termination Stratum and has the following functions:

**PDL Contract Termination Function (P-CTF):** All the smart contract variables are cleared at this stage. The access rights are automatically revoked by the P-ACF, however, the governance will ensure that access rights are revoked and the smart contract is no longer accessible by any stakeholder.
PDL Final Report Function (P-FRF): The system will also generate a final report. This report will include SLA lifecycle parameters such as the SLA start and end times, proof of successful deactivation and details of the involved parties. The governance of PDL, will keep these reports for the audit purposes.

**Considerations**

This section highlights the considerations necessary for adopting smart contracts as SLAs.

**SLA Enforcement**

Distributed ledgers are a network of worldwide set of distributed nodes. When an SLA is installed on the ledger and the nodes are distributed in a global footprint, different jurisdiction rules apply to the node and smart contract enforcement across the national borders is still a challenge.

**SLA Compositability**

An SLA has horizontal and vertical components [10]. The horizontal SLAs represents an agreement whereby provider and customer roles correspond to actors from the same layer, e.g., contract between two network service providers. The vertical SLA represents an agreement whereby provider and customer role corresponds to actors from different layers e.g., the contract between the vendor and service provider. These horizontal and vertical components are chained together as functions to form an end-to-end SLA. During service provisioning, one SLA function would often call other SLA functions.

Indeed, smart contracts can automate this end-to-end process. However, they are auto-executable. This means an uncontrolled or unmanaged function can initiate a chain of authorized and unauthorized executions.

To enable security in smart contracts, they should be coded so that they are allowed to access other functions with a comprehensive security framework including strict access-control [3].

**AI-Enabled Smart Contracts**

As discussed earlier, future SLAs may need to be AI-enabled. Yet, computations required for AI-based systems would require compatible hardware and software. Generally, the distributed ledger nodes’ processing resources are dominated by the node management tasks such as transaction processing. One of the possible solution is to offload computational intensive task to an external system [3]. With modularization, a smart contract can be divided into several small modules, and computation-intensive parts can be installed on appropriate hardware.

**Inherent Properties**

**Immutability:** Despite its advantages, immutability has risks, most prominently, security risk. Because smart contracts are unamendable, they cannot be updated or patched and will stay accessible by every member of the ledger. If a term in an SLA is wrongly coded, it will stay in the ledger forever. The possible solutions to these problems are careful planning and Termination Function [3]. Note that the termination functions cannot remove a smart contract from the ledger but will clear all

the variables and deactivate it, therefore scalability challenges remain. Solutions such as off-chain storage [3, 11] are proposed by the research community to solve scalability challenges.

**Scalability:** Immutability also leads to scalability limitation, that is, if a contract cannot be deleted or amended, erroneous and expired contracts will stay in the ledger forever. This problem is particularly severe in the situations of zero-touch networks where the nodes will need to maintain a large amount of data such forwarding and routing information for the decisions and future audit. Scalability in distributed ledgers can be improved through techniques such as offchain and sidechain storage [3]. In such techniques, the mainchain is used only to save the major details such as pointers to external storage. The remaining resource intensive data is stored in external storage (i.e., offchain) or in a sidechain (i.e., sidechain).

**Transparency:** In distributed ledgers, every node keeps a copy of the ledger, and all the transactions and smart contracts are replicated across the network. This is problematic in the scenario when a number of competitors are in the ledger working as a network. Techniques such as Hyperledger Fabric’s Channels [12] allows the PDL users to create private channels of communication within a ledger network.

**Data Inputs**

Data in smart contracts is entered automatically, which means that it is injected through either another smart contract or an external oracle. The whole process is automated, in the sense that there is no way to pick up if the internal or external sources are entering the wrong data in the smart contracts.

Secure data feed proposals such as Town Crier [13] can be used but the limitations and requirements of these mechanisms should be considered. Another option is an internal oracle service as proposed in [3]; whereby the service is managed and maintained by the distributed ledger network and overseen by the governance.

**Interoperability**

An SLA generally involves two or more organizations. If two organizations operate different ledger types, the apparent challenge is interoperability between the ledger types due to factors such as varied consensus algorithms and access control mechanisms.

Strategies such as Notary schemes, Relay schemes or Hashlocking [14] are discussed in the literature and can be adapted to enable interoperability between distributed ledgers. However, network designers should be careful to guarantee that both ledgers are synchronized in a timely manner so that SLA integrity and enforcement is not affected.

**Security**

Security is of paramount importance for SLAs; if an SLA is tampered with or executed with inaccurate data, it can cause monetary losses such as payment to illegitimate parties. Particular care must be taken when SLAs are coded as smart contracts because smart contracts are vulnerable to several security problems such as their inherent properties and hardware security challenges. Hardware or physical layer attacks, for instance, Man-in-the-Middle Attack, can be protected by holding the device owner accountable for any wrongdoings. The gov-
Future Directions & Challenges

Open and Distributed Networks

Future 6G networks will be more open and agile. This means that:
1. Network functions will be more plug-and-play, thanks to the open interface
2. Players will be able to join the network or leave the network without any cumbersome process.

Typically, distributed ledgers require a certain number of nodes to run the consensus, and due to agility of the nodes, the PDL will not reach to a consensus.

A potential solution is to design an appropriate consensus mechanism for the PDL and a strong governance model with a governance body (e.g., Ofcom) enforcing and ensuring a threshold number of nodes stay in the system.

Integration of Non-Public (Private) and Public Network

Private 5G networks, i.e., non-public 5G networks, are getting attention. By providing authority over wireless coverage and capacity, the private 5G network market ensures guaranteed and secured connectivity.

The midterm roll-out of private 5G services will build on Public Network Integrated Non-Public Network (PNI-NPNs). In this modeality, the private 5G network is provisioned with the support of public network (PLMN) resources. The role of PLMN here is to provide service continuity for those cases where the user moves out of private network coverage. For this hybrid (private-public) model, it is needed to formulate the SLA as a weighted composition of NPN’s SLA and PLMN’s SLA, with PLMN’s SLA set based on negotiation between the mobile network operator and the customer.

In future 6G networks, integration and interoperability between private and public networks will happen at a much wider scale, connecting together networks of different nature and scale, thereby realizing the so-called network of networks.

In these scenarios, SLAs will be defined as a composition of a number of fine-grained, context-aware SLAs, with much more variants that shall be managed appropriately.

Performance Evaluation

In a related study [15], authors demonstrated that smart contracts in the context of SLAs add only milliseconds of overhead. These results are promising and we aim to extend these findings and evaluate our proposed architecture in several telco settings.

Conclusion

This work highlights the requirements of the SLAs for future 6G networks. Based on the identified requirements, we argued that smart contracts are the key to future SLAs. They pose the essential properties for future SLAs, and on this notion, we proposed a modularized, PDL-enabled SLA architecture for future networks of 6G. Like every technology, distributed ledgers have limitations, which may hinder their adoption as SLAs. We discussed these limitations in detail and argued that these limitations could be managed and mitigated through comprehensive planning and standardization. We believe this work will mark a new era of SLAs which are trustable by all the stakeholders.

References

[1] L.-J. Jin, V. Machiraju, and A. Sahai, “Analysis on Service Level Agreement of Web Services,” HP, vol. 19, 2002, pp. 1–13.
[2] L. Wu and R. Buyya, “Service Level Agreement in Utility Computing Systems,” Performance and Dependability in Service Computing: Concepts, Techniques and Research Directions, 2012.
[3] “Details of ETSI Work Item: DCG/PDL:0011 Specifications of Requirements of Smart Contracts’ Architecture and Security,” https://bit.ly/3gCiZQk, accessed: 17-11-2021.
[4] W. Tan et al., “A Novel Service Level Agreement Model Using Blockchain and Smart Contract for Cloud Manufacturing in Industry 4.0,” Enterprise Information Systems, 2021, pp. 1–26.
[5] T. Hewa et al., “The Role of Blockchain in 6G Challenges, Opportunities and Research Directions,” 6G SUMMIT, 2020, pp. 1–5.
[6] T. Makosyuky et al., “Blockchain-Based Decentralized Service Provisioning in Local 6G Mobile Networks,” ACM Sensys, 2021, pp. 518–19.
[7] H. Zhou, C. de Laat, and Z. Zhan, “Trustworthy Cloud Service Level Agreement Enforcement with Blockchain Based Smart Contract,” IEEE CloudCom, 2018.
[8] R. B. Uriarte et al., “Distributed Service-Level Agreement Management with Smart Contracts and Blockchain,” Concurrency and Computation: Practice and Experience, vol. 33, no. 14, 2021, p. e5800.
[9] J. Ordonez-Lucena et al., “On the Rollout of Network Slicing in Carrier Networks: A Technology Radar,” Sensors, vol. 21, no. 23, p. 8094, 2021.
[10] E. Marilly, O. Martinot, S. Betge-Brezet, and G. Delegue, “Requirements for Service Level Agreement Management,” IEEE IPOM, 2002, pp. 57–62.
[11] J. Eberhardt and J. Heiss, “OfChaining Models and Approaches to OfChain Computations,” SERIAL, 2018, pp. 7–12.
[12] Hyperledger Fabric, “Channels,” https://bit.ly/3eyaa7X, accessed: 22-12-2021.
[13] T. Zhang et al., “Town Crier: An Authenticated Data Feed for Smart Contracts,” ACM SIGSAC, 2016, pp. 270–82.
[14] T. Koons and E. Poll, “Assessing Interoperability Solutions for Distributed Ledgers,” Pervasive and Mobile Computing, vol. 59, 2019, p. 101079.
[15] T. Faisal et al., “Beat: Blockchain-Enabled Accountable and Transparent Infrastructure Sharing in 6G and Beyond,” IEEE Access, vol. 10, 2022, pp. 48,660–72.

Biographies

TOOBA FAISAL (M) (Tooba.faisal@kcl.ac.uk) received her Ph.D. from KCL and currently working as Research Web3.1 Engineer at Nokia Bell-Labs. Her current research interests are designing secure smart contracts, DLT and 6G networks. She also has a MRes in Security Science from UCL. She is the rapporteur of smart contract ETSI PDL CR and GS.

JOSÉ ORDOÑEZ LUCENA joined Teléfonica I+D in 2018 as a Technology Analyst and Standards Specialist, within the Global CTO Unit. He is currently involved in technology exploration activities for BSG systems, with a special emphasis on end-to-end network slicing solutions and APIs for network capability exposure, considering their joint applicability for public-private network integration scenarios.

DIEGO R. LOPEZ joined Teléfonica I+D in 2011 as a Senior Technology Expert on network middleware and services. He is currently in charge of the Technology Exploration activities within the GCTO Unit of Teléfonica I+D. He is currently focused on identifying and evaluating new opportunities in technologies applicable to network infrastructures.

CHONGGANG WANG (F) is currently a Principal Engineer with InterDigital, Inc. He has more than 20 years of experience in the fields of wireless communications, networking, and computing, including research, development, and standardization. He is the Founding Editor-in-Chief of the IEEE IoT Journal and is currently the Editor-in-Chief of IEEE NETWORK.

MISCHA DOHLER (F) is the Chief Architect in Ericsson and visiting Professor in wireless communications at KCL, driving cross-disciplinary research and innovation in technology, sciences, and arts. He is a Fellow of the Royal Academy of Engineering, the Royal Society of Arts (RSA) and the Institution of Engineering and Technology (IET).