Going Beyond the Blockchain Hype: In Which Cases are Blockchains Useful for IT Applications ?

Nour El Madhoun, Julien Hatin, Emmanuel Bertin

Going Beyond the Blockchain Hype:

In Which Cases are Blockchains Useful for IT Applications ?

Nour El Madhoun*, Julien Hatin†, Emmanuel Bertin†
* LISITE Laboratory, ISEP, 10 Rue de Vanves 92130 Issy-les-Moulineaux, France
†Orange Labs, 42 rue des Coutures BP 6243 14066 Caen, France
Email: nour.el-madhoun@isep.fr; {julien.hatin, emmanuel.bertin}@orange.com

***Abstract*—Blockchain technology is today proposed as a new technical infrastructure for several types of IT applications. This interest is due to its unique property that allows two entities to perform transactions, in a very secure manner, without going through a TTP. However, a blockchain comes along a number of other intrinsic properties, which may not be suitable or beneficial in all the envisaged application cases. Consequently, we propose in this paper a decision tree to identify when a blockchain may be the appropriate technical infrastructure for a given IT application, and when another system (centralized or distributed peer-to-peer) is more adapted. This tree is based on a comparative study between the blockchain and classical "centralized and distributed peer-to-peer" systems.**

***Index Terms*—Application, blockchain, IT, peer-to-peer, permissioned, permissionless, security, TTP.**

I. Introduction

In recent years, blockchain technology is receiving an increasing attention from research and industry in various domains. It is proposed as a new technical infrastructure for several types of Information Technology (IT) applications. It is considered today a new innovation to revolutionize communications in an IT application. This importance is due in fact to its unique property that allows two users to perform transactions without going through a Trusted Third Party (TTP), while offering a transparent and fully protected data storage [1].

By implementing a blockchain infrastructure, a networked distributed ledger is formed where it will be managed between the blockchain users without the intervention of a central TTP: "the intrinsic main property of a blockchain". Each user has a copy of this ledger, can check its validity and participates actively with the other users in its creation and its constant updating (the blockchain users communicate together in a distributed manner without the need they know or trust each other before) [2]. In addition, this ledger stores, in several interconnected blocks, the history of all the exchanges made between the users and constitutes an immutable element. Thus, the security of a blockchain comes mainly from the fact that its ledger is immutable [3].

Indeed, a blockchain infrastructure comes along a number of other intrinsic properties which may not all be suitable or beneficial for all types of IT applications, while a classical "centralized or distributed peer-to-peer" system may be more appropriate. Consequently, in this work, we are interested in proposing a decision tree identifying whether a blockchain is the best solution for a given IT application, or a classical system is more adapted. Our proposal relies on a comparative study between the blockchain and classical systems.

This paper is organized as follows. In section II, we introduce an overview of classical systems, and in section III, we present a background on the blockchain technology. In section IV, we compare between the three infrastructures: blockchain, centralized and distributed peer-to-peer, and in section V, we describe our proposed decision tree. In section VI, we review a selection of the related works and we compare them to our proposal. In the last section, we provide a brief conclusion. We note that all our abbreviations are illustrated in Tab-I.

II. About Classical Systems

In order to develop an IT application, we typically have the choice between a centralized system and a distributed peer-to-peer system. This choice is based on the needs of the IT application and the properties of each system.

- Centralized System: the users of this system can communicate together in a rapid way thanks to a TTP. The latter is a powerful central authority with a large calculation capacity which is able to process multiple requests at the same time. This TTP is considered the main center of management and trust for all users. In addition, it represents the primary database of the system [4]. We note that the implementation of a centralized system does not provide a security layer by default, so, it is necessary to add a cryptographic layer to this system in order to ensure security properties such as integrity, authentication, etc.
- Distributed Peer-to-Peer System: it is generally used to share files between the system users (peers) without the need for a TTP: if for example a peer *U1* needs a specific file *F*, *U1* sends a direct request for all the other peers and the peer *U2* owning *F* responds to *U1* by sending *F*. In fact, in this system, there is no centralized database but each peer has its own database which stores a list of files to be shared with other peers. We specify that the peers do not share their databases with each other but they share files on demand [5]. About the security level, the implementation of a distributed peer-to-peer system is similar to a centralized system where a cryptographic layer is needed for security purposes.

TABLE I
ABBREVIATIONS

Abbreviation	Description
Ui	User or Peer *i* (*i=1,2,..*)
B	Block
L	Ledger
T	Transaction
F	File
PK(Ui)	Public Key of a User *Ui*
SK(Ui)	Secret Key of a User *Ui*
H(T)	One way hashing function of *T*
Hpow(B)	One way hashing function of *B* generated with the PoW
SignUi(T)	Electronic Signature of *T*. It is generated thanks to *SK(Ui)*

III. BLOCKCHAIN TECHNOLOGY

A. Background

The blockchain infrastructure is based on the principle of a distributed peer-to-peer system where two peers can communicate together without going through a TTP. However, it is different from the conventional peer-to-peer system in its specifications and execution environment as follows [6]:

- The blockchain can be used for file sharing but it is principally designed and intended to execute trusted transactions between users such as that based on cryptocurrencies (see section III-C).
- The blockchain database is represented in the form of a ledger *L* which does not store a list of files as in the peer-to-peer system, but it stores in a tamper-proof and secure way the history of all the exchanges made between the users. In addition, each user has a copy of *L* and there is a consensus algorithm ensuring that each user owns the same copy of *L* as the other users [7] (see section III-B).
- The security layer is indeed one of the intrinsic properties of blockchain technology (see section III-E), which it is not the same case in conventional systems. For authentication purposes, each user in the blockchain has its own key pair (public/secret) which is generated thanks to the Elliptic Curve Digital Signature Algorithm (ECDSA). For the integrity of transactions and blocks, hashing functions are used [8].
- In a blockchain infrastructure, when a user *U1* performs a transaction *T* with another user *U2*, the processes that will be executed are as follows [9]:
  1. *U1* generates *SignU1(T)* by signing the hash *H(T)* using its secret key *SK(U1)*. This signature guarantees the authentication of *U1* and the integrity of *T*.
  2. *U1* broadcasts *T* and *SignU1(T)* to all the other users.
  3. Each receiver verifies the authenticity of *U1* and the validity (integrity) of *T* using *PK(U1)*.
  4. All users participate together in the execution of a consensus algorithm in order to obtain the eligible hash of the block (see section III-B).
  5. The user who first finds the eligible hash, creates the block *B* and broadcasts it to all the other users. *B* mainly contains the following elements [10]:
     - Data of *T*: depend on the type of the blockchain application. The *SignU1(T)* is included also in these data.
     - *Hpow(B)*: it is the eligible hash identifying the current *B* and ensuring its integrity.
     - *Hpow(previous B)*: it links the current *B* to the previous *B*. This link creates a chain of blocks.
  6. Each user adds *B* to its copy of *L* by linking it to the previous block thanks to *Hpow(previous B)*.

B. Consensus Algorithm

In a blockchain infrastructure, a consensus algorithm is the responsible for maintaining the security of the blockchain. It is a mechanism by which a blockchain network reaches a consensus. It allows the distributed peers to agree on the validity of the transactions and therefore. This consensus algorithm requires execution time in order to allow all the blockchain users to agree on the same block, ensure that the last block has been correctly added to the chain and protect the blockchain against malicious attacks. There are several types of consensus algorithms. The most common implementations are Proof of Work (PoW) and Proof of Stake (PoS) [11] [12] [13].

1) Proof of Work (PoW): it is also known as mining. It is a data that is difficult to produce because it requires a time for consumption (ten minutes in average). The PoW is easy to check by the other users of the blockchain. In fact, the production of a valid PoW is an eligible hash which depends on a random process with a low probability, so that a lot of trial and error is required on average before a valid PoW is generated.

2) Proof of Stake (PoS): the PoS consensus algorithm was developed in 2011 as an alternative to PoW. Although PoW and POS share similar goals, they also have some fundamental differences and features. The PoS asks the user to prove possession of a certain amount of cryptocurrency (their participation) to claim to validate additional blocks in the blockchain and to receive the reward.

C. First Blockchain Application (Bitcoin)

The idea of blockchain technology was introduced in 1991 by a group of researchers to time-stamp digital documents that could not be backdated or change their contents [14]. Then, it was not really used until Satoshi Nakamoto used

this concept in 2009 to create the Bitcoin payment system [15] [16]. The latter was therefore the first application using the blockchain infrastructure. Bitcoin system is an application of cryptocurrencies allowing two persons to perform financial trusted transactions without passing through a TTP, and then without passing through a banking network [17] [18]. So, for a transaction *T* of exchanging *1 bitcoin* from the user *U1* to the user *U2*, the contents of the *B* are: Data of *T* (the sender *U1*, the receiver *U2* and the amount *1 bitcoin*, the signature *SignU1(T)*), the eligible hash *Hpow(B)*, the link to the previous block with its eligible hash *Hpow(previous B)*.

D. Types of Blockchains

A blockchain infrastructure may be permissionless or permissioned. With a permissionless blockchain, any user can read or write at any time. With a permissioned blockchain, only a set of users which is allowed to write and read [19]. In addition, a blockchain may be public or private. In a public blockchain, each user is allowed to contribute in the validation of a block. In a private blockchain, all users are known and the validation of a block is done by a selected set of users [20]. Consequently, we can conclude the three main types of blockchain implementations: 1. Permissionless blockchain, 2. Public permissioned blockchain, 3. Private permissioned blockchain.

E. Intrinsic Properties of the Blockchain

We summarize the intrinsic properties of the blockchain in this section [21] [22]:

1. *Distribution (No need for a TTP)*: as presented in sections I and III-A, it is the main property of the blockchain and it means that the blockchain communications do not rely on a TTP, the data are stored in a distributed manner and the blockchain users communicate together without the need for a TTP and without the need they know or trust each other before.
2. *Data Replication*: all the blockchain users have the same copy of the ledger and then the data are duplicated throughout the system.
3. *Data Transparency*: each user of the blockchain can observe how blocks have been added over time: everything (transactions, messages, etc.) is transparent and that is why the blockchain technology can be trusted.
4. *Data Integrity & Authentication of the Origin*: the eligible hash of the block allows to guarantee its integrity. The electronic signature of the transaction generated by the user allows to ensure the integrity of this transaction and to ensure the authenticity of the user (see section III-A).
5. *Data Immutability*: the information stored in the blocks are indeed reserved forever and cannot be changed unless an attacker can gather of more than 51% of the computational power network (see section I).

IV. Comparative Study between the Blockchain & Classical Systems

In this section, we provide a comparative study between the three infrastructures: blockchain, centralized and distributed peer-to-peer. This study is based on the characteristics/properties of each infrastructure.

1. *Data Replication*: in a blockchain infrastructure, the replication of data is an intrinsic property as presented in the previous section. In a peer-to-peer distributed system, the replication of data is needed to encourage file sharing. However, it is not advantageous to implement a peer-to-peer system without data replication. In a centralized system, the replication may be used or not and it depends on the application requirements: it aims to protect the data and to ensure a better scalability [11] [23].
2. *Cryptographic Layer*: as presented in section II, a new cryptographic layer is needed to implement for classical systems in order to ensure security objectives such as: data integrity, authentication, etc. However, in a blockchain infrastructure, a security layer is presented in the intrinsic properties 3., 4. and 5. (see the previous section).
3. *Response Time Versus Loading Time*: in a centralized system, the response time to a request is less than one second [24] and the system adapts systematically to the loading time if there is a large volume of requests [25]. This advantage is due to the powerful TTP that supports several management mechanisms. Consequently, we can guarantee with a centralized system that the response time always remains < 1 second (fast option) even if the TTP is overloaded. However, we cannot guarantee this property with a distributed peer-to-peer system or a blockchain because: the peers do not have a large calculation capacity, they need to send data to all the other peers and especially with the blockchain infrastructure an additional complexity is added through the consensus algorithm (see section III-B). For example, Bitcoin can only execute a seven transactions per second, while the Visa centralized system can execute more than fifty thousand transactions per second [26].
4. *Secure Code Execution, Execution Transparency*: it means a secure and transparent calculation that represents the execution of trusted transactions between users (such as financial transactions). The main role of a blockchain is the implementation of this kind of operations that can be done by default. As well as, we can implement them in other classical systems but by adding a new security cryptographic [23] [27].
5. *Restricted Access*: it usually exists in a distributed peer-to-peer system and a blockchain infrastructure but not in a centralized system [11]. It presents the question of writing and reading access to the application: can everyone access it or only a specific group of users?. In response to this question: - for a peer-to-peer system, we can implement it with a public or private (or also trusted, see boxes (8) and (11) in section V-A), - for a blockchain infrastructure, we can implement it with a permissionless or permissionned type (see section III-D) [23] [27].

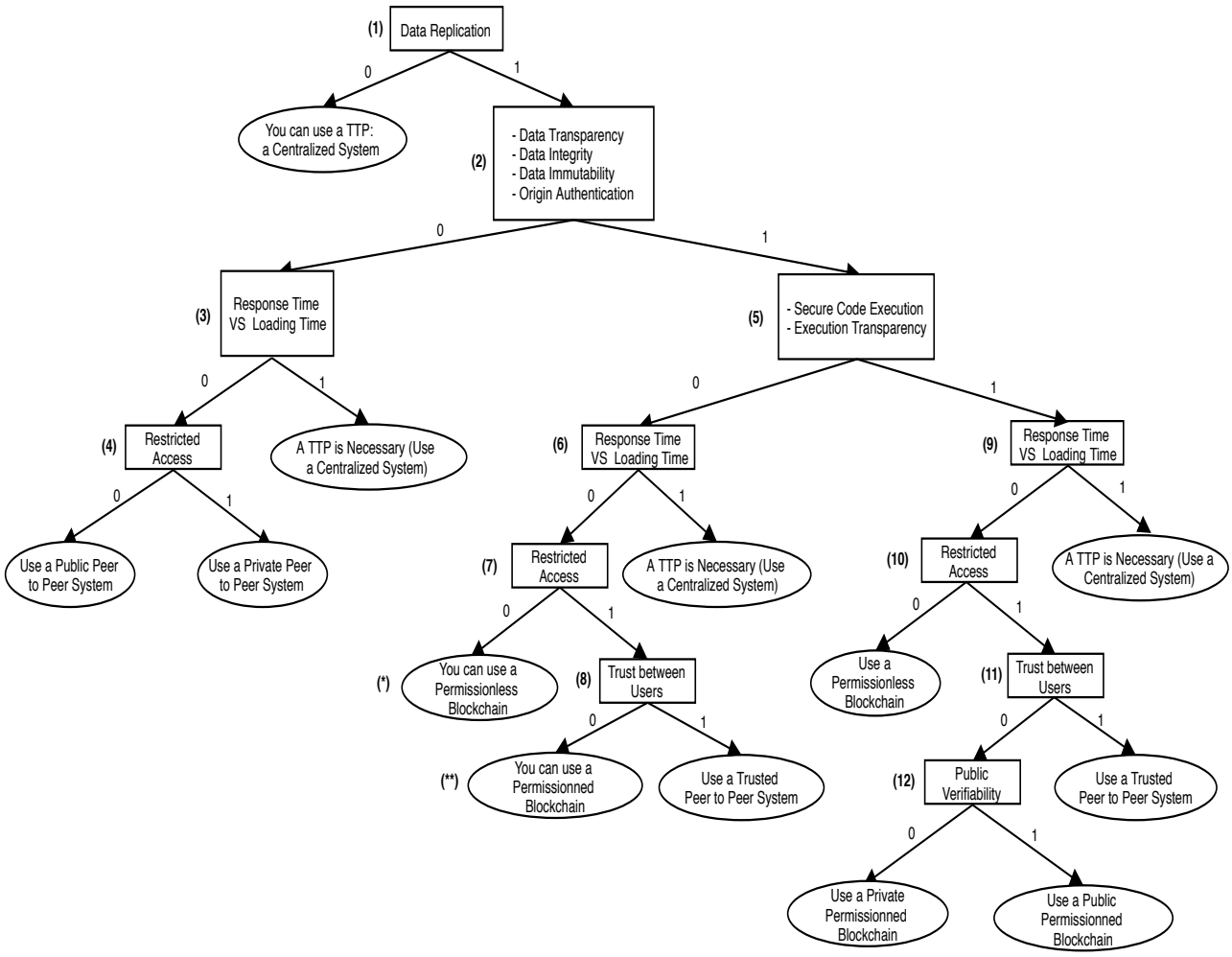

Fig. 1. Proposed Decision Tree

V. Proposed Decision Tree

In this section, we describe our proposed decision tree illustrated in Fig-1. According to the needs of the IT application, our proposal allows to identify which infrastructure is the best solution for the implementation.

A. Proposal Description

The design of our decision tree is based on the needs of the IT application that are directly related to the comparison characteristics discussed in section IV. As illustrated in Fig-1, each box represents one or more requirements for the IT application, the leaves represent the different possible decisions for the infrastructures, the right part is the true case and the left part is the false case. We number the boxes from (1) to (12). This numbering does not mean that the execution of the boxes is sequential but it is intended to facilitate the explication of our proposal.

- We note that the "Presence of TTP" is considered in our work as a decision result in a centralized system and not as a requirement for the IT application. We have aggregated the properties of a centralized system in "data replication" and "Response Time VS Loading Time". As presented in section IV, if the application needs or not for the data replication than a centralized system may be used, and if the application needs to ensure the property of the response time < 1 second than we absolutely need to choose a centralized system.
- We note that the needs given in (see Fig-1): box (2), box (5) and boxes (3), (6), (9) are considered as essential elements for making the result decisions in the best conditions according to our comparative study in section IV.

We now describe how our tree makes a decision of an infrastructure for a type of an IT application:

(1) We start by asking if the IT application needs the replication of data:

– If it does not need, then we decide to implement a

centralized system. As presented in section IV, this decision is due to the fact that it is not advantageous to implement a peer-to-peer system without replicating the data and it is not possible to implement a blockchain without replication. Consequently, the non-replication of data is better supported in the case of a centralized system.
  - If it needs, then we go to the box (2).
- (2) If the "data transparency, data integrity, data immutability and the origin authentication" are not required for the IT application, then the implementation of a blockchain is not necessary and we will have, after running the box (3), two choices: either a peer-to-peer distributed system or a centralized system. Otherwise, we go to the box (5).
- (3) As presented in section IV, the response time to a request is < 1 second in a centralized system and this property is not guaranteed with a peer-to-peer distributed system or a blockchain. Consequently, if the application needs to ensure this property, then we need to choose a centralized system. Otherwise, we can only implement a peer-to-peer system after running the box (4).
- (4) At this stage, we ask the question on access to the IT application. If the application access is restricted then we can use a private peer-to-peer system, if not a public peer-to-peer can be used.
- (5) If the IT application needs or not to execute trusted transactions, we will ask the need of response time versus loading time respectively in boxes (9) and (6).
- (6) If the application needs to respect the response time < 1 second, then we need to use a centralized system. Otherwise, we go to the box (7).
- (7) If the access to the application is not restricted, then we can use a permissionless blockchain (∗). Otherwise, we go to the box (8).
- (8) When the IT application needs to guarantee the requirements of box (2) and the application access is restricted, we need to ask this question about the trust between users. If the application needs that users trust each others, then we can use a trusted peer-to-peer system. This system can be both secured and distributed but remains less expensive than a blockchain. Otherwise, we can use a permissioned (because of the restricted access is true) blockchain "private or public" (∗∗).
  - For (∗) and (∗∗), the use of a blockchain is not advantageous if we do not need to do trusted transactions and we only need to implement the file sharing for example. The blockchain can be used as well as we need to guarantee the requirements of box (2), and these requirements cannot be ensured with a public/private peer-to-peer system without implementing a new cryptographic layer.
- (9) If the application needs to respect the response time < 1 second, then we need to use a centralized system. Otherwise, we go to the box (10).
- (10) If the application access is not restricted, then we can use a permissionless blockchain. Otherwise, we go to the box (11).
- (11) If the application needs the trust between users (i.e: a group of friend), then we can use a trusted peer-to-peer system which can be both secured and distributed but remains less expensive than a blockchain. Otherwise, we go to the box (12).
- (12) As presented in section III-D, the public verifiability allows any user to verify the correctness of the blockchain system. in the private verifiability, a set of specific users that can verify the state of the blockchain. Therefore, if the IT application needs the public verifiability, then we can use a public permissioned blockchain. Otherwise, a private permissioned blockchain is necessary. The permissioned is because the restricted access is true.

B. Discussions (Examples)

In this section, we provide some examples aiming to discuss and validate our decision tree.

- A notarial IT application: it is an application for writing notarial contracts. By running our tree, we obtain:
  - The application needs: data replication in (1), data Transparency /Integrity /Immutability and the origin authentication in (2), trusted transactions in (5).
  - The application does not need to respect that the response time is < 1 second in (9). The application access is not restricted in (10).
  - (1) true, (2) true, (5) true, (9) false, (10) false –> Use a Permissionless Blockchain.
- Rental application "owner-tenant": it is an application to rent housing between owners and tenants. By running our tree, we obtain:
  - The application needs: data replication in (1), data Transparency /Integrity /Immutability and the origin authentication in (2), trusted transactions in (5).
  - The application does not need to respect that the response time is < 1 second in (9). The application access is restricted (a set of owners) in (10).
  - The trust between users is not needed in (11). The public verifiability is not needed in (12) (only the set of owners which can validate the blocks).
  - (1) true, (2) true, (5) true, (9) false, (10) true, (11) false, (12) false –> Use a Private Permissionned Blockchain.
- Family file sharing application: it is a simple application to share files between family members or friends. By running our tree, we obtain:
  - The application needs: data replication in (1), data Transparency /Integrity /Immutability and the origin authentication in (2).
  - The application does not need: to perform trusted transactions in (5), to respect that the response time is < 1 second in (6). The application access is restricted in (7). The trust between users is needed in (8).

  - (1) true, (2) true, (5) false, (6) false, (7) true, (8) true –> Use a Trusted peer-to-peer System.
- Navigation application "Waze": it is an application of mobile navigation. By running our tree, we obtain:
  - The application needs: data replication in (1), data Transparency /Integrity /Immutability and the origin authentication in (2), trusted transactions in (5), that the response time is < 1 second in (9).
  - (1) true, (2) true, (5) true, (9) true –> A TTP is Necessary (Use a Centralized System).
- B2B traceable supply chain: in this application the different actors of the supply chain do not trust each other. By running our tree, we obtain:
  - The application needs: data replication in (1), data Transparency /Integrity /Immutability and the origin authentication in (2), trusted transactions in (5).
  - The application does not need to respect that the response time is < 1 second in (9). The application access is restricted (the different actors of the supply chain) in (10).
  - The trust between users is not needed in (11). The public verifiability is needed in (12) (all actors participate to validate a blocks).
  - (1) true, (2) true, (5) true, (9) false, (10) true, (11) false, (12) true –> Use a Public Permissionned Blockchain.

VI. Related Works

In literature, several decision models have been proposed aiming to identify whether a blockchain is needed or not for a given IT application. We review in this section a selection of these models and we compare them to our proposed decision tree. In fact, our decision tree is based on the needs of the IT application that are directly related to the comparison characteristics discussed in section IV. Our proposed tree helps an IT application to identify exactly which infrastructure is the best solution for the implementation: blockchain or centralized or distributed peer-to-peer. In addition, it specifies the types of infrastructures such as private peer-to-peer system, permissionless blockchain, public permissioned blockchain, etc. The strong point of our decision tree (see Fig-1) is that it addresses the needs given in: box (2) (data transparency, data integrity, data immutability, origin authentication), box (5) (secure code execution, execution transparency) and boxes (3), (6), (9) (response time versus loading time). These needs make the result decisions in the best conditions (see sections IV and V-A).

In the research work [28], a simple decision model has been proposed which allows to identify which type of Distributed Ledger Technology (DLT) is the appropriate solution for an IT application. This model takes into consideration only the characteristics: restricted access and the data integrity and it does not treat the decisions for the infrastructures: blockchain, centralized and distributed peer-to-peer. Authors in [19] present a decision model that aims to identify the best suited solution among: permissionless blockchain, public permissioned blockchain, private permissioned blockchain, do not use a blockchain. Another decision model is proposed in [29] which allows to choose the appropriate solution among: public blockchain, private blockchain, do not use a blockchain. Thus, in the research work [30], a decision model is introduced which identifies the most adapted solution among: permissionless blockchain, permissioned blockchain, do not use a blockchain.

In fact, the three models [19], [29] and [30] are based only on the following characteristics/needs: restricted access, trust between users, public verifiability and the presence of a TTP. They lack addressing the following needs: a fast execution (as in boxes (3), (6), (9) in our proposal), a security layer and trusted transactions (as in boxes (2), (5) in our proposal). In addition, the model introduced in [19] does not address the decisions: public/private/trusted peer-to-peer system, centralized system. The model presented in [29] does not take into consideration the decisions: permissionless blockchain, public/private permissionned blockchain, public/private/trusted peer-to-peer system, centralized system. The model proposed in [30] lacks to identify the solutions: public/private/trusted peer-to-peer system, centralized system.

We note that the three models [19], [29] and [30] only indicate the decision of "do not use a blockchain" without specifying which conventional system is the best suited. To the best of our knowledge, our proposal was not presented with the same ideas/needs in the literature. This make us the first to give better result decisions than the related works.

VII. Conclusion

In this paper, we proposed a decision tree identifying whether a blockchain is the appropriate solution for a given IT application, or a classical "centralized or distributed peer-to-peer" system is more adapted. The design of our decision tree is based on the needs of the IT application and the main properties of the three infrastructures.

References

[1] Y. Caseau and S. Soudoplatoff, "La blockchain, ou la confiance distribuée," *Fondation pour l'innovation politique*, 2016.

[2] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges." *IJ Network Security*, 2017.

[3] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," *Future Generation Computer Systems*, 2017.

[4] Z. Wan, R. H. Deng, and D. Lee, "Electronic contract signing without using trusted third party," *International Conference on Network and System Security*, 2015.

[5] S. Zaid, G. Linscott, A. Becevello, T. Zaid, and P. Lem, "System and method for anonymous addressing of content on network peers and for private peer-to-peer file sharing," *Google Patents, US Patent 9,112,875*, 2015.

[6] M. Pilkington, "11 blockchain technology: principles and applications," *Research handbook on digital transformations*, 2016.

[7] M. Belotti, N. Bozic, G. Pujolle, and S. Secci, "A vademecum on blockchain technologies: When, which and how," 2018.

[8] J.-P. Delahaye, "Le bitcoin une monnaie révolutionnaire?" *UMR*, 2014.

[9] S. Norton, "Cio explainer: What is blockchain?" *The Wall Street Journal*, 2016.

[10] A. M. Antonopoulos, "Mastering bitcoin: unlocking digital cryptocurrencies," *" O'Reilly Media, Inc."*, 2014.

[11] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in internet of things: challenges and solutions," *arXiv preprint arXiv:1608.05187*, 2016.
[12] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, "A review on consensus algorithm of blockchain," *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2567–2572, 2017.
[13] L. Bach, B. Mihaljevic, and M. Zagar, "Comparative analysis of blockchain consensus algorithms," *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1545–1550, 2018.
[14] D. Yermack, "Corporate governance and blockchains," *Review of Finance*, 2017.
[15] S. Nakamoto, "Bitcoin a peer-to-peer electronic cash system," *Working Paper*, 2008.
[16] M. C. K. Khalilov and A. Levi, "A survey on anonymity and privacy in bitcoin-like digital cash systems," *IEEE Communications Surveys & Tutorials*, 2018.
[17] D. Ron and A. Shamir, "Quantitative analysis of the full bitcoin transaction graph," 2013.
[18] J. Pons, "La mise en œuvre de la blockchain et des smart contracts par les industries culturelles," *Annales des Mines-Réalités industrielles*, 2017.
[19] K. Wüst and A. Gervais, "Do you need a blockchain?" *IACR Cryptology ePrint Archive*, 2017.
[20] A. Ellervee, R. Matulevicius, and N. Mayer, "A comprehensive reference model for blockchain-based distributed ledger technology," *ER Forum*, 2017.
[21] P. K. Sharma, S. Y. Moon, and J. H. Park, "Block-vn: A distributed blockchain based vehicular network architecture in smart city," *Journal of Information Processing Systems*, 2017.
[22] M. Pilkington, "Blockchain technology: principles and applications. research handbook on digital transformations, edited by f. xavier olleros and majlinda zhegu," *Edward Elgar*, 2016.
[23] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2016.
[24] J. A. Miller, C. Bowman, V. G. Harish, and S. Quinn, "Open source big data analytics frameworks written in scala," 2016.
[25] F. L. Haddi and M. Benchaïba, "A survey of incentive mechanisms in static and mobile p2p systems," *Journal of Network and Computer Applications*, 2015.
[26] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2016.
[27] E. Hildenbrandt, M. Saxena, X. Zhu, N. Rodrigues, P. Daian, D. Guth, and G. Rosu, "Kevm: A complete semantics of the ethereum virtual machine," *Technical Report*, 2017.
[28] D. Birch, R. G. Brown, and S. Parulava, "Towards ambient accountability in financial services: Shared ledgers, translucent transactions and the technological legacy of the great financial crisis," *Journal of Payments Strategy & Systems*, 2016.
[29] B. Suichies, "Why blockchain must die in 2016," 2015. [Online]. Available: https://medium.com/ @bsuichies/why-blockchain-must-die-in-2016-e992774c03b4
[30] M. E. Peck, "Do you need a blockchain?" 2017. [Online]. Available: https://spectrum.ieee.org/computing/networks/do-you-need-a-blockchain