energies

MDPI

Article

A Distributed Ledger-Based Automated Marketplace for the Decentralized Trading of Renewable Energy in Smart Grids †

Dušan B. Gajić [1], Veljko B. Petrović [1], Nebojša Horvat [1], Dinu Dragan [1], Aleksandar Stanisavljević [1], Vladimir Katić [1] and Jelena Popović [2,*]

[1] Faculty of Technical Sciences, University of Novi Sad, 21000 Novi Sad, Serbia; dusan.gajic@uns.ac.rs (D.B.G.); pveljko@uns.ac.rs (V.B.P.); horva.n@uns.ac.rs (N.H.); dinud@uns.ac.rs (D.D.); acas@uns.ac.rs (A.S.); katav@uns.ac.rs (V.K.)

[2] Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7522 NH Enschede, The Netherlands

* Correspondence: j.popovic@utwente.nl

† This paper is an extended version of our paper published in the Proceedings of the 21st IEEE International Symposium on Power Electronics (Ee)—Invited paper, Novi Sad, Serbia, 27–30 October 2021.

Abstract: We present a prototype of a decentralized power trading system based on the use of distributed ledger technology. This sort of efficient, decentralized marketplace is needed to empower prosumers and make them first-class members of a smart, decentralized power grid in order to drive further renewable energy adoption. Unlike the bulk of previous work in this field, we focus on private permissioned distributed ledgers rather than conventional blockchains. The proposed solution is entirely independent of cryptocurrency, with an explicit design capability of being adapted piecemeal without any fundamental changes to the present regulatory environment. To be economical, efficient, and scalable, our prototype is based on a lean, Corda-based private permissioned distributed ledger. It allows for instant, automatic bidding on and trading of 'power promises' and the robust implementation of short-term, small-scale liquid electrical power futures. We demonstrate that the prototype performs well and presents several clear advantages over existing solutions based on conventional blockchains. Therefore, the proposed approach represents a promising, robust solution to the smart grid decentralized power trading problem.

Keywords: energy trading; renewable energy; smart grid; distributed systems; distributed ledger technology; blockchain

check for updates

Citation: Gajić, D.B.; Petrović, V.B.; Horvat, N.; Dragan, D.; Stanisavljević, A.; Katić, V.; Popović, J. A Distributed Ledger-Based Automated Marketplace for the Decentralized Trading of Renewable Energy in Smart Grids. *Energies* **2022**, *15*, 2121. https://doi.org/10.3390/en15062121

Academic Editor: Alberto Geri

Received: 27 January 2022
Accepted: 11 March 2022
Published: 14 March 2022

1. Introduction

The ever-increasing presence of renewable sources of energy is one of the most notable trends in power grids today [1,2]. The process of the decentralization of previously centralized power grids is characterized by the development of new kinds of electrical power plants and distributed energy sources (DERs), leading to a completely new type of grid entity—prosumers [2]. Another important development is the advent of distributed storage and electrical vehicles, which represent movable, distributed demand and storage potential [2]. This change in the power system architecture towards decentralization has also been happening almost simultaneously in the field of information systems and databases. When distributed ledgers and blockchain technology were originally proposed in 2008 [3], a fundamentally new decentralized way of performing transactions and value exchanges was created.

The synergy of decentralized technologies for the generation, storage, and use of renewable energy and blockchain technologies creates fertile ground for a fundamental shift in the very nature of power systems [4,5]. Since its inception, the process of electrical energy production, distribution, and trading was centralized and based around a limited number of regional, vertically integrated monopolies. Today, the share of power produced using renewable energy generators by prosumers is constantly increasing [4]. One of the main driving forces in this transformation of the grid are power electronics converters, which introduce much faster control and thus lead to power systems characterized by low inertia. Further, a trend of energy companies increasingly reporting higher energy costs and lower revenues has also become evident [6]. Concurrently, these companies also face requirements for increasing transparency set by the regulatory authorities [6].

As a result, any possibility of optimization through cost savings, creation of new value streams, and improvements of efficiency in the operation of power systems and markets is a significant research and development goal clearly worth investigating. The research we report on in this paper was motivated by this necessity and led to the development of a transparent, traceable, and auditable information system that can support the trading of electrical power produced in such a decentralized setting by making use of distributed ledger technology (DLT). Both power companies and prosumers need to be able to independently verify that power transactions, which are automatically executed in their thousands every second, are being performed in accordance with the pricing schemas and rules agreed upon in advance. DLTs have also already proven that not only can they enable prosumers to become first-class participants in the energy system but they can also help to achieve optimization by correcting the energy index of the whole system [7].

Therefore, we propose a new system that is built for energy trading in a decentralized grid with many prosumers. We call it a Distributed Renewable Energy Automated Marketplace (DREAM). The system is built around a lean private permissioned distributed ledger and features support for intelligent agent trading and has infrastructure features permitting drastically lower-cost participation in the system by members of the general public. To the best of our knowledge, this is the first proposal in the literature of a decentralized power trading marketplace based on a DLT that is not a blockchain. DLTs, which operating on the transaction rather than block level like blockchains do, are free of many overheads while still providing full decentralization and improved security.

Further, DREAM is based on the notion of a shared supraordinal grid and a common carrier/producer that serves as the medium of exchange and supplier of last resort. Once it enters production use, DREAM is expected to offer low operational costs and high scalability due to its use of a private permissioned distributed ledger in a similar way to the work presented earlier in [8,9]. Furthermore, DREAM offers a transparent view of the operations of the system to all stakeholders and assures direct oversight of transactions to regulatory authorities. Therefore, we believe that the proposed system can serve as a novel backbone for exchanging data on power transactions in both contemporary and future energy systems. We also present how our solution can be extended to support a future-proof fully decentralized smart grid while still respecting the characteristics of current power grids. To the best of our knowledge, ours is the first publication in the literature to discuss an implementation of a decentralized power trading system using a DLT that is not a blockchain.

The remainder of the paper is organized as follows. In the next section, we first offer an overview of distributed ledger technology and blockchain as its subset. Then, we discuss both current and future applications of DLTs in the energy sector. In Section 3, we present the proposed DLT-based system for decentralized power trading. We first make a clear distinction between other existing solutions and ours. Afterwards, we elaborate upon its architecture, use cases, technology stack, proposed classes of network participants, and details of its DLT subsystem. In Section 4, we discuss in detail the performance of the prototype, its extension, and future avenues of work, including both work on the prototype itself and new lines of enquiry and general research that seek to quantify the applicability of the solution and to broaden the conceptual reach of the system. We conclude the paper by summarizing its main contributions.

2. Background Theory and Related Work

To make the paper self-contained, in this section we offer a short overview of the main concepts involved in distributed ledger technologies. We also explore the state-of-the-art in applications of DLTs and blockchain in the energy sector, with special attention given to existing blockchain applications for peer-to-peer (P2P) energy trading. We conclude with a section that seeks to illustrate what we believe are the chief problems faced by these solutions and how our contrast approach might be able to overcome these difficulties.

2.1. DLT and Blockchain

There is a common misconception that puts an equality relation between cryptocurrencies and blockchain. As a remedy, before we elaborate upon the state-of-the-art in DLT applications in the energy sector, we will briefly address the terminology.

A distributed ledger or distributed ledger technology is a special type of distributed database that assumes the potential existence of malicious users within the system's P2P network [3,10–12]. DLT can also be defined as a consensus of replicated, shared, and synchronized digital data geographically spread across multiple locations [9,11,12]. A blockchain is a data structure, originally proposed in [3]. It implements a distributed ledger in one specific way—as a chain of cryptographically linked blocks containing sets of transactions [3]. In blockchain, instead of writing each transaction separately to the distributed ledger, a whole "page" with several different transactions is added to the ledger. In addition to the distributed ledger and the P2P network, blockchains have two other key components—smart contracts and the consensus algorithm. Smart contracts are self-executing programs that automatically perform certain actions, such as reading from or writing to the distributed ledger, when certain preset conditions are met. The consensus algorithm needs to ensure that data on the ledger are the same for all participants in the blockchain's P2P network and in so doing it prevents malicious actors from manipulating the data.

In terms of read and write operations, DLTs can be divided into four classes. Read access to the distributed ledger can be made public or private, while writing to the ledger may or may not require permissions. This leads us to the four classes of DLTs: public permissionless, private permissionless, public permissioned, and private permissioned. Two out of these four dominate real-world DLT applications. The first one is public permissionless DLTs, such as Bitcoin [3,10] and Ethereum [10,13]. This class of DLTs is typically used as a basis for cryptocurrencies and offers pseudo anonymity to their users. The second class, private permissioned DLTs [11,12,14,15], is mostly used in an enterprise context. When compared to their public counterparts, private permissioned DLTs offer more granular data access control, more privacy, higher transaction throughput, better scalability, and improved modularity, all at the price of required user authentication and authorization. The most popular representatives of private permissioned DLTs include Hyperledger Fabric [15] and R3 Corda [14]. It should be noted that R3 Corda performs read and write operations to the distributed ledger on the level of separate transactions instead of creating blocks of transactions and is thus a DLT but does not use the blockchain data structure [14].

2.2. Applications of DLTs in the Energy Sector

The energy sector has recently been one of the fields where an immense rise in interest in the application of DLTs has been observed. This development was analyzed and summarized in detail in a broad study performed by Andoni et al. [4]. This study first explains the key principles of distributed ledgers and then offers a systematic review of their applications in the energy sector. As shown on the graph in Figure 1, which is based on the data from [4], one third of all energy sector DLT use cases relate to energy trading, which is also the main topic of this paper. The use of cryptocurrencies and tokens is in the second place, followed by applications of blockchain in smart devices, automation, and asset management. A significant number of projects are also concerned with topics such as metering, billing, and security, as well as carbon certificates and carbon trading.

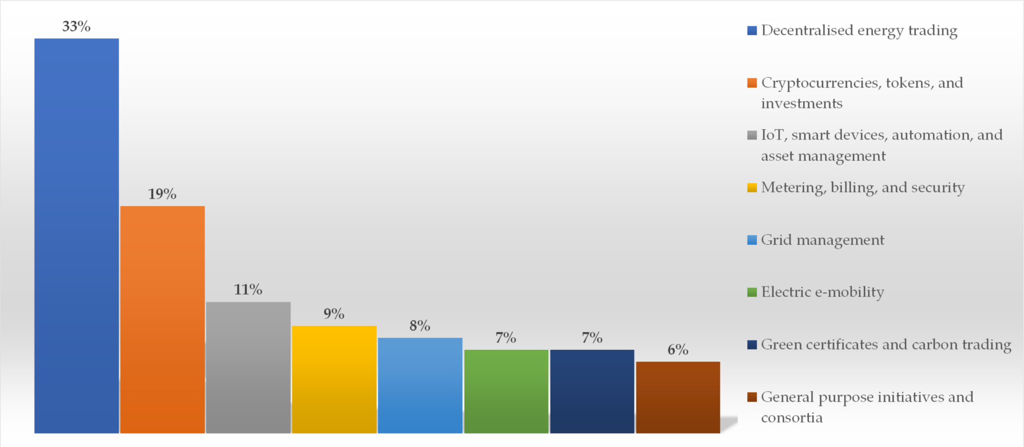

Figure 1. DLT energy sector use case classification according to activity field.

Research presented in [4] also includes a classification of DLT energy sector use cases by DLT platform used, as shown on the graph in Figure 2. One half of the projects studied use the public permissionless Ethereum blockchain, with a further tenth of the market taken by Energy Web, which is also Ethereum-based. The Hyperledger private permissioned family of blockchains were also found in a significant number of projects, while all the other DLTs combined commanded one-quarter of the use cases. Since [4] was published in 2019, interest in applying private DLTs in the enterprise domain has significantly increased, mainly due to the improved maturity of these technologies [12,14,15].

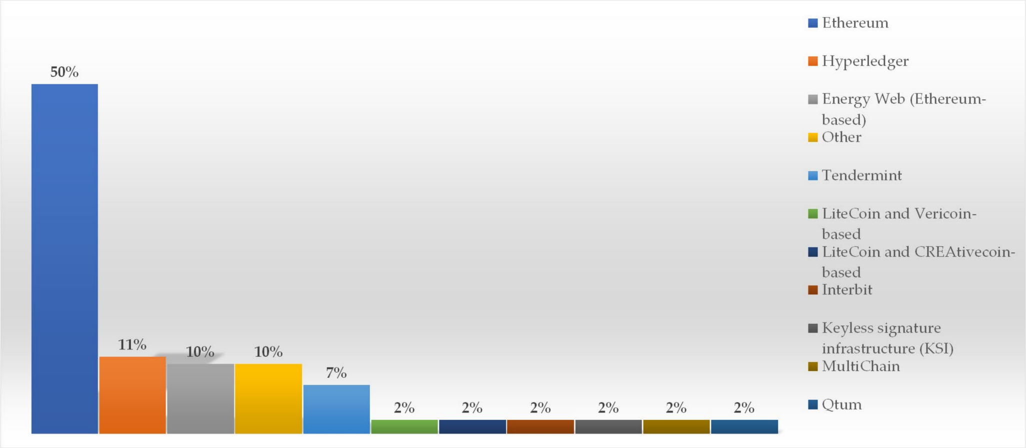

Figure 2. DLT energy sector use case classification according to DLT platform used.

We also performed a detailed analysis of the academic research, industry-led projects, and commercial enterprises dealing with P2P energy trading. We found that a rising number of projects [16,17] have already successfully tested DLT-based energy trading in practice. This proves the overall feasibility and value of the approach to decentralized energy trading based on DLTs and attracts further interest from researchers.

The Brooklyn Microgrid case study, which completed a successful three-month trial run of P2P energy trading in a small Brooklyn, New York community in the United States, is described in detail in [16]. Prosumers were allowed to sell energy surpluses directly to their neighbors using the implemented microgrid infrastructure. The Brooklyn Microgrid platform used public Ethereum-based smart contracts and a PBFT [18] consensus implemented by Tendermint to achieve the goals of the project.

Another similar project is Power Ledger [17], which offers a widely tested and mature power trading solution. At the time of writing, Power Ledger is used in nine countries, which signals that blockchain-based decentralized power trading platforms are a viable solution and have the potential to truly transform the way we produce and trade energy. The Power Ledger platform uses a hybrid public and consortium blockchain solution to minimize energy consumption that comes with the public Ethereum ledger and the proof-of-work (POW) algorithm that it uses. Power Ledger offers tokens (named POWR and Sparkz) that can be publicly traded to make a profit or consumed to provide electricity.

The existing academic research on the topic of DLT-based energy trading, which is also our topic in this paper, includes references such as [8,9,19–22]. The research reported in [21] presents the state-of-the-art in applications of blockchain for energy trading in power systems in 2019. This study first identifies and summarizes challenges in blockchain-based energy trading. Afterwards, they discuss the existing energy trading schemes and classify them into three categories based on their main focuses: energy transaction, consensus mechanism, and system optimization. Further, in [8] a Hyperledger Fabric-based model for localized P2P energy trading is proposed. The model consists of three entities that are energy nodes, energy aggregators, and smart energy meters. Energy nodes are used for selling or buying energy and their role depends on the energy state of the owner. Energy aggregators perform the role of brokers that regulate trade between energy nodes. Smart meters record energy trading in real time. A system for energy trading based on a blockchain network that uses tokens to allow P2P trading is also proposed in [20]. Tokens are created or consumed by nodes that transfer energy through the energy storage system. The research reported in [19] proposes a custom blockchain network that uses the proof-of-work (POW) consensus algorithm [3] to store all energy trading transactions. The use of the POW algorithm introduces delays in transaction validation, which is addressed by designing a credit-based payment schema. Nodes apply for loans according to their credit strategy. They also propose an optimal loan pricing strategy for credit-based payment schemas to maximize the utility of credit banks.

Research reported in [9] proposes a solution with a provided case study and a prototype for P2P energy trading based on the Hyperledger Fabric private permissioned blockchain. This implementation allows users to seamlessly trade energy with the use of a two-phase algorithm. The first phase deals with scheduling energy generation day-ahead, while the second phase is used for real time network management (hour-ahead scheduling). The research presented in [22] offers an energy trading scheme for a Virtual Power Plant (VPP) using smart contracts executed on the Ethereum network. The scheme proposed in [22] aims to solve the same problem that we are addressing but it uses a public blockchain and operates auctions via Ethereum smart contracts requiring gas fees for their computation.

As reported in [4], in addition to energy trading, a significant number of other use cases of DLT in the energy sector exist. Ponton has developed a pilot blockchain-based solution called Gridchain that simulates processes used for real-time grid management [23]. They focus on in-advance balancing of power loads. This allows smart control over power generators before they are ramped up and keeps grid load and frequency stable. Wipro and SAP have developed a green energy tracking and distribution platform based on SAP's blockchain cloud platform [24]. This system leads to a reduction in the cost of green energy transactions and facilitates new business models between retailers, prosumers, and commercial consumers. It provides a cheaper alternative for demand side management when compared to large capital expenditure-driven green energy power plants [24]. It also increases adherence to regulatory demands in regard to green energy sourcing [24].

General Electric is investigating potential blockchain use in their digital power plants (DPS) [25]. As the trend moves away from large central power plants towards distributed small or medium generators, General Electric DPS is considering the use of Hyperledger fabric as a blockchain solution, along with other software innovations, in order to satisfy the diverse needs of power assets with high-speed and intelligent infrastructure [25]. In addition to these industry-led projects, academic research on the use of blockchain/DLT technology for addressing different challenges in smart grids and renewable energy systems is also very extensive since DLTs allow for the decentralization of the data management in a secure and transparent way. A comprehensive survey of these topics is presented in [26], as well as in the book [27]. The challenges, limits, and potential of applying DLTs in the energy sector are also studied thoroughly in [27]. Further examples of academic research in the field include papers such as [7,28–30]. The authors in [7] present a framework that allows small energy producers, such as solar panel grids, or prosumers, to offer their excess power production through an intelligent brokerage that is based on blockchain technology. Technical challenges and guidelines for implementing DLTs in various energy sector applications is presented in [28]. In [29], a blockchain-based tool for managing transactions in a smart grid is discussed. In [30], smart contracts executed on a simulated Ethereum blockchain are used to coordinate the decentralized approach to the economic dispatch (ED) problem of distributed generation (DG) units.

2.3. Problems of Applying DLTs to Decentralized Energy Trading and Their Solutions

The solutions described in Section 2.2 display several positive traits and have a commendable degree of technical sophistication. The Brooklyn Microgrid [16], for instance, represents an incredibly detailed, practically functional vision of the future. Power Ledger [17] and Energy Web [31] present concerted efforts to make a software infrastructure that makes such a future possible. As praiseworthy as these efforts are, our approach in this paper will be different.

The reasons for a different approach are both technical (related to the problems these solutions must face) and philosophical (related to the intentions of the design). Technically speaking, the biggest problem (which has also been recognized by others) is the wide use of proof-of-work consensus algorithms [3]. Proof-of-work algorithms are a mechanism by which no transaction is recorded on a blockchain without first being aggregated into a block (and verified by one of the miners keeping the network consistent and preventing double spending and other malicious phenomena). No block is valid without including some sort of proof that a complex cryptographic puzzle intimately connected to the content of the block has been solved. This typically involves computing a string of bits appended to the data of the blockchain that causes the hash of the whole to satisfy some condition. Given that cryptographic hash functions used for this task are deliberately built with strong collision and preimage attack resistance, computing this requires a large amount of trial-and-error. The computational costs are large and, consequently, so are the energy costs. The Ethereum blockchain, used in most of the cases studied, for instance, currently consumes about 70 TWh of power every year, about as much as the entirety of Austria.

The eventual wider adoption of proof-of-stake consensus algorithms may ameliorate this problem, but public permissionless blockchains pose difficulties of their own, independent of the consensus algorithm. The permissionless, public, and geographically unconstrained nature of the public blockchain makes it difficult to regulate via laws such as the GDPR. As things are, the power system is not P2P: there are definite roles to its participants and definite limits to what they may or must do, and the system modeling this for the purposes of trade should adapt to this reality. Of note is the difficulty posed by the need for a trading system to have good access to external sensors and other fundamentally centralized subsystems. This is impractical for most public blockchain implementations, and, while strides are being made with technologies like Chainlink and other oracle providers, most of the technologies are not ready for production use, especially in something that needs to function as flawlessly as a system integrated with the power grid. Further, each such system causes costs and issues to increase.

Some solutions adopt private permissioned blockchains based on a consortium of users [8,9]. We believe that this is the correct direction; however, these solutions also have their issues. The system is not expensive compared to public blockchains, it has a much larger throughput, and integration with external centralized systems is an order of magnitude simpler. Here, the problem is chiefly on-ramping users: for most private permissioned blockchain solutions, joining requires a steep burden when it comes to server and organizational costs. Most of these solutions were designed for serious enterprise use, which means that direct participation by prosumers that would otherwise be easy in a public blockchain becomes prohibitive.

As we've said before, the difference in approach is not solely technical. Rather, it is the case that we had different intentions than the authors of other solutions, especially regarding adoption gradients and the possibility of undirected growth within the system. All the solutions surveyed have one of the two fundamental blockchain risks we illustrated in detail above: either they are expensive at scale as all processing time must be bought from a public blockchain at a non-negligible cost [20] or they are expensive both at scale and at the start due to the substantial costs of running blockchain node instances for every participant when private distributed ledgers are used [8,9]. This is not an issue with the solutions as such: merely an attendant risk of using DLT technology in a real-world context.

We would like to propose an alternative approach to earlier works on the topic of blockchain-based energy trading such as [8,9,19,20]. Our approach is centered on its ability to make a maximum of sense for both the producers, the prosumers, and the electrical distribution companies at any scale. A system that might be worth joining and using even if you are one of the first users, and something that can grow and expand on its own merits without requiring too much in the way of investment and too much in the way of changing the problem to fit the solution.

Ultimately, the goal of this paper is to describe a solution that can be implemented here and now and can be useful immediately. We believe that it is necessary to let the system grow and develop under real-world conditions and for that to be a possibility it must make economic and logistical sense immediately. To this end, we operate under a few limitations that also make our work distinct from what others have presented in the past:

- There are no tradable 'coins' or other such tokens in the system we propose. No currencies except those already in existence are used.
- We do not require a microgrid or any specialized infrastructure, though the solution can be adapted to use both, but rely solely on present, common grid implementations.
- We do not use any mining or proof of work and impose no computational/resource costs above what a traditional system of servers would impose.
- The solution we propose does not require significant alterations to the legal framework.
- The solution is closed, and all data flow is controlled, requiring no more effort to secure the integrity, confidentiality, and availability of the system above that required to secure a system not based on DLTs.

On aggregate, these changes allow for a completely different kind of decentralized trading solution.

3. DREAM: A Different Kind of Decentralized Power Trading

This section is composed of four subsections. First, we present the overall structure of the proposed system and what it is meant to do, as well as a desired use-case. Afterwards, we describe the classes of participants that we identified as part of the DREAM network. The details of the system architecture are elaborated upon in the third subsection. The fourth and final section presents the inner workings of the DLT subsystem of DREAM in greater detail.

3.1. System Structure and Goals

For the purposes of this paper, we will mostly use a specific environment that was chosen to be realistic but simple. The simplicity is there to make the DREAM architecture and prototype behavior easier to follow and understand. The environment is characterized, above all, by a single distribution provider with a single unified grid connecting every participant in the system. This distribution provider then serves as a common carrier and, since in our model it is also a producer, it will serve as the power provider of last resort. This is a condition which, to a greater or lesser extent, is close to the actual situation on the ground in various parts of the world where until recently regional or state power companies held de facto monopolies. The system then supports multiple producers, multiple prosumers, and a great many consumers, a breakdown that closely matches what we expect in the early stages of product adoption.

A problem for any energy trading system is that energy storage is limited and introduces unavoidable losses, and the network is such that the inputs must match the outputs. Therefore, at no point can there be a shortfall or an excess of power in the system as they can cause faults in the supply or even complete or partial grid collapse. The implications of this are these: the only things that can be traded are, in financial terms, futures. Specifically, each trade consists of one party promising to supply a given amount of power to the grid, and another promising to *use* that given amount of power. Since the system relies on those promises to function, the question arises: what to do if they are broken? Therefore, the system needs a failsafe to guarantee the integrity of the power grid. In the scenario we are using, this is the power company itself, which is a producer in its own right and controls the distribution system. Its job will be to either store or destroy power in the case of shortfalls or surpluses, respectively.

Of course, reliable, scalable power storage technology is under intense research and as its share of the grid expands, the trading system can expand to adapt to it. Fundamentally, in the present system an energy-storage provider would function much like a prosumer, both 'consuming' electricity (by storing it) and 'producing' it (by releasing it from storage). Such participants would operate in exactly the same way, technologically, as other prosumers, but their role, economically speaking, would be quite different, providing stability and liquidity to the market in direct proportion to their storage capacity and efficiency.

Given the situation described, especially before the advent of suitable energy storage at scale and without the use of a specially built infrastructure, market participants cannot in truth send power to each other. With the way the grid works they can only receive power or send it. The power grid makes a deal with its producers and prosumers about its obligation to accept production surplus (at a nominal rate) and provide agreed-upon power to all consumers (at a specified rate). Depending on technological and economic limiting factors, this deal may also have provisos about the limiting of production, say, when there is so much surplus the grid can no longer accept it. This agreement represents the floor of the market and guarantees that the failure state of the solution is a system operating exactly as it does now.

Anyone producing electricity can create a power promise. This is a promise within the system to produce and provide to the grid a given amount of energy over a given time. This promise can then be the subject of trade and is meant to be auctioned off. When the transaction is complete, it means that the producer (or prosumer) has made the promise to supply the specified energy over a given time and that the purchaser has promised to accept the given power in the given timeframe. If both sides keep to their side of the bargain, this means that the transfer is a success and that a suitable amount of money is transferred internally. If either side defaults on its promise, the grid authority is responsible. This is necessary for the grid to continue functioning smoothly but should be avoided as much as possible as it demands the costly maintenance of reserved energy generation or storage capacities. The more often a power promise is broken, the more often the power output must be tweaked to cover the shortfall. This imposes both real and opportunity costs on the power grid and should therefore be discouraged. Therefore, two things are needed: a limit to the position a seller of power promises can take (therefore limiting the risk) and a disincentive to defaulting on the promise. Both conditions can be met at the same time by using an escrowed deposit system.

The grid authority has a known-in-advance schedule of costs for emergency power supply. This is the price per kWh for power supplied due to a seller defaulting on a promise. It covers all the costs for the grid authority and is further increased as a strong economic incentive to not make promises that cannot be reasonably expected to be kept. As part of the mechanism for making a power promise, the seller must deposit the total cost of this penalty for the amount promised. This stake represents a guarantee of the seller's bona fides and reduces the probability of deceit. If the seller should default, the amount in escrow is forfeited. If they should succeed, any service fees will be deducted and returned to them.

The way this prototype system serves everyone is that it allows the consumer to buy electricity at a more advantageous rate than the default cost offered by the power distribution company. Likewise, it allows the producer and, more importantly, the prosumer to sell their surplus at prices that are higher than the nominal fee paid to them normally. The margin between the price paid for surplus power and the price paid for power from the distribution company is where trading profit can be made.

3.2. Network Participants

In the DREAM prototype implementation, we identified the following classes of participants in the business network:

- **Grid Authority**. The grid authority is any institution that is responsible for the grid and monitors the sensors attached to it. The fact that this is a singular entity everyone trusts is antithetical, admittedly, to the spirit of the decentralization of blockchains, but in most cases there is only one grid and thus only one institution responsible for it. Therefore, decentralizing all of its functionality is at present impossible. The decentralized nature of DLT solutions, however, will permit any future organizational changes that permit greater decentralization to be implemented.
- **Power Company**. The grid authority is only responsible for the grid itself and monitoring it. A power company, however, represents a commercial enterprise that is engaged in producing power or buying it in bulk from other producers (or both) and then selling it according to, typically, a fixed schedule of costs. In the prototype, it is assumed that the distribution system is monolithic and so this role is fully integrated with that of the grid authority.
- **Producer**. An enterprise primarily engaged in the production of electricity for consumption. A producer can be of any size encompassing both relatively small producers, typically of renewables, and networks of vast power plants. Producers typically have bulk contracts with power companies. Therefore, it is possible that they may be part of the grid but have no legitimate reason to participate in the DLT solution.
- **Prosumer**. Unlike a producer, a prosumer is a private individual not an enterprise and is engaged in the generation of a small amount of power primarily for personal use while hoping to sell the surplus. Empowering prosumers and making it easier and more attractive to become one is one of the main design goals of the system.
- **Consumer**. A private individual or company who needs electrical power and wishes to purchase it.

3.3. System Architecture

This subsection deals with the software architecture of the system under the assumption that the grid already allows us to track the power consumed by any given user and the power emitted into the system by any given user using built-in sensors. This, given modern infrastructure, is true in many cases and certainly for all prosumers.

The architecture of the prototype system is composed of two fundamental subsystems and it is presented in Figure 3. The first subsystem is the core of the solution and it is built using the R3 Corda private permissioned distributed ledger [14]. The second subsystem is web-based, and it serves as a wrapper around the Corda DLT core. It offers an easy-to-use user interface.

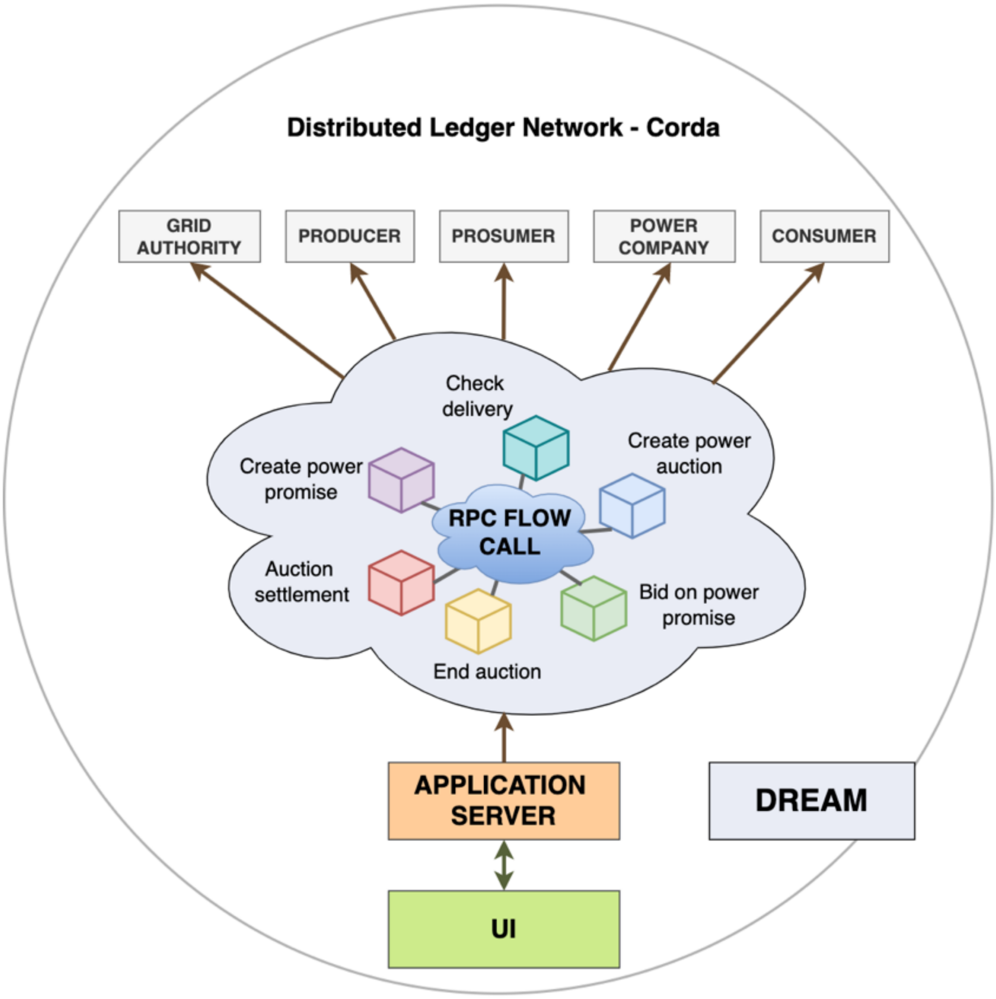

Figure 3. A high-level overview of the architecture of the DREAM prototype.

The distributed ledger subsystem is fundamentally a set of protocols implemented through specialized software that permits any number of independent computer systems, both actual and virtualized, to participate in the system by helping to store its data and process changes to the system. A computer implementing the relevant software and protocols is termed a 'node' and is the primary constituent of a DLT system. In the prototype, every participant in the system maintains at least one node of their own, which remains under their complete control and guarantees that no changes concerning that participant can be made in the system without said participant concurring according to the rules of the system. This communication takes place without any centralized server representing 'the system'. Rather, the system arises from the organized communication of the network participants who communicate in a P2P fashion.

For the purposes of the DREAM prototype, we implemented the following five main classes of participants (please see Figure 3): grid authority, power company, producer, consumer, and prosumer. The prototype P2P network implementation features a single instance of each of the five classes of participants due to the memory limitations of the available test infrastructure. This limitation is entirely due to hardware constraints: with sufficient resources (which, per-node, do not exceed the requirements of a moderately demanding conventional application [32]), the network could be expanded further to suit different requirements (see Section 4.2 for details).

The node with the most responsibilities is the one maintained by the grid authority participant due to its vital role in managing the power grid itself, specifically its role in bridging the gap between the digital representation of the system and the physical world. Power delivery and power consumption are managed by the digital system, naturally, but this would be a futile effort if there is no way to establish that what has been agreed upon and paid for in the digital system has actually happened in reality. In the DREAM prototype, the grid authority is maximally centralized for performance and simplicity reasons so that it represents the sum of the system's interactions with reality. Section 4.2 contains some details on how this may be improved upon to limit the power of a single node.

The power company node in the DREAM prototype is mostly of secondary importance in the organization but is fundamental to how the system operates because its buying/selling behavior (serving as the default power supplier) provides a guarantee of grid continuity and provides a floor to the market—both vital functions. Likewise, producers are vital to the grid but in the digital system their role is to produce and sell conservative power promises. In our prototype, our profile of a typical producer who participates not only in the grid but in the DLT system is most likely a small enterprise producing renewable energy at a modest scale.

Consumers are the end users of electricity and their role in the system is to be the final purchaser of power promises to attain electricity for personal or commercial use. Their role in the DLT system specifically is to bid on power promise auctions to get electricity that is more affordable or, as the prototype expands, has other beneficial properties.

Prosumers can play the role of both producer and consumer. They represent an increasing number of power grid participants who would traditionally just consume energy but now have renewable energy production capability. In most cases they use the energy they produce, but if there is a surplus of energy, they can also sell it. When they produce an insufficient amount of power for their needs, instead of selling the surplus they can buy more in the same way that any consumer does.

Users who manage nodes can access them through a web server. A design goal of the system is to minimize the work done by a 'back-end' component or rather to make the DLT system itself the back end. The web server is there to provide a suitable interface. In the prototype only one server is used to manage all the nodes in the system, with the node selected for management determining which functionalities are available. As the prototype expands, of course, specialized servers can be employed and optimized for the needs of individual participants.

3.4. The DLT Subsystem of DREAM

The main purpose of a distributed ledger subsystem is to store sensitive data and to ensure that this data cannot be changed without this change being indelibly recorded on one hand, and allowing multiple independent parties to verify that the change of the data was done according to pre-arranged rules on the other. There are several technologies implemented in this approach. We have selected R3 Corda, a private permissioned DLT solution with a semi-open structure, for implementation, and it is necessary to briefly explain why we made this choice. The blockchains most commonly mentioned, such as Bitcoin [3] and Ethereum [13], are public, meaning that any operator may create a node of their own and, as long as they follow the protocol of the blockchain, they become a part of its operation. This is maximally scalable and decentralized, but it is a very poor fit for the legal framework in which such a system must operate. For the whole mechanism to function adequately, identities within a system must be associated with full legal identities and with specific pieces of power transmission and generation hardware.

Thus, a private ledger is required. However, we propose that the best results would be achieved with a semi-open design in which anyone fulfilling certain requirements is allowed to operate a node and join the network. Furthermore, it is evident that the relations between the various participants of the network are asymmetrical with different roles apportioned to different participants. Therefore, we cannot employ a permissionless distributed ledger as our basis, and as such a DLT assigns the same role and gives the same rights to all its participants. Thus, we need a private permissioned distributed ledger in which different roles, permissions, authorities, and rights are given to different participants. There are multiple implementations of such DLTs available [10,14,15]. The choice of Corda in our system was motivated by the goal of creating a lean distributed ledger that complies easily with industry standards and can operate on a "per transaction" level. In such a setting, Corda is found to be a superior solution over Hyperledger Fabric [32,33], which has been used as the basis of other similar energy trading systems based on private permissioned DLTs [8].

Safe and secure data storage represents one half of the core function of a distributed ledger like Corda. Each network node has its own database (vault) in which it stores states contained in transactions it has participated in. A state is one way of storing data: it is one fact about the system that can be marked either as 'spent' or not, meaning it is either extant in the system or has been altered in some way. Transactions are recorded changes to data performed through flows that function as what is called a 'smart contract' or 'chaincode' in other networks. Transactions operate by transforming certain states (spending them in the process) into other states. Smart contracts, on the other hand, serve within R3 Corda only to enforce limitations on which states and transactions are valid, which is to say, unlike typical smart contract implementations, they operate far more like their real-world counterparts.

DREAM uses two states to store data. The first is a "power promise" (PP), which holds grid-related information (amount of power in kWh, duration of power delivery, delivery date, delivery time, owner, supplier, and whether power was delivered as promised or not). The second one is an "auction state" (AS). It stores all information relevant to the selling of power promises (base price, highest price, highest bidder, bid end time, winner of bid, and auctioneer).

As we mentioned before, contracts in Corda are slightly different from smart contracts in other distributed ledger and blockchain platforms [13,14]. They emulate real-world contracts and ensure that rules associated with transactions are imposed. In our prototype, contracts ensure that the amount of power offered in a "power promise" is not too small and insignificant and not too big and unrealistic. Contracts also ensure that auctioning is handled properly by ensuring that only the owner can create an auction, only the highest bidder can become the new owner, and that money is sent after ownership transfer.

Communication between parties in Corda is also implemented through flows (please see Figure 4). It can be thought of as unfinished transaction sharing between nodes. Initiating flow parties creates transactions using the TransactionBuilder framework class. It reduces the complex creation process to a simple listing of essential transaction parts: states, contracts that govern states, and notaries. Notaries are specialized network participants introduced by Corda R3. They are meant to verify transactions, monitor the state lifecycle, and generally prevent fraudulent protocol participation, double spending, and forestall these sorts of problems. The party that initiates the flow also states who the transaction participants are, aside from the notary. Participants are particularly important where, as part of formulating a transaction, a digital signature is needed. There are also observer parties that monitor transaction validation and save it in their vault. All transaction signing and propagating mechanisms are integrated into the Corda framework in the form of predeveloped sub flows. This allows developers to handle complex RPC (remote procedure call) operations between multiple distributed nodes without major difficulties.

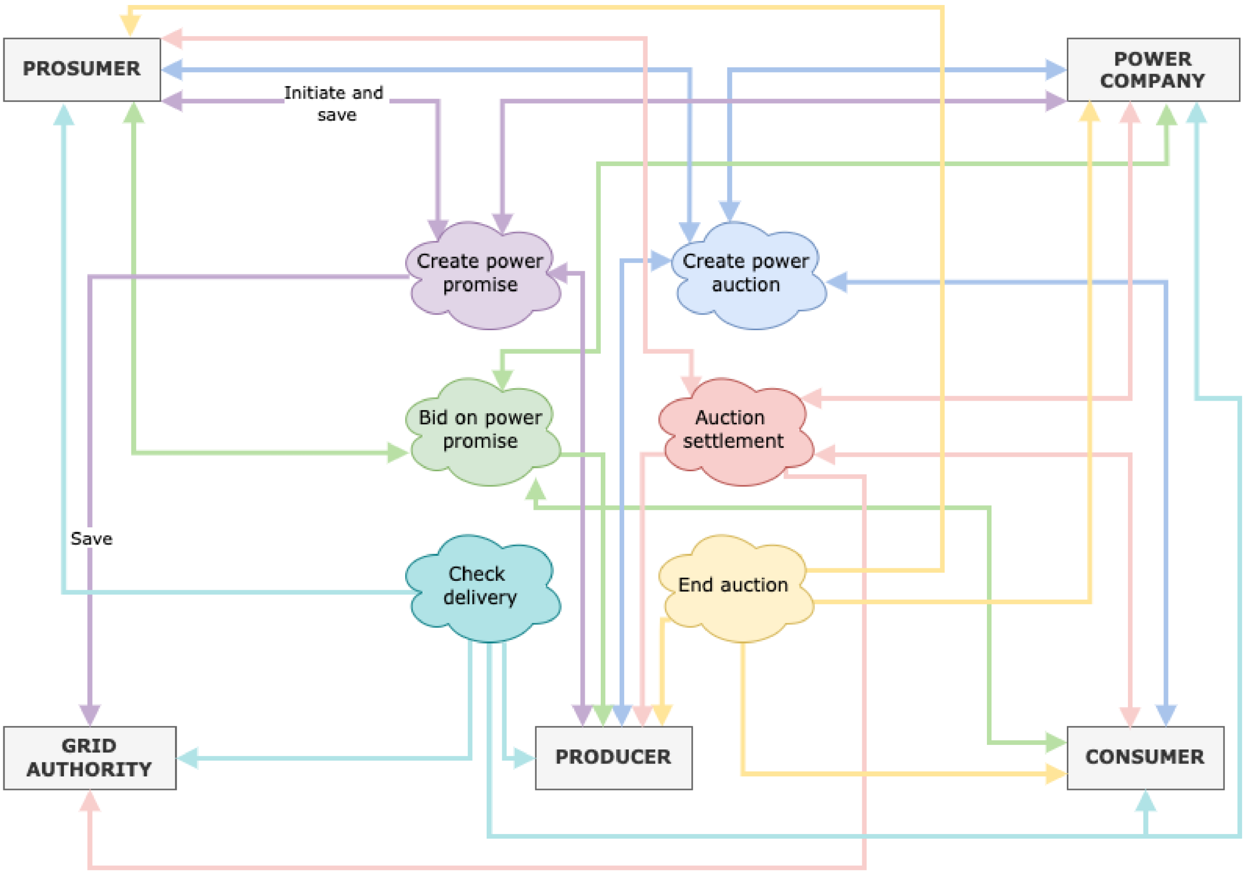

Figure 4. A high-level overview of Corda flows in the DREAM prototype.

The Corda flows implemented in the DREAM prototype are presented in Figure 4 and are defined as follows:

- **Create power promise—**represents the beginning of the whole power trading process. When parties anticipate surplus in electrical energy production, they initiate this flow. By doing so, they create a "power promise" (PP) that some amount of electrical energy will be injected into the electrical grid at a specific time. This PP is later traded in the network. The party that owns the PP at the delivery time is allowed to spend a promised amount of energy. The flow's product is a transaction that contains the initial PP state. Transaction participants are the party that creates the PP and the grid authority. The grid authority monitors the power grid and ensures that the promise is fulfilled when the delivery time comes. Only the producer, prosumer, and power company parties are allowed to initiate the creation of a power promise flow because only they can insert power into the electrical grid.

  The pseudo code for this flow is shown in Appendix A as Algorithm A1. This flow's logic starts by creating a transaction builder and specifying a notary. The second step in the prototype is randomly determining whether a PP is fulfilled or not. The next step is influenced by the outcome of step two. If PP is fulfilled, the third step is sending deposited funds back to the supplier. If PP is not fulfilled, the third step is to send deposited funds to the Power Company that delivers electricity instead of the supplier. The fourth step is to change PP and record which of the two options occured in the third step. The fifth step is signing the transaction and sending the transaction to be signed to the supplier if the PP is fulfilled or to the power company if the PP is not fulfilled. The sixth step is receiving a signed transaction from the counterparty. If the transaction is valid then it will be saved in the vault. If the transaction is not valid, the flow will produce an exception. In that case, the transaction will be discarded, no data will be saved in the vault, and the participants will be informed of the problem.

- **Create power auction—**represents the first flow in the process of PP trading. To initiate this flow, a participant must be the owner of a PP that has not expired. A "power promise" expires after the delivery time has passed. By initiating flow, PPs are put into auction. The owner must state the base price of the PP and the auction deadline. The product of the flow is an "auction state" (AS) that will be used by bid, auction settlement, and auction end flows to carry on PP trading. Flow initiators can all be PP owners. This excludes only the gird authority from the list of network parties. All other parties save the grid authority also participate in the transaction, but their signature is not required to validate it. They are only observers, which means that they save transactions in their vault and are updated on any state change.

  This flow's logic starts by creating a transaction builder and specifying a notary. The second step is creating an AS. The next step is the signing of a transaction by the initiator. Due to this step, other observer parties can make sure that the AS that is offered and used to sell PP is indeed signed by the PP's owner. The fourth step is sending the transaction to observers for them to save the AS contained in the transaction in their vault. The final step is saving the AS in the initiator's vault.
- **Bid on power promise—**represents a flow that can be used after the AS has been created and is flagged as an active AS. The initiator of this flow wants to offer a higher price for the AS to become the highest bidder. Prices offered by the initiator must be higher than base price in the case of the first bid or higher than the best price so far the in case of subsequent bids. If a higher price is offered, the initiator becomes the new highest bidder and earns the opportunity to become the new owner of the PP when the auction deadline is reached. The flow initiator can be any party except the grid authority and the AS owner. Parties that are not the initiator or the grid authority can serve as transaction observers, verifying that the rules are followed and storing the transformed data within their vaults.

  This flow's logic starts by creating a transaction builder and specifying a notary. The second step is checking if the new price is higher than the previous highest price. If that is not the case, the transaction is rejected, and no states are changed. The third step is modifying the AS state by acknowledging a new highest bidder. The next step includes placing the modified AS in a new transaction. Then, this transaction is signed and dispatched to all other observing parties.
- **Auction settlement—**represents the final flow in the power trading process. After the auction deadline is reached, the highest bidder can initiate the auction settlement flow. This flow removes funds from the initiator's account and sends them to the PP's current owner. It also modifies the PP state by making the flow initiator the new owner. The transaction participants are the initiator or the party identified as the highest bidder for the AS, the party that is the current PP owner, and the grid authority. The grid authority monitors the power delivery in the physical grid and thus it must be informed about the PP ownership change. The old PP owner is required to sign a transaction for all network participants to acknowledge the ownership transfer.

  This flow's logic starts by creating a transaction builder and specifying a notary. The second step is transferring funds from the new PP owner to the old one. The third step is modifying the PP state by making the initiator party the new owner. The fourth step is creating a transaction and making the modified PP state its central part and signing this change. The fifth step is sending a self-signed transaction to the grid authority and the previous PP owner for signing and validation. If they provide counter signatures, the initiator collects them and finalizes the transaction. All participants save the modified PP state in their vaults.
- **End auction—**represents a scheduled flow in the power trading process that makes AS inactive and prevents further bidding. When the auction deadline is reached, the end auction flow is automatically initiated by the AS owner. The flow logic is simple as

it only changes the AS state flag from active to inactive. All network participants who are AS observers are informed about the auction's end by receiving the transaction with a modified AS object, which they save to their vault.

This flow's logic starts by creating a transaction builder and specifying a notary. The second step is modifying the AS to make it inactive, then the transaction is signed. The fourth step is sending the newly signed transaction to all observers. Finally, the transaction is saved in the participants' vaults.

- **Check delivery—**represents the final flow in the whole power delivery process and is a scheduled flow. Before the delivery time expires, the PP can be traded in the network. This flow is scheduled to automatically initiate itself on the grid authority nodes when the delivery time is reached. This flow is essential as its role is to connect the real-world electrical grid with the DREAM system. The prototype does not engage any sectors as it is still a technology testing platform and instead randomly registers certain transfers as 'failed' to test the system under anomalous conditions. The full version, however, will rely on sensors that track and attribute all production and consumption. This will make it easy for the GA to ascertain that one party produced a certain amount of energy, and that the other party used said energy. If the promise is detected as having fulfilled itself, the money held in escrow is released back to the original producer, minus a minimal service fee. If the promise is not fulfilled, the deficit in the power generation will be taken care of by the supplier of last resort, who will be paid the full amount (minus fees) of the escrowed funds.

  The pseudo code for this flow is shown in Appendix A as Algorithm A2. This flow's logic starts by creating a transaction builder and specifying a notary. The second step in the prototype is randomly determining whether a PP is fulfilled or not. The next step is influenced by the outcome of step two. If the PP is fulfilled, the third step is sending deposited funds back to the supplier. If the PP is not fulfilled, the third step is to send the deposited funds to the power company that delivered the electricity rather than the supplier. The fourth step is to change the PP and record which of the two options occurred in the third step. The fifth step is signing the transaction and sending the transaction to be signed to the supplier if the PP is fulfilled or to the power company if the PP is not fulfilled. The sixth step is receiving a signed transaction from the counterparty. If the transaction is valid then it will be saved in the vault. If the transaction is not valid, the flow will produce an exception. In this case, the transaction will be discarded, no data will be saved in the vault, and the participants will be informed about the problem that occurred.

4. Discussion

We have built a prototype solution that shows that the underlying technology is adequate, and that the central idea has merit. The basic functions of the trading system operate as intended. The structure of power promises and auctions fits well into the DLT framework, and it appears that they can be secured adequately. However, this is only the beginning of the research program necessary to put this into full use. This section is, therefore, divided into four subsections: prototype performance analysis, chief development risks, prototype augmentation, and future research. In the first subsection, we present and analyse the experimental results considering the prototype performance. The chief development risks subsection details the problems that may derail the implementation of the DREAM system or render its use impractical or even detrimental. The prototype augmentation subsection discusses the alterations we see ourselves making to the prototype to bring it further in line with our design goals for this project. It represents the improvement and development work within easy conceptual reach. The research sections represent more abstract goals: fewer concrete improvements and more work necessary to expand the conceptual reach of the project and ascertain vital facts about its practical future use.

4.1. Prototype Performance Analysis

To establish the scalability of the solution in practice and start the work of determining the likely infrastructure costs of a deployed solution, we have performed a series of experiments to measure the real-world performance and resource consumption of several different computer topologies. Due to technical limitations, all tests were performed on one computer system where multiple nodes were executed as independently orchestrated processes. The experimental protocol was executed on a computer with an Intel Core i5-9400 CPU with six cores and six system threads running at 2.9 GHz, 16 GBs of system RAM, and an SSD HDD running Ubuntu version 20.04.

Each experimental run varied the rough network topology (six, eight, or ten nodes total), the total number of simultaneous requests sent for processing (1, 10, or 100), and the task selected for processing (creating a power auction, creating a power promise, and bidding). When collecting the data, individual executing processes were instrumented and data on their performance was collected. The performance of these processes was then aggregated based on 'usage classes', which indicate what the purpose of the individual process was:

- **Infrastructure:** processes spawned as part of the orchestration of the system dealing with executing sub-processes.
- **Client:** processes executed as a part of the test that sent requests to the system as part of the stress test.
- **Grid Authority:** processes responsible for maintaining the grid authority node.
- **Consumer:** processes responsible for maintaining the nodes of all the consumers that are part of the system.
- **Producer:** processes responsible for maintaining the nodes of all the producers that are part of the system.
- **Prosumer:** processes responsible for maintaining the nodes of all the prosumer nodes in the system.
- **Power Company:** processes responsible for the power company node.
- **Notary:** processes needed to run the notary node that is a necessary part of Corda systems.

The parameters tracked for this experiment were time taken for batches of requests (usually expressed as time per request to operate with commensurate quantities) and memory occupied. When computing memory usage, the resident set size was the metric selected for instrumentation, as it correlated the best with actual physical RAM used. To minimize noise, the entire network was restarted and redeployed before each test run, therefore isolating each run from the others.

While these tests are not fully realistic insofar as they execute on synthetic data and only one computer system, they represent likely trends in the scaling of resources used and demonstrate scalability. Power consumption was not a tracked parameter because Corda is not a proof-of-work DLT, has no mining operation, and thus requires no more energy than any other comparable piece of software. Therefore, the power costs are no greater than ones that are silently counted as part of normal hosting costs. This property of Corda was one of the reasons for selecting it for our prototype development.

Figure 5 demonstrates a graph of memory usage versus the number of parallel requests. Given the complexity of the test protocols, the diagram has been faceted so that columns are different usage classes and rows are different tasks. 'Create PP' is an abbreviation of 'create power promise'. The image shows promising scalability with the number of requests, which does stress the system but does not affect the memory consumption to a significant degree. The only increase can be seen in network topologies where certain instructive phenomena may be observed.

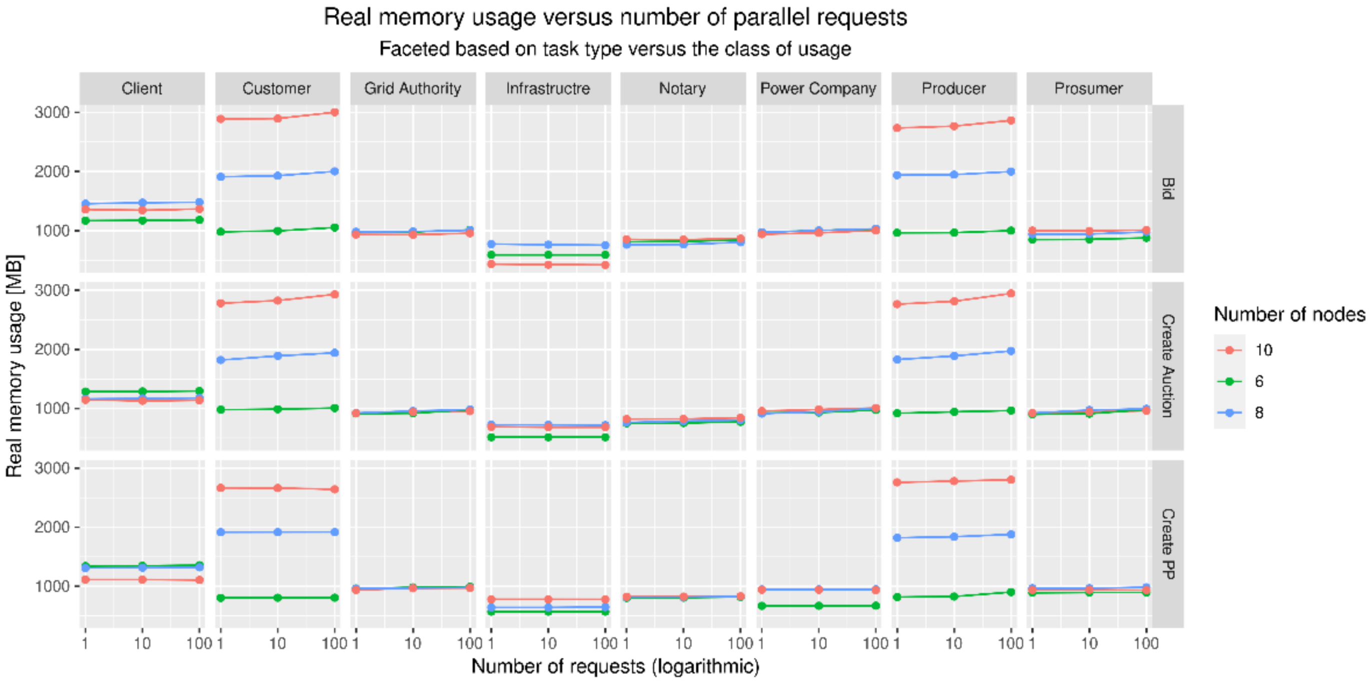

Figure 5. Plot of memory usage versus parallel requests faceted over tasks and class of usage. Color indicates the number of nodes.

The memory graphs for producers and consumers are very similar because those are the replicated nodes in the multi-node topologies. The equal spacing and equally shaped dependence curves show that the scaling of memory in this instance is purely linear: the more nodes there are, the more memory will be required in strict proportion. Other nodes show no strong correlation and are either identical, as in the case of the prosumer, or show minor differences, e.g., in the case of the infrastructure usage class. It is likely that the orchestration process lasted long enough for the garbage collector pass to occur.

Given the difference between usage classes, it may be instructive to observe Figure 6, which shows the relative use of memory between usage classes as faceted by the number of nodes used and the task type. What this figure readily shows is that there is a change in proportion depending on network topology. This is to be expected as it results directly from creating new processes for new nodes of a specified type. However, when tracking changes in memory distribution due to changes in request number or changes in tasks being processed, no noticeable change in distribution can be observed. This indicates that the relative memory requirements of various subsystems are stable.

The relationship between time and the factors we have been studying thus far is more interesting. Figure 7 shows the relationship between time required per transaction and the total number of requests faceted by the number of nodes (for columns) and the task being performed. Certain patterns are instantly noticeable: the behavior of request time versus the number of requests is largely unchanged in systems with more complex topology. The shape of the line remains largely the same, even though the line is shifted slightly up, indicating a performance penalty for more complex networks.

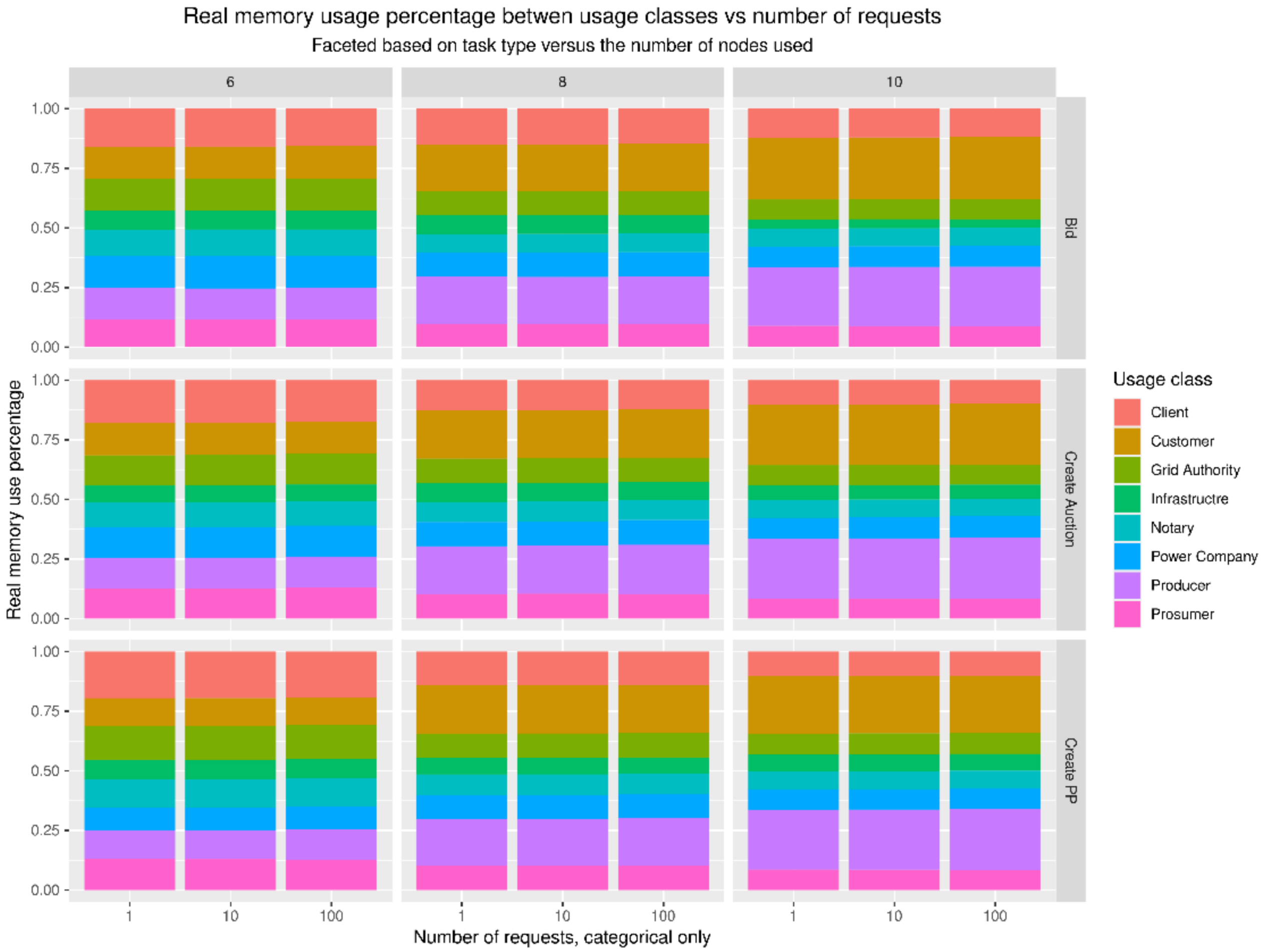

Figure 6. Proportion of memory use per usage class versus the number of requests faceted based on the number of nodes involved and task performed.

For bidding and creating auctions, the time per request falls as the number of requests increases, suggesting that there is an initial setup cost to the transaction that is amortized across multiple requests. Creating a power promise, however, does not exhibit this kind of behavior and actually becomes slower with each request. This indicates a bottleneck operation based on code analysis and is likely the creation of the power promise itself.

This measurement shows that there are some conclusions to be drawn and some research paths that need to be explored before the system is implemented. The system is fairly stable when it comes to memory use, which helps with provisioning in the case of wider-scale deployments. The limiting factors, and therefore development risks, seem to be the increased times in the case of more complex network topologies and the behavior of request time with the 'create PP' operation.

It should be possible to handle the problem with arbitrarily large network topologies using Corda's ability to form ad hoc groups of participants and restrict transactions to just those hosts, thus effectively creating micro-DLTs as they are needed. This should reduce the amount of communication needed and may serve to minimize the performance hit of large networks. However, further testing is needed.

Creating power promises, likewise, requires significant further testing. The behavior of bids and creating auctions shows that there is no technical reason to expect operations to become slower as they are processed in large numbers. It is likely only the case that there is a possible optimization in the code that has not been implemented yet.

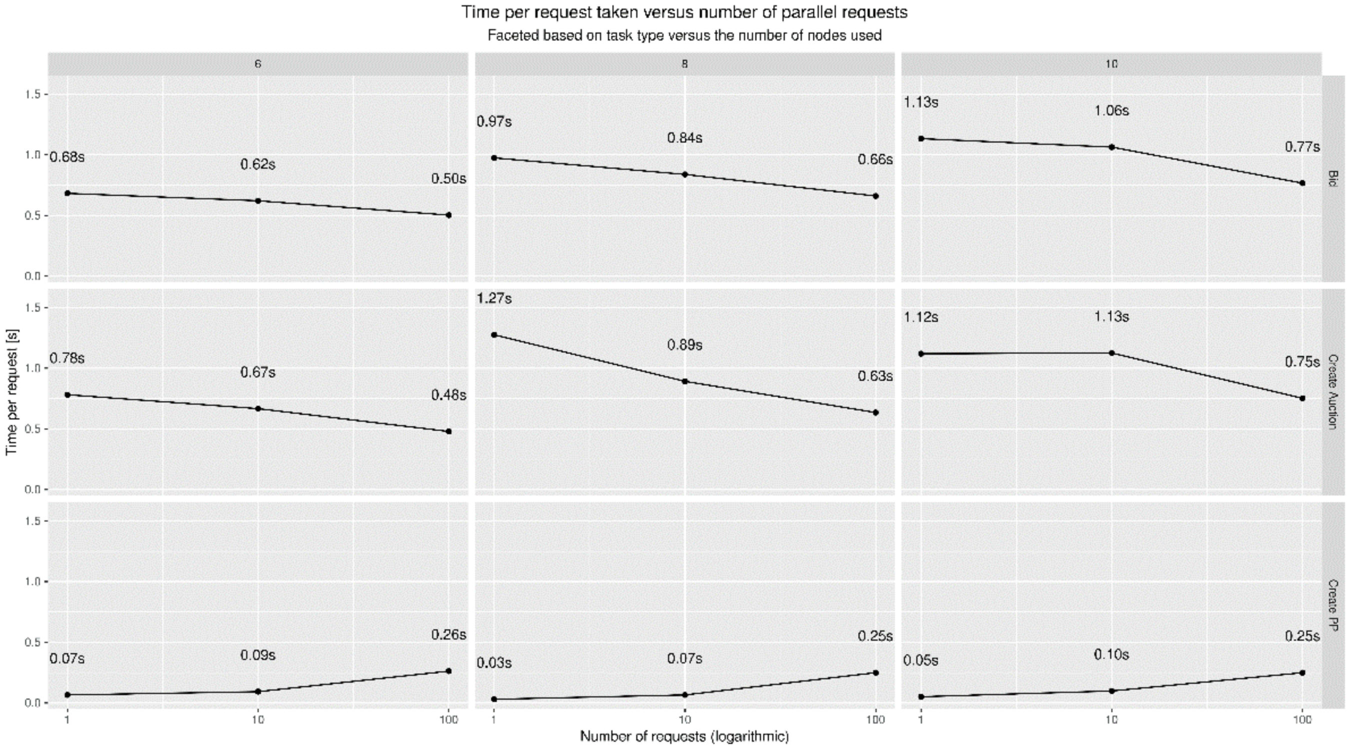

Figure 7. Time per request versus number of requests faceted based on number of nodes and tasks performed.

Overall, however, the system shows excellent behavior under strain and does not demonstrate worrying requirements when it comes to processor time, RAM, or power consumption. With proper engineering, everything indicates that it should be possible to create the necessary infrastructure without undue financial or ecological cost while providing satisfactory performance.

4.2. Chief Development Risks

DREAM has been engineered to sidestep most of the problems that derail the wider use of decentralized power trading systems. It can be changed to fit most regulatory regimes, it is scalable and should require comparatively little in the way of resources. It is not without drawbacks and risks, however—chief among which are the **rapid grid transformation**, **single points of failure**, and **economic barriers**.

Rapid grid transformation. While DREAM can adapt to both microgrids and other specialized infrastructure designed specifically to support peer to peer power trading, the fact remains that it is deliberately designed to work without any such system existing. This means that taking advantage of the unique possibilities such a system provides needs to be engineered into the system, and here DREAM may run into the problem of being outcompeted by a competitor that makes full use of such a functionality, provided such a competitor is implemented parallel with a massive grid overhaul. This would interfere with the operation or implementation of DREAM and would make its use economically untenable. If DREAM is fully implemented by the time grid undergoes such a transformation, there will be time to adapt it to fully make use of the changes.

Single points of failure. As described, DREAM relies on a few mechanisms, largely those related to the power grid, to operate flawlessly for it to function. The fact that one cannot trade power while the grid is inoperable is not worth remarking upon, of course, but DREAM also relies on the power grid system for its source of truth regarding who produced how much, and this information is both crucial and time-sensitive. Without the sensors operating accurately and without delay, DREAM cannot work. This drawback can be staved off by ensuring a very reliable power grid and very reliable sensors, but ultimately the only way it can be truly resolved is if there are multiple sources of truth and independently operating sensors in the network, allowing the system to operate, albeit at a reduce capacity, even in cases of a partial outage.

Economic barriers. One drawback to not minting and using a token to represent power traded and not listing it on a public blockchain is that this inevitably leads to power being more thinly traded. A very large audience already trades various cryptocurrencies and this represents a natural market for power-related tokens as well. If participating in a closed system is necessary for trading, like with the DREAM architecture, the market will be stagnant until a critical mass of users start trading within the system. A sufficiently fast uptake of users, perhaps through some sort of institutional backing, removes this risk. Another approach may be finding a way to create a token linked with the closed system and freely tradeable on most major blockchains. The tokenomics of such a token would require more research, but this might allow for an additional popular secondary market.

4.3. Augmenting the DREAM Prototype

The DREAM prototype serves to only test the fundamental technology of the system and provide a path towards further development. It is easy, however, to expand it with several features or infrastructure augmentations that would make it better suited for transformation into a system ready for real-world deployment, albeit at a smaller, more manageable scale. The modifications suggested here as promising avenues for future development work are **aggregator nodes, full power fungibility, rich power metadata, support for more specialized participants,** and **trading automation.**

Aggregator nodes. Typically, each participant in a P2P network of this type would have its own node, i.e., a computer, virtual or actual, under its control that implements the DLT protocol suite and makes sure no state change gets written into the ledger that runs contrary to the agreed-upon rules. The prototype is built on the assumption that all participants maintain one such node. This, however, is impractical in the real world. Certainly, the grid authority and the large companies can maintain their own servers. The cost for a node is not significantly greater than for a reasonably scaled application server and for a large enterprise and is not a significant cost. An individual prosumer, however, is already operating on slim margins of profit: the surplus generated is a windfall, not a steady income, and the costs of running one's own node are constant and they mount. Further, maintaining a node is a nontrivial task and conflicts with a design goal of the project: that the solution be turnkey and require no technical sophistication on the part of the end-user.

Therefore, we propose the implementation of aggregator nodes. These nodes function in place of the individual nodes of arbitrary subsets of participants. In essence, these nodes serve as the personal node of more than one person and represent their interests in the distributed ledger system. They do this by holding in trust a key derived from the private key of the prosumer. Transactions signed with this key are counted as signed by the prosumer, giving the aggregator node the 'power of attorney' for these purposes. However, the prosumer retains control because they can terminate their relationship with an aggregator node at any point with a single on-chain transaction, revoking the power of the derived key to verify transactions. This can prevent an aggregator node that has acted maliciously or has in any way lost the confidence of the end-user from doing any further damage.

To make aggregator nodes responsive to user wishes, the protocol of the network can be such that aggregator nodes must, upon request, relinquish all data they have on a given participant they are managing in an agreed-upon format. An aggregator node can be operated like a business, essentially selling blockchain access in much the same way that an ISP sells internet access. These two technical solutions (easy key revoking and easy data mobility) mean that migrating from one aggregator to another is trivial, allowing for market mechanisms to swiftly deal with underperforming aggregators.

Full power fungibility. As things stand, a power promise is traded whole: someone promises a certain amount of kWh and that whole promise is traded as-is. It would be entirely possible to allow any given non-expired PP to be split apart into multiple PPs whose promised power generation exactly matches that of the original PP. Likewise, it should be possible to code a mechanism by which multiple promises can be combined into one larger PP. This fungibility would simplify more complex trades and would also permit the implementation of a system in which a prosumer/producer dealing with technical malfunctions can avoid paying penalty costs for a missed delivery. If PPs may be amalgamated, it should also be possible to implement PP substitution in which an entity holding a PP from a third party (or a fraction of one, or a combination of multiple given the new fungibility rules) may be used to cover a PP that the entity itself had issued. In other words, a prosumer, say, who has promised a certain amount of power over a certain time and who must default on that promise due to technical faults can, instead of waiting for the failure to deliver to be discovered and paying penalty fees, buy enough power promises for the period and use them to cover its own promise. This strengthens the secondary market and, should it be healthy, may significantly lower the burden on the supplier of last resort.

Rich power metadata. States may be extended, within reason, as far as the developer wishes in R3 Corda. This means that a PP may have other data as its payload if this is so desired. One of the things it could be engineered to have is a set of claims about the *nature* of the power being generated. Of particular interest are ecological claims regarding, say, carbon emissions, the precise source of the energy, whether it is produced from renewable sources, and so on. Anyone making a PP could make any claim they wished, of course, so there would need to be a specialized network participant, a 'verification agency' that will be discussed in greater detail shortly. Verification agencies will, through countersigning the claims, certify that energy on the market satisfies the claims made. This, in turn, makes it possible, for instance, to run a factory entirely on renewable power not by building one's own generation capacities, which takes time, suitable natural conditions on the factory site, and a significant outlay of capital. Instead, the prospective green manufacturer may simply buy renewable-derived energy on the open market. A greater demand for energy that meets certain ecological or ethical standards can, then, make it profitable to go into the business of manufacturing it, thus increasing renewables adoption rates.

Specialized participants. The prototype only includes the necessary participants. In production, as the system grows, we expect that there will be a place for more specialized network participants to provide specific services within the network and make certain types of trading more efficient.

- **Marketplace**. A marketplace is a specific participant who takes care of specialized trading methods. The default method implemented in the prototype is the simple auction, but a marketplace may implement other approaches to trading such as, e.g., transaction pools with automated periodic settlement and matching. The marketplace can implement and provide new trading techniques and the necessary infrastructure to use them and charge a transaction fee on top of what the network itself already collects as payments for services rendered.
- **Verification Agency**. As things stand, only one agency can reliably report whether promised energy is being delivered to the grid or not. However, other claims relating to reality may be amenable to decentralized review and authentication. This is done by having a verification guarantee claims about power through countersigning, claiming by doing so that they have done some sort of audit or examination in the real world and verified that what is claimed about the energy being promised is indeed true. A verification agency trades on its reputation: the better it does its job the more people trust it, and the more people trust it the more its word is worth.

- **Speculator.** Given the way PPs can be traded between people such that neither is the producer of the energy, nor the person intending to use it and given the fungibility augmentation already discussed it is natural that the system will draw speculators. Speculators are a class of participants in the system whose interest lies in buying and selling PPs in such a way that they make a profit on the trade. The only modification that needs to be made to accommodate this group of participants is to establish both rules and enforcement systems that restrict speculators to a limited volume of trading. This is important because it is very important that investors not drive out people buying energy to run homes and businesses.
- **Energy storage provider.** As we mentioned before, someone providing energy storage is, for the system, essentially nothing more than a prosumer. Like a prosumer, it both buys power on the open market and produces and sells power, i.e., it creates power promises. The difference is that the power bought is not used but stored, and the power produced is released from storage. Therefore, as the importance of energy storage within the system increases, implementing support for energy storage providers and factoring in their storage capacity and efficiency in trading automation and configuration should not present any significant challenges.

Trading automation. While the prototype system works as described, the complete production version of the solution is expected to include further automation on the marketplace based on the application of artificial intelligence and machine learning. For example, some prosumers' automated trading agent can reference current production, the area under solar panels available, and the weather forecast for tomorrow and offer a certified-green "power promise" for a specific timeslot tomorrow. It does this on a local marketplace. The offer must be co-signed by the prosumer's aggregator node, the marketplace node, and a verifier node. A verifier node is specifically there to make sure that the rules of the system are followed and can be maintained by any entity that is trusted enough. Certainly, the power distribution companies will, at a minimum, run their verifier nodes and so can the marketplaces. The verifier node must certify that the prosumer is in good standing and the offer fits within the limits the prosumer must operate under.

A speculator may notice this and decide to buy the obligation on the market and send the request to the marketplace where it is signed by the marketplace, a verifier, which checks for speculator limits and good standing status, and finally the prosumer's aggregator node. Once it is signed, this triggers an external integration system to perform a money transfer using either existing payment infrastructure or through cryptocurrency. The fact that the speculator now owns the "power promise" is written into the ledger. The prosumer then owes the speculator a given amount of power at a given time.

If they fail to deliver as much power as agreed upon, the grid authority will deliver it and charge a penalty to the prosumer. The speculator in turn can sell the power debt to another speculator or to a consumer. If the speculator cannot sell the power debt by its settlement slot it must either accept the power itself, spending it, or the grid will do so, and the speculator will pay the penalty cost. This way while you can trade freely with future obligations once final settlement time comes the rules of the power grid must be followed with inputs and outputs being equal.

4.4. Future Research

The research still needs to expand the scope of the project and assure that its implementation is reduced to a solely engineering problem, encompassing two vital lines of enquiry. The first is the question of the 'bridge to the real world.' For DREAM to manage real-world power transfer it must have a fairly good idea of what is happening in reality. This means that the system must have a mechanism for determining true facts about the real world and the domain it operates in. Ideally, the method it gets these facts through should be distributed, decentralized, robust, and fully automated. There is currently no industry standard on how to create such a bridge between a DLT solution and the real world. The research required is to determine if the present system (which relies in its entirety on the grid authority and its power consumption/production sensors) can be expanded to include third-party verifiers, possibly leading to a separate high-performance high-throughput distributed ledger/blockchain used to connect sensors and verifiers into one decentralized system for establishing true facts about the power grid. This research must be, at present, somewhat tentative. The power grid is a critical piece of national infrastructure, no matter the country, and so rules about who gets to participate and how, including measuring it, are strict. Therefore, this research must range in advance of the actual state of the art waiting for the legal framework to catch up.

The second line of enquiry is the question of scaling. The DREAM prototype is limited not conceptually but by the available hardware. Still, the question remains: just how efficient is the system and what is the load it could take and how much in the way of resources would this load demand? To this end, we need to determine the precise load the prototype and its expanded versions impose on a system and how this load scales with various projected usage patterns. This will reveal any limitations to scaling in the technology itself, as well as determine the precise transaction costs—both monetary and in power consumption—of using the system. If these costs equal or exceed the expected gains of using the technology in the first place, the algorithms in question must be optimized and the infrastructure employed enhanced until the use of the system is profitable both economically and in terms of more rational power usage. We propose rigorous stress testing of the prototype in various configurations and measuring its resource consumption to build up a comprehensive model of future resource use.

5. Conclusions

We set out to create a prototype proof of concept decentralized power trading system, and to do it in a way that can be implemented swiftly and can fit into the present environment. Our choice of technology seeks to compliment the present organization, both legal and technological, of power distribution grids, and to be scalable, efficient in its power/computation use, and economical. To this end, we created a prototype web application for power trading that is backed by a distributed ledger implemented in Corda. This prototype allows for the trading of promises of future power delivery in a decentralized and efficient manner. We have accomplished some of our initial goals with this prototype and are on the way to accomplishing more, but the work is just beginning.

The most important next step is the further validation of the technology itself. We plan to subject the prototype to various simulated loads in various network conditions to ascertain whether the system can be scaled and if it can be used without excessive infrastructure or energy cost. We also plan to work on integration with external systems so that when the time for application comes (as dictated by institutional decisions since the system cannot work without institutional backing), the system can be bridged with the real world without undue difficulty.

The prototype itself needs to be expanded to better meet all the challenges of actual use and to be a better testing platform. We will expand it with improved infrastructure scalability through aggregator nodes, improved trading support through full power fungibility, rich power metadata that allows for the tracing of energy origin, and support for more specialized trading roles (such as speculators and marketplaces) to make full trading automation possible, allowing for frictionless turnkey use by end users.

Author Contributions: Conceptualization, D.B.G., V.B.P., and N.H.; methodology, D.B.G. and V.B.P.; software, N.H.; visualization, D.D.; validation, D.D. and A.S.; writing—original draft preparation, D.B.G., V.B.P., and N.H.; writing—review and editing, D.D., V.K., and J.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been partly supported by the Ministry of Education, Science, and Technological Development of the Republic of Serbia through project no. 451-03-68/2022-14/200156 "Innovative scientific and artistic research from the FTS (activity) domain" (2020–2022).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Algorithm A1: Create power promise flow logic pseudo code.

```
Initiating flow
    Get available notary from network utility
    if Initiating node is not Producer, Prosumer, or Power Company then
        throw Unauthorised action exception
    end if
    Create transaction builder
    if Initiating party has sufficient funds then
        Transfer funds to Grid Authority escrow account
    else
        throw Insufficient funds exception
    end if
        Create Power Promise state and incorporate it into the transaction
    Sign transaction
    Send transaction to Grid Authority Responder flow for signing
    Collect signature from Grid Authority
    if Transaction is verified and signed by initiator and Grid Authority then
        Save transaction in vault
    else
        throw Transaction not verified exception
    end if

Responder flow
    Receive transaction from initiator
    if Transaction is valid then
        Sign transaction
        Send signed transaction to initiator
    else
    throw Invalid transaction exception
    end if
    if Transaction is verified and signed by initiator and Grid Authority then
        Save transaction in vault
    end if
```

Algorithm A2: Check delivery flow logic pseudo code.

```
Initiating flow
    Get available Notary from network utility
    Create transaction builder
    if PP is fulfilled then
        Reduce deposited funds by grid maintenance fee
        Return reduced deposited funds to Supplier
        Record that PP is fulfilled in PP state
        Place modified PP in new transaction
        Sign transaction
        Send signed transaction to Supplier
        if Transaction is verified and signed by both participants then
            Save transaction in vault
        else
            throw transaction not verified exception
        end if
    else
        Reduce deposited funds by grid maintenance fee
        Send reduced deposited funds to Power Company for last minute
     delivery
        Record that PP is not fulfilled in PP state
        Place modified PP in new transaction
        Sign transaction
        Send signed transaction to Power Company
        if Transaction is verified and signed by both participants then
            Save transaction in vault
        else
            throw Transaction not verified exception
        end if
    end if

Responder flow
    Receive transaction from Grid Authority
    if Transaction is valid then
        Sign transaction
        Send signed transaction to Grid Authority
    else
        throw Invalid transaction exception
    end if
    if Transaction is verified and signed by PC/Supplier and GA then
        Save transaction in vault
    end if
```

References

1. Gajić, D.B.; Petrović, V.B.; Horvat, N.; Dragan, D.; Stanisavljević, A.; Katić, V. Blockchain-based Smart Decentralized Energy Trading for Grids with Renewable Energy Systems. In Proceedings of the 2021 21st International Symposium on Power Electronics (Ee), Novi Sad, Serbia, 27–30 October 2021; pp. 1–7, invited paper.
2. Refaat, S.S.; Ellabban, O.; Bayhan, S.; Abu-Rub, H.; Blaabjerg, F.; Begovic, M.M. *Smart Grid and Enabling Technologies*; John Wiley & Sons: Hoboken, NJ, USA, 2021; ISBN 978-1-119-42231-0.
3. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 11 December 2021).
4. Andoni, M.; Robu, V.; Flynn, D.; Abram, S.; Geach, D.; Jenkins, D.; McCallum, P.; Peacock, A. Blockchain Technology in the Energy Sector: A Systematic Review of Challenges and Opportunities. *Renew. Sustain. Energy Rev.* **2019**, *100*, 143–174. [CrossRef]
5. Bürer, M.J.; de Lapparent, M.; Pallotta, V.; Capezzali, M.; Carpita, M. Use Cases for Blockchain in the Energy Industry Opportunities of Emerging Business Models and Related Risks. *Comput. Ind. Eng.* **2019**, *137*, 106002. [CrossRef]
6. European Commission. Energy Prices and Costs in Europe. Available online: https://ec.europa.eu/energy/sites/ener/files/documents/epc_report_final.pdf (accessed on 11 December 2021).

7. Markakis, E.K.; Nikoloudakis, Y.; Lapidaki, K.; Fiorentzis, K.; Karapidakis, E. Unification of Edge Energy Grids for Empowering Small Energy Producers. *Sustainability* **2021**, *13*, 8487. [CrossRef]
8. Kodali, R.K.; Yerroju, S.; Yogi, B.Y.K. Blockchain Based Energy Trading. In Proceedings of the TENCON 2018—2018 IEEE Region 10 Conference, Jeju, Korea, 28–31 October 2018; pp. 1778–1783.
9. Wang, S.; Taha, A.F.; Wang, J.; Kvaternik, K.; Hahn, A. Energy Crowdsourcing and Peer-to-Peer Energy Trading in Blockchain-Enabled Smart Grids. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 1612–1623. [CrossRef]
10. Chowdhury, M.J.M.; Ferdous, S.; Biswas, K.; Chowdhury, N.; Kayes, A.S.M.; Alazab, M.; Watters, P. A Comparative Analysis of Distributed Ledger Technology Platforms. *IEEE Access* **2019**, *7*, 167930–167943. [CrossRef]
11. Gorbunova, M.; Masek, P.; Komarov, M.; Ometov, A. Distributed Ledger Technology: State-of-the-Art and Current Challenges. *Comput. Sci. Inf. Syst.* **2022**, *19*, 65–85. [CrossRef]
12. Gajić, D.; Dragan, D. Blockchain for Business: Technology and Applications. In Proceedings of the International Conference on. Applied Internet and Information Technologies—ICAIIT, Zrenjanin, Serbia, 3–4 October 2019; pp. 17–23, ISBN 978-86-7672-327-0. invited paper.
13. Buterin, V. Ethereum Whitepaper. Available online: https://whitepaper.io/document/5/ethereum-whitepaper (accessed on 12 December 2021).
14. Corda Platform R3. Available online: https://www.r3.com/corda-platform/ (accessed on 12 December 2021).
15. Hyperledger–Open Source Blockchain Technologies. Available online: https://www.hyperledger.org/ (accessed on 11 December 2021).
16. Mengelkamp, E.; Gärttner, J.; Rock, K.; Kessler, S.; Orsini, L.; Weinhardt, C. Designing Microgrid Energy Markets: A Case Study: The Brooklyn Microgrid. *Appl. Energy* **2018**, *210*, 870–880. [CrossRef]
17. Powerledger. Available online: https://www.powerledger.io/ (accessed on 11 December 2021).
18. Lamport, L.; Shostak, R.; Pease, M. The Byzantine Generals Problem. In *Concurrency: The Works of Leslie Lamport*; Association for Computing Machinery: New York, NY, USA, 2019; pp. 203–226. ISBN 978-1-4503-7270-1.
19. Li, Z.; Kang, J.; Yu, R.; Ye, D.; Deng, Q.; Zhang, Y. Consortium Blockchain for Secure Energy Trading in Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3690–3700. [CrossRef]
20. Pee, S.J.; Kang, E.S.; Song, J.G.; Jang, J.W. Blockchain Based Smart Energy Trading Platform Using Smart Contract. In Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Okinawa, Japan, 11–13 February 2019; pp. 322–325.
21. Wang, N.; Zhou, X.; Lu, X.; Guan, Z.; Wu, L.; Du, X.; Guizani, M. When Energy Trading Meets Blockchain in Electrical Power System: The State of the Art. *Appl. Sci.* **2019**, *9*, 1561. [CrossRef]
22. Seven, S.; Yao, G.; Soran, A.; Onen, A.; Muyeen, S.M. Peer-to-Peer Energy Trading in Virtual Power Plant Based on Blockchain Smart Contracts. *IEEE Access* **2020**, *8*, 175713–175726. [CrossRef]
23. Gridchain–Blockchain-Based Process Integration for the Smart Grids of the Future. Available online: https://enerchain.ponton.de/index.php/16-gridchain-blockchain-based-process-integration-for-the-smart-grids-of-the-future (accessed on 11 December 2021).
24. SAP and Wipro. Green Energy Tracking and Distribution. Available online: https://www.sap.com/products/business-technology-platform/use-cases.html (accessed on 11 December 2021).
25. General Electric. The Digital Power Plant. Available online: https://www.ge.com/digital/sites/default/files/download_assets/GE-Digital-Power-Plant-Brochure.pdf (accessed on 11 December 2021).
26. Mollah, M.B.; Zhao, J.; Niyato, D.; Lam, K.-Y.; Zhang, X.; Ghias, A.M.Y.M.; Koh, L.H.; Yang, L. Blockchain for Future Smart Grid: A Comprehensive Survey. *IEEE Internet Things J.* **2021**, *8*, 18–43. [CrossRef]
27. Shafie-Khah, M. *Blockchain-Based Smart Grids*; Elsevier: Amsterdam, The Netherlands, 2020; ISBN 978-0-12-817862-1.
28. Hrga, A.; Capuder, T.; Žarko, I.P. Demystifying Distributed Ledger Technologies: Limits, Challenges, and Potentials in the Energy Sector. *IEEE Access* **2020**, *8*, 126149–126163. [CrossRef]
29. Agung, A.A.G.; Handayani, R. Blockchain for Smart Grid. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *34*, 666–675. [CrossRef]
30. Kappos, I.; Kouveliotis-Lysikatos, I.; Hatziargyriou, N. A Blockchain Platform for the Decentralized Operation of Active Distribution Networks. In Proceedings of the 2021 IEEE Madrid PowerTech, Madrid, Spain, 28 June–2 June 2021; pp. 1–6.
31. Energy Web. Available online: https://www.energyweb.org/ (accessed on 11 December 2021).
32. *Alein Payments System Architecture Design*; Final Draft of the Technical Report for the project no. 1910281-0611; Icelandic Research Fund—Rannis: Reykjavík, Iceland, 2020.
33. Shin, A.K. Hyperledger Fabric vs. R3 Corda: A Business Perspective. Available online: https://medium.com/@kaishinaw/hyperledger-fabric-vs-r3-corda-a-business-perspective-cf935824de9 (accessed on 11 December 2021).