A

MAJOR PROJECT REPORT ON

# BIJLI: A RENEWABLE POWER MANAGEMENT SYSTEM USING BLOCKCHAIN TECHNOLOGY

*Submitted in partial fulfilment of the requirement for the award of the degree*

*Of*

**BACHELOR OF TECHNOLOGY**

*In*

**INSTRUMENTATION AND CONTROL ENGINEERING**

*Submitted by*

**NEHA GUPTA (03411503014)**
**RACHNA PATHAK (03611503014)**

*Under the guidance of*

**MR. PORAS KHETARPAL**

**(Assistant Professor, I.C.E. Department)**

**DEPARTMENT OF INSTRUMENTATION AND CONTROL ENGINEERING**

**BHARATI VIDYAPEETH'S COLLEGE OF ENGINEERING, NEW DELHI**

**(AFFILIATED TO GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY, DELHI)**

**NEW DELHI-110063**

**MAY 2018**

# CANDIDATE’S DECLARATION

It is hereby certified that the work which is being presented in this B. Tech Major Project Report entitled **"BIJLI: A Renewable Power Management System Using Blockchain Technology"** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology**, submitted in the **Department of Instrumentation and Control Engineering** of **Bharati Vidyapeeth’s College of Engineering, New Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi)** is an authentic record of our own work carried out during a period from **January 2018 to May 2018** under the guidance of **Mr. Poras Khetarpal, Asst. Professor, Department of Instrumentation and Control Engineering.**

The matter presented in this B. Tech Major Project Report has not been submitted by us for the award of any other degree of this or any other Institute.

DATE:

PLACE:

**Neha Gupta**
**En. No: 03411503014**

**Rachna Pathak**
**En. No: 03611503014**

# CERTIFICATE

This is to certify that the above statement made by the candidates is correct to best of my knowledge and belief.

**Mr. Poras Khetarpal**
**Assistant Professor**

**Mrs. Aarti Kane**
**HOD & Associate Professor**

The B. Tech Major Project Viva-Voce Examination of Neha Gupta (En. No: 03411503014), and Rachna Pathak (En. No: 03611503014), has been held in May 2018.

**Mr. Poras Khetarpal**
**Project Co-ordinator**

**Ms. Ruchi Sharma**
**Project Co-ordinator**

# ABSTRACT

Via this project, we present a peer-to-peer based power management system using the blockchain technology. We propose a platform where consumers and producers of renewable energy can unite and the consumers can purchase renewable energy from the producers. The use of blockchain increases security and robustness of the whole transaction process, thus introducing transparency in the system. This helps in prevention of frauds and allows each member to view past transactions without letting any member manipulate the real records. We believe that innumerable threats can be avoided with the introduction of blockchain technology in power management systems. Consumers can make informed decisions about from where and from whom do they want to purchase their energy resource. Depleting levels of non-renewable fuel resources pose as a major challenge faced by people all over the world. Consequently, substitution of a significant part of present-day conventional power generation techniques by electricity, wind energy, etc. is being planned by nations and is being implemented in progressive stages. Thus, developing a platform where power management and related exchanges could take place effectively and transparently is something we believed to be of high importance.

# ACKNOWLEDGEMENT

This report is an amalgamation of knowledge gathered from a number of people. We express our gratitude for our mentor Mr. Poras Khetarpal (Assistant Professor), Department of Instrumentation and Control Engineering, Bharati Vidyapeeth's College of Engineering, New Delhi for his kind guidance and support without which this work would not have been possible. We also thank him for bearing with our shortcomings and helping us raise our level. Our sincere thanks to Ms. Aarti Kane, the Head, Instrumentation and Control Engineering Department, Bharati Vidyapeeth's College of Engineering, New Delhi for providing necessary research facilities to carry out this work.

# TABLE OF CONTENTS

# LIST OF FIGURES

# CHAPTER 1: INTRODUCTION & CONCEPTS INVOLVED

## 1.1 MOTIVATION

The Earth is 4.6 billion years old. If we scale that down to 46 years, we've been here for 4 hours. Our industrial revolution began 1 minute ago. In that time we've destroyed more than 50% of the world's rain forests. This isn't sustainable. There is a need to make a difference now. We believe sustainability isn't just a trend. It should not be a job left to the next generation. We see it as our own responsibility.

Electricity is one of the cornerstones for a modern society to function. Households, hospitals, air traffic systems, road infrastructures, communication and financial service industries are all dependent on electricity. The renewable energy market is developing fast, due to increasing energy demands and greater awareness of climate changes. This consequently opens new and interesting opportunities. Research by Bloomberg New Energy Finance shows that by 2040 more than 60% of total investment into the energy sector will go into renewables, which means that the total global investment will be $11.4 trillion of which $7.8 trillion will go into renewable energies and only $3.2 trillion into fossil fuel energy.

This is a significant increase of investments into the renewable energy sector, especially into wind and solar power energy; the latter two, according to Bloomberg, amount to more than 65% of total investment into renewables. In its current state, the energy market is facing challenges in the form of centralized conventional power stations that often require high costs of energy transmission over long distances. The existing electricity model with its infrastructure will not be able to cope with the increasing electricity demand that is expected to more than double by 2050. A change of the model is necessary with a shift to decentralized energy production supported by renewable energy. There has already been some movement towards renewable and sustainable distributed energy systems in recent years. Let's look at an example from the USA and the research done by Rocky Mountain Institute. Projections have shown that in 2015 roughly 15% of the total power generated in the USA was from wind and solar power. By 2050, it is expected to be around 71% and if you add other renewable energy sources like hydro, geothermal power and biomass it could go as high as 80%.

Renewable energy obtained mostly from hydro, wind and solar power will definitely help pave the way to a cleaner, more sustainable energy future. It is clear that over the past few years the cost of solar energy systems has dropped significantly, giving easier access to affordable clean energy. With renewable resources, energy production became more decentralized, local and moved closer to consumption points. A distributed energy system generates power on-site, at the point of consumption and therefore significantly decreases the cost, complexity, interdependencies and inefficiencies associated with transmission and distribution.

Along with increased adoption, generation and consumption of renewable energy arises a need to monitor and control its production and distribution. We believe this is a crucial matter that needs to be addressed while integrating renewable energy into power grids and cities on a large scale. Renewable energy resources are spread over vast geographical areas whereas conventional sources of energy are restricted only to certain regions. With a rise in technological advancements and computational powers of both hardware and software, it has become convenient to build dedicated systems that can provide almost fraud-proof management facilities. We propose a specialized software system that can be used as a user-friendly web application, but has a powerful, reliable and robust backend built upon the Hyperledger fabric based blockchain.

Blockchain-based applications within the electricity market are not only new to the energy business but also a young field within academic research. In their research, F. Imbault et al. [1] proposed to explore the implementation of a blockchain technology into an industrial operating system (Predix) to develop new applications for optimizing the future distributed energy systems. They also believe that much research remains to be done both on the theoretical foundations of the blockchain and on the relevant business applications for energy management, and that more experimental settings are needed to fully envision how the direct use of real-time measurement data from "Internet of Things (IOT)" sensors or from other embedded instrumentation can be achieved.

Kenji Tanaka et al. [2] proposed to use Blockchain-based electricity trading system with Digitalgrid router as an underlying platform because they believed that power grids should be bi-directional systems interconnected using distributed energy management software, and that being able to provide a secure and decentralized control to these autonomous, peer-to-peer exchanges is one of the biggest challenges. Digitalgrid router, which consists back-to-back bi-directional digital inverters with software-based control, enables us to realize power exchange.

Some researchers [3, 4] so far have analyzed how blockchain technology can support the energy management of the distribution grid and within residential micro-grids while integrating distributed RES. Furthermore, M. Mihaylov et al. [5] have introduced a digital currency that allows producers and consumers on the smart grid to trade their produced renewable energy. Tai et al. [6] have looked more closely at blockchain-based electricity transactions and congestion management.

While the academic research mentioned analyses how blockchain technology can be used to solve some of the open questions and issues concerning the electricity market, it does not address business cases and potential organizational opportunities. Research covers business models involving the smart grid or explores business models that encourage the flexibility of distributed energy resources (DER). Hence, there is a research gap regarding business models and software systems based on the blockchain within the electricity market.

F. Imbault et al. in their research, also concluded that the blockchain-IoT combination is powerful and can cause significant transformations across several industries, paving the way for new business models and novel, distributed applications. Completely decentralized cryptocurrencies like Bitcoin have captured the public's attention, and have been more successful than any prior attempts at introducing electronic cash. Many researchers believe this rise of electronic currencies to be a technological revolution.

Emerging coins such as Ethereum and Counterparty extend Bitcoin's design by offering a rich programming language for writing smart contracts. Smart contracts can be thought of as user-defined programs that specify rules for transactions and are enforced by a network of peers, assuming the underlying crypto-currency is secure. In comparison with traditional financial contracts, smart contracts carry the promise of low legal and transaction costs, and lower the bar of entry for users. On the same grounds, we introduce the concept of "BijliCoin" (BIC), which is a payment/reward system that is used as a commodity for trading renewable energy in our proposed ecosystem.

## 1.2 CONCEPTS INVOLVED

### 1.2.1 THE BLOCKCHAIN

#### *How It Works*

Blockchain technology is a way to structure, organize and keep data without the requirement of any central controlling/governing authority. One major aspect of a blockchain is a database that hosts shared records. The database stores records in blocks instead of collecting them in a single file. Each new added block is hence “chained” to previous block, in linear, chronological order, using a cryptographic hash.

As a result of this process, the records can then never be revised and any efforts to introduce changes are visible to all participants. This allows blockchains to be used as ledgers, which can be shared and bolstered by any person/organization possessing appropriate permissions. These distributed ledgers can ultimately be spread across multiple web sites, countries or institutions. Thus, blockchain serves as a very powerful tool.

A blockchain is a distributed data structure that is replicated and shared among the members of a network. It was introduced with Bitcoin to solve the double-spending problem [11]. As a result of how the nodes on the Bitcoin network (the so-called miners) append validated, mutually agreed-upon transactions to it, the Bitcoin blockchain houses the authoritative ledger of transactions that establishes who owns what.

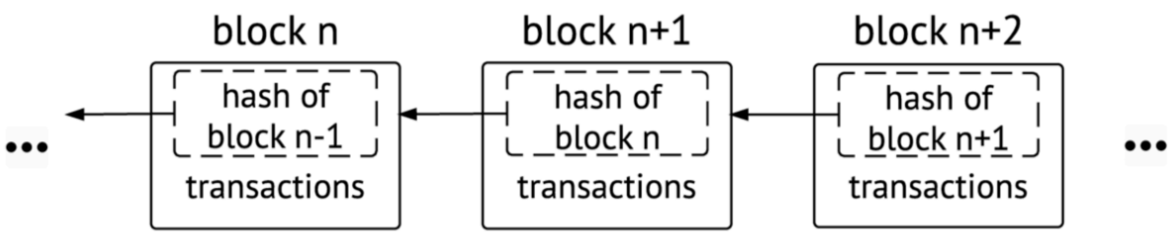



**Figure 1:** Depicting 3 blocks of a blockchain

However a blockchain can stand on its own just fine – no cryptocurrency needed. We can think of the blockchain as a log whose records are batched into time stamped blocks. Its cryptographic hash identifies each block. Each block references the hash of the block that came before it. This establishes a link between the blocks, thus creating a chain of blocks, or blockchain, as shown above. Any node with access to this ordered, back-linked list of blocks can read it and figure out what is the world state of the data that is being exchanged on the network.

We can get a better understanding of how a blockchain works, when we examine how a blockchain network runs. This is a set of nodes (clients) that operate on the same blockchain via the copy each one holds. A node can generally act as an entry point for several different blockchain users into the network, but for simplicity, we assume that each user transacts on the network via their own node. These nodes form a peer-to-peer network where:

- Users interact with the blockchain via a pair of private/public keys. They use their private key to sign their own transactions, and they are addressable on the network via their public key. The use of asymmetric cryptography brings authentication, integrity, and non- repudiation into the network. Every signed transaction is broadcasted by a user's node to its one-hop peers.

- The neighboring peers make sure this incoming trans- action is valid before relaying it any further; invalid transactions are discarded. Eventually this transaction is spread across the entire network.

- The transactions that have been collected and validated by the network using the process above during an agreed-upon time interval are ordered and packaged into a time stamped candidate block. This is a process called mining. The mining node broadcasts this block back to the network. (The choice of the mining node and the contents of the block depend on the consensus mechanism that the network employs.) Whether the hash is generated over the block's contents or its header, as is the case in Bitcoin for instance, is a design choice. Depending on the implementation, the address can be the public key itself or (usually) a hash of it.

- The nodes verify that the suggested block contains valid transactions, and references via hash the correct previous block on their chain. If that is the case, they add the block to their chain, and apply the transactions it contains to update their worldview. If that is not the case, the proposed block is discarded. This marks the end of a round. Note that this is a repeating process.

When we talk about transaction validation, a natural question that might arise is: what constitutes a valid transaction? We need to remember that in a blockchain network what we have essentially is a set of non-trusting writers sharing a database with no trusted middleman. In order to prevent chaos from erupting in this distributed environment, and in order to help the network reach a common global view of the world (i.e. reach consensus), each blockchain network needs to establish certain rules that each database transaction should conform to.

These application-dependent rules are programmed into each blockchain client, which then uses them to decide whether an incoming transaction is valid, and consequently whether it should be relayed to the network or not. In the simplified "shared database" model we present here, let us assume that each row of the database is mapped to a public key (or address) that controls who can edit it. A valid transaction then is one that attempts to modify a row for which the corresponding signature is present.

When each node in the network follows the steps listed above, the shared blockchain it operates on becomes an authenticated and time stamped record of the network's activity. Note how the nodes do not have to trust any other entity, giving rise to the term trustless environment; instead, trust is achieved as an emergent property from the interactions of different participants in the system.

The above is what happens from a high-level view and in the general case.

### *Reaching Consensus in Blockchain Network*

The nodes need to agree on the transactions and the order in which these are listed on the newly mined block. Otherwise, the individual copies of the blockchain will diverge and we will end up with forks; the nodes will have a different view of the world state and the network will no longer be able to maintain a unique authoritative chronology (i.e. blockchain) unless this fork is resolved.

A distributed consensus mechanism is therefore needed in every blockchain network in order to achieve that. The type of consensus mechanism used depends on the type of the blockchain network and the attack vector that the network operator adopts. (The "it depends" answer will unfortunately be a recurring theme throughout this work. It may make things less straightforward to explain, but it also speaks to the versatility of the blockchain to adapt in a number of situations.)

In an ideal scenario, all validating nodes would vote on the order of transactions for the next block, and we would go with what the majority decided. In an open network that anyone can join though, this would be catastrophic because of the Sybil attack: a single entity could join with multiple identities, get multiple votes, and thus influence the network to favor this entity's interests. In other words, a minority could seize control of the network.

Bitcoin works around this problem by making mining computationally expensive; impersonating multiple entities on the network will not help, as the computing resources of any single entity are limited. Specifically, any node can have their assembled block be the next mined block on the network, if they can find the right random number (nonce) in that block's header that will make the SHA-256 hash of the header to have the amount of leading zeroes that the network expects.

Any node that can solve this puzzle, has generated the so-called proof-of-work (PoW) [12] and gets to shape the chain's next block. Since a one-way cryptographic hash functions is involved, any other node can easily verify that the given answer satisfies the requirement (and adopt this block for its blockchain if that's the case), but cannot do the opposite; i.e. guess from the result requirement what the input should be.

It should be noted that a fork may still happen on the network, when two competing nodes mine blocks almost simultaneously. Such forks are usually resolved automatically by the next block; the proof-of-work mechanism dictates that the nodes should adopt the fork that carries the greatest amount of work, and it is unlikely that the two competing forks will generate the next block simultaneously. The nodes first will adopt whichever fork grows longer as the correct one. This allows the network to reach consensus on the proper order of events again.

Other hashing algorithms can be used for PoW besides SHA-256, such as Blake-256 and scrypt. There are also mechanisms that combine several of these algorithms together, such as Myriad. Proof-of-stake (PoS) [12] is an alternative to proof-of-work that requires far fewer CPU computations for mining. In PoS, the chances of a node mining the next block are proportional to that node's balance. PoS schemes have their own strengths and weaknesses, and actual implementations are proving to be quite complex. In private networks however, where the participants are whitelisted, costly consensus mechanisms such as proof-of- work are not needed; the risk of a Sybil attack is not there. This practically removes the need for an economic incentive for mining, and gives us a wider range of consensus protocols to pick from.

Practical Byzantine Fault Tolerance (PBFT) [13] is such an algorithm. It provides a solution to the Byzantine Generals Problem that works in asynchronous environments like the Internet. (Bitcoin, via the mechanism described above, also provides a practical solution to the same problem.) It involves a three-phase protocol and the notion of a “primary” (leader) node that acts as the block miner; the leader can be changed by the rest of the network via a so-called “view-change” voting mechanism, if it crashes or if it exhibits arbitrary behavior (Byzantine faults). PBFT works on the assumption that less than one third of the nodes are faulty (f), which is why say that it requires at least 3f + 1 nodes.

Tangaroa, a Byzantine Fault Tolerant (BFT) variant of the popular Raft algorithm, is used as a consensus mechanism in Juno. Tendermint provides BFT tolerance and is similar to the PBFT algorithm; however it provides a tighter guarantee with regards to the results returned to the client when more than one third of the nodes are faulty, and allows for a dynamically changing set of set of validators, and leaders that can be rotated in a round-robin manner, among other optimizations.

Ripple’s consensus algorithm uses “collectively- trusted sub networks” called “Unique Node Lists” (UNL) to deal with the high latency that usually characterizes BFT-tolerant systems. A node needs to query only its own UNL, instead of the whole network, in order to reach consensus. It can tolerate less than one fifth of its nodes being faulty (5f + 1 resilience). In the mining diversity scheme (used in Multi-Chain), whitelisted miners add blocks to the chain in a round-robin manner, with some degree of leniency to allow for malfunctioning nodes. A network parameter called “mining diversity” is used to calculate the number of blocks that a miner should wait for before attempting to mine again (otherwise its suggested block will be rejected).

A low value of the mining diversity parameter means that fewer miners need to collude in order to take over the network; if the number of colluding miners is equal to or bigger than the number of blocks each miner should wait before attempting to mine again, and then there is a probability that this will happen. Conversely, a higher value of the mining diversity parameter ensures that more permitted miners are included in the rotation, thus making the network take-over by a minority more difficult.

Sieve, a mechanism used in the Hyperledger Fabric project, augments the PBFT algorithm by adding speculative execution and verification phases, inspired by the execute-verify architecture presented in. This allows the network to detect and filter out possible non-deterministic requests, and also achieve consensus on the output state of the suggested transactions (in addition to consensus on their input order). It should be noted that regardless of the consensus mechanism used, the miners in a blockchain network have far less power than the owner of a traditional centralized database since they cannot fake transactions.

## *Transferring Digital Assets on The Blockchain*

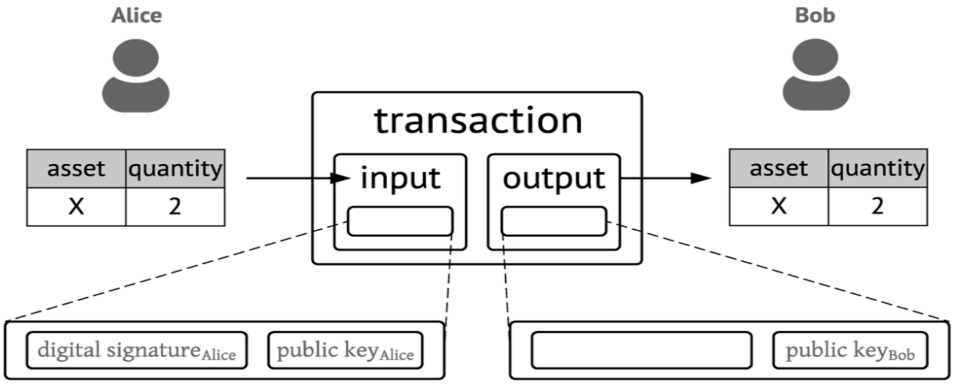



**Figure 2:** Example of a transaction between Alice & Bob

In order to show how an asset transfer works, it is best to consider a simplified example from the banking world. Imagine a bank's (centralized) database that tracks the aggregate balances of each customer. We are basically looking at a table with three columns: "asset type", "owner" and "quantity". For example, a row in that table with "USD", "Alice", "10" identifies Alice as having $10 deposited in that bank. Bob has an account in the same bank with $0 in it. When Alice transfers $2 to Bob's account, the "quantity" of the USD/Alice (asset type/owner) row gets updated to $8, and that of USD/Bob now reads $2. An asset ($2 USD), or rather the digital representation of this asset, was transferred between two entities via a transformation of the appropriate rows in the database.

This transfer of digital tokenized assets can be achieved easily and in a cryptographically verifiable manner using a blockchain network that employs the Bitcoin transactional model. Consider a model of a database that is shared by non-trusting writers in a trustless environment. Each row carries the same fields as in the banking example; with the difference that the "owner" field now holds the public key of the user that is allowed to edit the row. Assume that the database shows that Alice owns 10 units of asset X. (We will get to how this truth was established, i.e. how these assets were generated shortly.)

A row in that database, carries Alice's public key in the "owner" column, and the values "X" and "10" in the "asset type" and "quantity" columns respectively. Assume that Alice knows Bob's public key. How does Alice transfer 2 units of X to Bob? She signs a transaction that modifies her row, decreasing the value of X by 2, and creates a new row, whose "owner" is set to Bob's public key, and whose "asset type" and "value" fields are set to "X" and "2" respectively.

Alice transferred 2 units of X to Bob by creating a new row with that information and assigning it to him; see Figure 2. (In fact, Alice's transaction also deleted her own row, created a new row assigned to one of her public keys, and moved the 8 remaining units of X she holds there. That is done in order to control concurrency and prevent conflicts between concurrent write operations in the system; rows are not modified, instead they are deleted and new rows are created with the updated values.) Bob's new balance of asset "X" can be calculated by aggregating all the rows in the database that correspond to his public keys, and whose "asset type" is set to "X" The same goes for Alice.

Some of the validation checks that we would encode into the nodes of a blockchain network that is set up for such asset transfers would be:

- Does the proposed transaction address an existing row?
- Is it properly signed so as to delete that row (or rows)?
- Has this row been addressed (used) by a previous validated transaction? An asset cannot be spent twice.
- Does it transfer the right amounts to new rows?

For example, if the row the transaction reads "10 units of X", an attempted transfer of "2 units of X" (to Bob) and "9 units of X" (back to Alice) should fail. Same goes for an attempted transfer of, say, "10 units of Y". The sum of inputs should equal the sum of outputs, i.e. a transfer should not increase the total quantity of an asset type. Note that a transaction can address several existing rows instead of just one, i.e. transfer assets scattered over the database, as long as it is properly signed to access them. These existing, not-yet-deleted rows are called unspent transaction outputs (UTXO) in Bitcoin; they were created by earlier transactions in the system. The UTXO that a transaction consumes are called transaction inputs; the UTXO that a transaction creates are called transaction outputs. A transaction then basically deletes a set of rows (UTXO) and creates a set of new rows (UTXO) in the database.

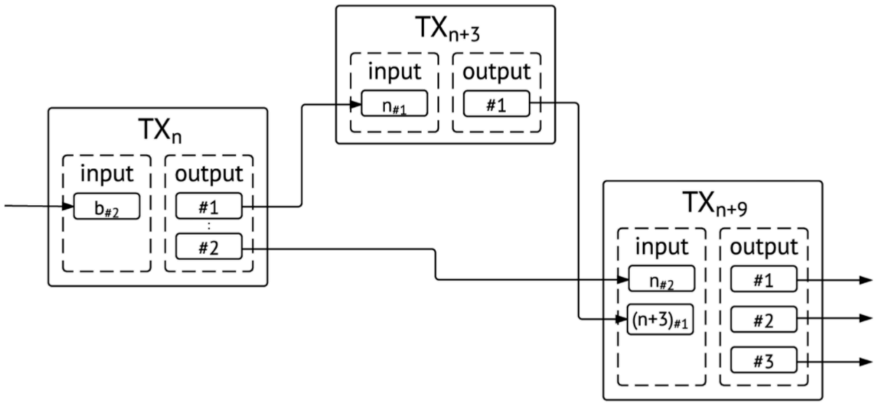



**Figure 3:** Transaction n spends the second UTXO that transaction b (not pictured) created (b#2), and generates two new outputs (n#1, and n#2), spent by transactions n + 3 and n + 9 respectively. A similar process applies to every transaction in the network. Transactions are therefore linked to each other and allow for easy provenance tracking.

One good question, we believe, from the description above is:

How do we generate assets and introduce them in the chain? Before we get to the state of Alice having 10 units of X, these 10 units of X have to come from somewhere. The answer is that it depends on the network and its purpose. Generally, a properly authorized node uses a special type of transaction to introduce an asset (or new units of the asset) into the network.

For example, assume a private blockchain network between Alice, Bob, and Carol. Carol sets this up using Multi-Chain [14], a blockchain platform that assigns permissions (can connect to the network, can transact to the network, can issue on the network) to public keys. Carol configures the network so that her public key can issue assets on the network. She invites Alice and Bob to join; both of them are OK with Carol's ability to issue assets on the chain.

All of them have a pair of private and public keys. Carol submits a signed transaction that generates 10 units of X. The nodes on the network consider this transaction valid, since her public key is permissioned appropriately. Carol then transfers these newly generated units of X to Alice, which brings us to Alice having 10 units of X.

In the case of Bitcoin, new Bitcoin are introduced into the network with every mined block: The mining node includes a so-called coinbase transaction in the block of transactions it broadcasts to the network. This coinbase transaction has no inputs and rewards the mining node with a pre-determined (by the network) amount of Bitcoin.

The key thing to keep in mind is this: if you have a set of users (a) who want to trade digital tokens, and (b) have agreed on how these tokens are generated, then a blockchain network is an ideal tool to use both for exchanging these tokens, and tracking who has what. No middleman is needed to facilitate the exchanges cause every node on the network runs the necessary checks and reaches consensus on the accepted result. Asset tracking comes out-of-the-box since every node has access to the agreed set of cryptographically verifiable transactions on the blockchain.

## *Types of Blockchains*

Based on the participants, blockchains are categorized as public, private or hybrid. This is similar to comparing the public Internet and a company's intranet.

- Public and permission less: Public and permission less blockchains resemble bitcoin, the original blockchain. All transactions in these blockchains are public and no permissions are required to join these distributed entities.
- Private and permissioned: These blockchains are limited to designated members, transactions are private, and permission from an owner or manager entity is required to join this network. These are often used by private consortia to manage industry value chain opportunities.
- Hybrid blockchains: An additional area is the emerging concept of side chain, which allows for different blockchains (public or private) to communicate with each other, enabling transactions between participants across blockchain networks.

## *Blockchain and the energy and commodity transaction lifestyle*

The energy and commodity transaction life cycle, even for simple transactions, involves a multitude of processes within each company and across market participants.

The counterparties have to verify and reconcile transaction data multiple times from execution through settlement of the transaction. Additionally, through the transaction life cycle, a company may need to interact with other counterparties, exchanges, brokers, logistics providers, banks, regulators and price reporters.

As illustrated in the figure below, a web of data interfaces, systems and processes are required to facilitate these interactions. In addition, organizations must maintain their own internal manual processes, systems and controls to maintain an accurate view of transaction data throughout the transaction life cycle.

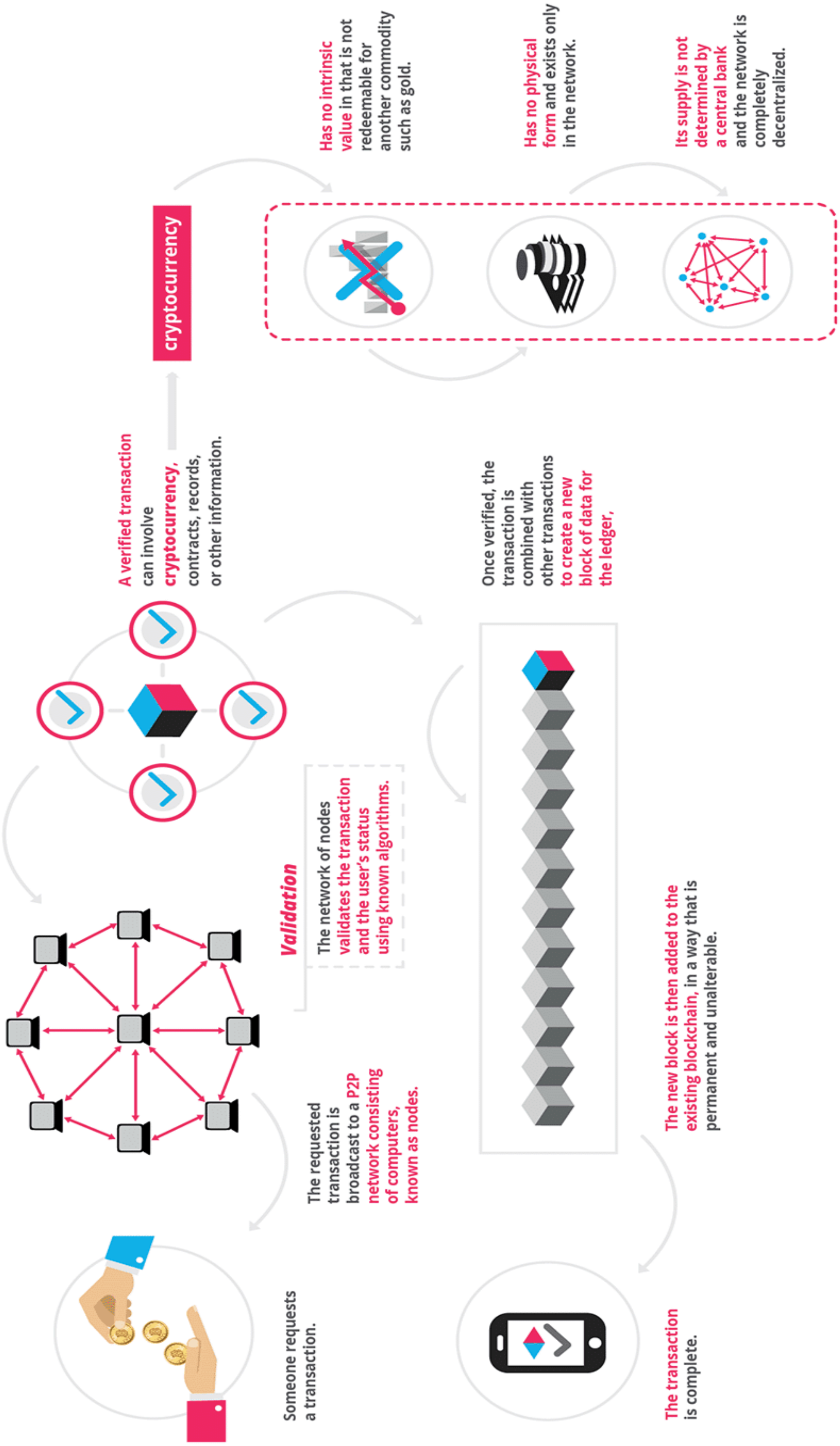



**Figure 4:** Detailed explanation of a blockchain

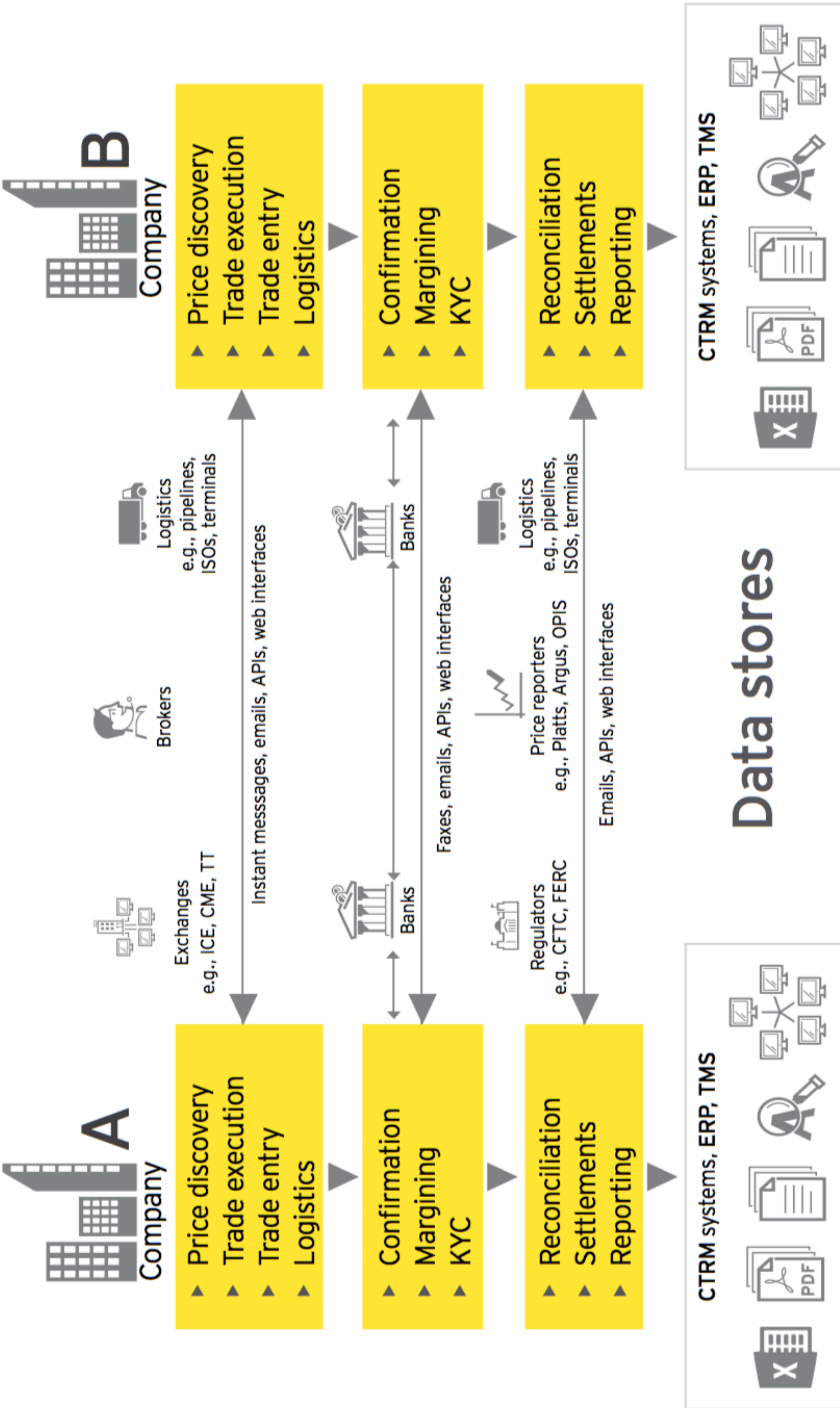



**Figure 5:** Blockchain and the energy and commodity transition life cycle

## 1.2.2 ANGULAR FRAMEWORK

AngularJS was originally developed in 2009 by Miško Hevery at Brat Tech LLC[16] as the software behind an online JSON storage service, that would have been priced by the megabyte, for easy-to-make applications for the enterprise. This venture was located at the web domain "GetAngular.com", and had a few subscribers, before the two decided to abandon the business idea and release Angular as an open-source library.

The 1.6 release added many of the concepts of Angular to AngularJS, including the concept of a component-based application architecture. This release among others removed the Sandbox, which many developers believed provided additional security, despite numerous vulnerabilities that had been discovered that bypassed the sandbox. The current (as of February 2018) stable release of AngularJS is 1.6.9.

In January 2018, a schedule was announced for phasing-out AngularJS: after releasing 1.7.0, the active development on AngularJS will continue till June 30, 2018. Afterwards, 1.7 will be supported till June 30, 2021 as long-term support.

Angular is a front-end JavaScript system that allows its users to develop single page web applications (SPAs). It comes bundled as a solitary JavaScript file, which user can incorporate on their site page. It's absolutely front end, and says nothing in regards to servers. The server only needs to be able to ship out JSON data to be able to communicate to Angular. Angular is a data- binding framework that uses an MVC (Model View Controller) pattern. It allows users to store values in a Model, e.g. an item in a shopping cart. It then automatically keeps that value synchronized with the View, which in Angular environment is an HTML page. When the model updates, the view also updates. The model can be pulled from anywhere on the Internet, since it is just a JSON object.

AngularJS is a full-featured JavaScript framework, with the core goal of simplification. It excels at building dynamic, single page web apps (SPAs) and supports the Model View Controller (MVC) programming structure. It powers sites include Google, Virgin America, and HBO’s mobile site for iPad. Other highlights about Angular include:

- Open-source, front-end JavaScript framework developed by Google
- A library of JavaScript code based on standard JS and HTML, with minimal modifications (meaning, it’s less likely to break)
- Handles the heavy lifting of DOM manipulation and AJAX glue that once had to be coded from scratch
- Encourages the developer to use modular building blocks of JavaScript code that can be categorized and are easy to test
- Can be added to any HTML page with a <script> tag

AngularJS tackles the problem of building dynamic web apps, allowing the developer to extend the functionality of HTML by giving them the ability to create new constructs with Angular directives. This effectively abstracts away tricky DOM manipulation, reducing it to simple elements that can be embedded directly into an HTML template. The most famous example of this is two-way data binding, a once code heavy task being relegated to simply wrapping around your expression.

AngularJS is built on the belief that declarative programming should be used to create user interfaces and connect software components, while imperative programming is better suited to defining an application's business logic. The framework adapts and extends traditional HTML to present dynamic content through two-way data-binding that allows for the automatic synchronization of models and views. As a result, AngularJS de-emphasizes explicit DOM manipulation with the goal of improving testability and performance.

AngularJS's design goals include:

- to decouple DOM manipulation from application logic. The difficulty of this is dramatically affected by the way the code is structured.
- to decouple the client side of an application from the server side. This allows development work to progress in parallel, and allows for reuse of both sides.
- to provide structure for the journey of building an application: from designing the UI, through writing the business logic, to testing.

AngularJS implements the MVC pattern to separate presentation, data, and logic components. Using dependency injection, Angular brings traditionally server-side services, such as view-dependent controllers, to client-side web applications. Consequently, much of the burden on the server can be reduced.

### 1.2.3 REST API

REST stands for Representational State Transfer. APIs are sets of subroutine definitions, rules, protocols, guidelines and tools that are used to create application software. APIs can be understood as set of predefined protocols for communication and interaction between software applications or other systems. REST, was introduced by Roy Fielding [15] in his dissertation. It separates user- interface concerts with data storage concerns. Requests from client to server must contain all the necessary information to process the request and cannot take context based help from the server. The cache constraints require that data within any response to request must be labeled as cacheable or non-cacheable. REST is defined by four interface constraints: identification of resources; manipulation of resources through representations; self-descriptive messages; and, hypermedia as the engine of application state.

The layered system style allows architecture to be composed of layers by constraining component behavior such that each component cannot see beyond the immediate layer with which they are interacting. REST allows client functionality to be extended by downloading and executing code in the form of applets or scripts. This simplifies clients by reducing the number of features required to be pre-implemented.

| Data Element | Modern Web Examples |
|---|---|
| resource | the intended conceptual target of a hypertext reference |
| resource identifier | URL, URN |
| representation | HTML document, JPEG image |
| representation metadata | media type, last-modified time |
| resource metadata | source link, alternates, vary |
| control data | if-modified-since, cache-control |

**Figure 6**: REST Data Elements

| Connector | Modern Web Examples |
|---|---|
| client | libwww, libwww-perl |
| server | libwww, Apache API, NSAPI |
| cache | browser cache, Akamai cache network |
| resolver | bind (DNS lookup library) |
| tunnel | SOCKS, SSL after HTTP CONNECT |

**Figure 7:** REST Connectors

| Component | Modern Web Examples |
|---|---|
| origin server | Apache httpd, Microsoft IIS |
| gateway | Squid, CGI, Reverse Proxy |
| proxy | CERN Proxy, Netscape Proxy, Gauntlet |
| user agent | Netscape Navigator, Lynx, MOMspider |

**Figure 8:** REST Components

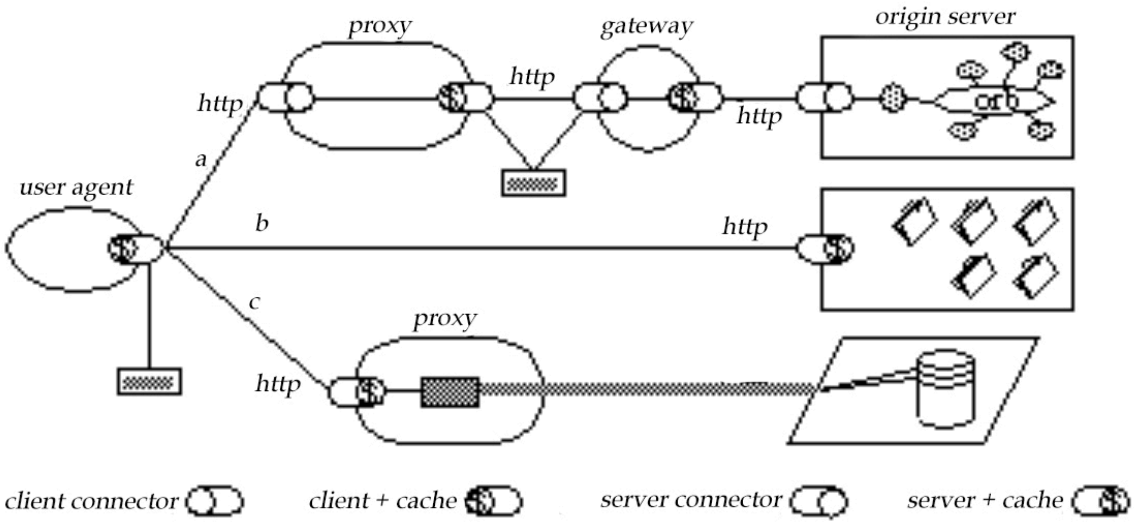



**Figure 9:** Process view of a REST based Architecture

### 1.2.4 GET calls

The GET method is used to retrieve representation of a resource. In the non-error path, i.e. upon a successful and fault-free call, GET returns a representation in XML or JSON format and an HTTP response code of 200 (which indicates OK). In an erroneous case, it usually returns a 404 (NOT FOUND) or 400 (BAD REQUEST) error.

According to design of the HTTP specification, GET requests are used to read data, not change it. Thus they are considered safe to use, i.e. they can be called without any fear of data being modified or corrupted, and also, calling it one time has the same effect as calling it ten times, or calling it zero times. GET is idempotent, which means that making multiple identical requests ends up having the same result as a single request.

### 1.2.5 Renewable Energy

Renewable energy is energy that is collected from renewable resources, which are naturally replenished on a human timescale, such as sunlight, wind, rain, tides, waves, and geothermal heat. Renewable energy often provides energy in four important areas: electricity generation, air and water heating/cooling, transportation, and rural (off-grid) energy services.

Based on REN21's 2016 report [16], renewables contributed 19.2% to humans' global energy consumption and 23.7% to their generation of electricity in 2014 and 2015, respectively. This energy consumption is divided as 8.9% coming from traditional biomass, 4.2% as heat energy (modern biomass, geothermal and solar heat), 3.9% hydro electricity and 2.2% is electricity from wind, solar, geothermal, and biomass. Worldwide investments in renewable technologies amounted to more than US$286 billion in 2015, with countries like China and the United States heavily investing in wind, hydro, solar and biofuels. Globally, there are an estimated 7.7 million jobs associated with the renewable energy industries. As of 2015 worldwide, more than half of all new electricity capacity installed was renewable.

Renewable energy resources exist over wide geographical areas, in contrast to other energy sources, which are concentrated in a limited number of countries. Rapid deployment of renewable energy and energy efficiency is resulting in significant energy security, climate change mitigation, and economic benefits.

#### Types of renewable energy include:

- **Wind Power**

Airflows can be used to run wind turbines. Modern utility-scale wind turbines range from around 600 kW to 5 MW of rated power, although turbines with rated output of 1.5–3 MW have become the most common for commercial use. The largest generator capacity of a single installed onshore wind turbine reached 7.5 MW in 2015. The power available from the wind is a function of the cube of the wind speed, so as wind speed increases, power output increases up to the maximum output for the particular turbine. Areas where winds are stronger and more constant, such as offshore and high altitudes, are preferred locations for wind farms. Typically full load hours of wind turbines vary between 16 and 57 percent annually, but might be higher in particularly favorable offshore sites.

Wind-generated electricity met nearly 4% of global electricity demand in 2015, with nearly 63 GW of new wind power capacity installed. Wind energy was the leading source of new capacity in Europe, the US and Canada, and the second largest in China. In Denmark, wind energy met more than 40% of its electricity demand while Ireland, Portugal and Spain each met nearly 20%.

Globally, the long-term technical potential of wind energy is believed to be five times total current global energy production, or 40 times current electricity demand, assuming all practical barriers needed were overcome. This would require wind turbines to be installed over large areas, particularly in areas of higher wind resources, such as offshore. As offshore wind speeds average ~90% greater than that of land, so offshore resources can contribute substantially more energy than land stationed turbines. In 2014 global wind generation was 706 terawatt-hours or 3% of the worlds total electricity.

## • Solar Energy

Solar energy, radiant light and heat from the sun, is harnessed using a range of ever-evolving technologies such as solar heating, photovoltaics, concentrated solar power (CSP), concentrator photovoltaics (CPV), solar architecture and artificial photosynthesis. Solar technologies are broadly characterized as either passive solar or active solar depending on the way they capture, convert and distribute solar energy. Passive solar techniques include orienting a building to the Sun, selecting materials with favorable thermal mass or light dispersing properties, and designing spaces that naturally circulate air. Active solar technologies encompass solar thermal energy, using solar collectors for heating, and solar power, converting sunlight into electricity either directly using photovoltaic (PV), or indirectly using concentrated solar power (CSP).

A photovoltaic system converts light into electrical direct current (DC) by taking advantage of the photoelectric effect. Solar PV has turned into a multi-billion, fast-growing industry, continues to improve its cost-effectiveness, and has the most potential of any renewable technologies together with CSP. Concentrated solar power (CSP) systems use lenses or mirrors and tracking systems to focus a large area of sunlight into a small beam. Commercial concentrated solar power plants were first developed in the 1980s. CSP-Stirling has by far the highest efficiency among all solar energy technologies.

In 2011, the International Energy Agency said, "the development of affordable, inexhaustible and clean solar energy technologies will have huge longer-term benefits. It will increase countries' energy security through reliance on an indigenous, inexhaustible and mostly import-independent resource, enhance sustainability, reduce pollution, lower the costs of mitigating climate change, and keep fossil fuel prices lower than otherwise. These advantages are global. Hence the additional costs of the incentives for early deployment should be considered learning investments; they must be wisely spent and need to be widely shared". Italy has the largest proportion of solar electricity in the world, in 2015 solar supplied 7.8% of electricity demand in Italy. In 2016, after another year of rapid growth, solar generated 1.3% of global power.

## • Hydropower

In 2015 hydropower generated 16.6% of the world’s total electricity and 70% of all renewable electricity. Since water is about 800 times denser than air, even a slow flowing stream of water, or moderate sea swell, can yield considerable amounts of energy. There are many forms of water energy.

Historically hydroelectric power came from constructing large hydroelectric dams and reservoirs, which are still popular in third world countries. The largest of which is the Three Gorges Dam (2003) in China and the Itaipu Dam(1984) built by Brazil and Paraguay.

Small hydro systems are hydroelectric power installations that typically produce up to 50 MW of power. They are often used on small rivers or as a low impact development on larger rivers. China is the largest producer of hydroelectricity in the world and has more than 45,000 small hydro installations.

Run-of-the-river hydroelectricity plants derive kinetic energy from rivers without the creation of a large reservoir. This style of generation may still produce a large amount of electricity, such as the Chief Joseph Dam on the Columbia river in the United States.

Hydropower is produced in 150 countries, with the Asia-Pacific region generating 32 percent of global hydropower in 2010. For countries having the largest percentage of electricity from renewables, the top 50 are primarily hydroelectric. China is the largest hydroelectricity producer, with 721 terawatt-hours of production in 2010, representing around 17 percent of domestic electricity use. There are now three hydroelectricity stations larger than 10 GW: the Three Gorges Dam in China, Itaipu Dam across the Brazil/Paraguay border, and Guri Dam in Venezuela.

Wave power, which captures the energy of ocean surface waves, and tidal power, converting the energy of tides, are two forms of hydropower with future potential; however, they are not yet widely employed commercially. A demonstration project operated by the Ocean Renewable Power Company on the coast of Maine, and connected to the grid, harnesses tidal power from the Bay of Fundy, location of world's highest tidal flow. Ocean thermal energy conversion, which uses the temperature difference between cooler deep and warmer surface waters, has currently no economic feasibility.

## • Bioenergy

Biomass is biological material derived from living, or recently living organisms. It most often refers to plants or plant-derived materials, which are specifically called lignocellulosic biomass. As an energy source, biomass can either be used directly via combustion to produce heat, or indirectly after converting it to various forms of biofuel. Conversion of biomass to biofuel can be achieved by different methods, which are broadly classified into: thermal, chemical, and biochemical methods. Wood remains the largest biomass energy source today; examples include forest residues – such as dead trees, branches and tree stumps, yard clippings, wood chips and even municipal solid waste. In the second sense, biomass includes plant or animal matter that can be converted into fibers or other industrial chemicals, including biofuels. Industrial biomass can be grown from numerous types of plants, including miscanthus, switchgrass, hemp, corn, poplar, willow, sorghum, sugarcane, bamboo, and a variety of tree species, ranging from eucalyptus to oil palm (palm oil).

Plant energy is produced by crops specifically grown for use as fuel that offer high biomass output per hectare with low input energy. Some examples of these plants are wheat, which typically yield 7.5–8 tonnes of grain per hectare, and straw, which typically yield 3.5–5 tonnes per hectare in the UK. The grain can be used for liquid transportation fuels while the straw can be burned to produce heat or electricity. Plant biomass can also be degraded from cellulose to glucose through a series of chemical treatments, and the resulting sugar can then be used as a first generation biofuel.

Biomass can be converted to other usable forms of energy like methane gas or transportation fuels like ethanol and biodiesel. Rotting garbage and agricultural and human waste, all release methane gas – also called landfill gas or biogas. Crops, such as corn and sugarcane, can be fermented to produce the transportation fuel, ethanol. Biodiesel, another transportation fuel, can be produced from leftover food products like vegetable oils and animal fats. Also, biomass to liquids (BTLs) and cellulosic ethanol are still under research. There is a great deal of research involving algal fuel or algae-derived biomass due to the fact that it's a non-food resource and can be produced at rates 5 to 10 times those of other types of land-based agriculture, such as corn and soy. Once harvested, it can be fermented to produce biofuels such as ethanol, butanol, and methane, as well as biodiesel and hydrogen. The biomass used for electricity generation varies by region. Forest by-products, such as wood residues, are common in the United States. Agricultural waste is common in Mauritius (sugar cane residue) and Southeast Asia (rice husks). Animal husbandry residues, such as poultry litter, are common in the United Kingdom.

Biofuels include a wide range of fuels, which are derived from biomass. The term covers solid, liquid, and gaseous fuels. Liquid biofuels include bioalcohols, such as bioethanol, and oils, such as biodiesel. Gaseous biofuels include biogas, landfill gas and synthetic gas. Bioethanol is an alcohol made by fermenting the sugar components of plant materials and it is made mostly from sugar and starch crops. These include maize, sugarcane and, more recently, sweet sorghum. The latter crop is particularly suitable for growing in dryland conditions, and is being investigated by International Crops Research Institute for the Semi-Arid Tropics for its potential to provide fuel, along with food and animal feed, in arid parts of Asia and Africa.

With advanced technology being developed, cellulosic biomass, such as trees and grasses, are also used as feedstock for ethanol production. Ethanol can be used as a fuel for vehicles in its pure form, but it is usually used as a gasoline additive to increase octane and improve vehicle emissions. Bioethanol is widely used in the United States and in Brazil. The energy costs for producing bio-ethanol are almost equal to the energy yields from bio-ethanol. However, according to the European Environment Agency, biofuels do not address global warming concerns. Biodiesel is made from vegetable oils, animal fats or recycled greases. It can be used as a fuel for vehicles in its pure form, or more commonly as a diesel additive to reduce levels of particulates, carbon monoxide, and hydrocarbons from diesel-powered vehicles. Biodiesel is produced from oils or fats using transesterification and is the most common biofuel in Europe. Biofuels provided 2.7% of the world's transport fuel in 2010.

Biomass, biogas and biofuels are burned to produce heat/power and in doing so harm the environment. Pollutants such as sulphurous oxides ($SO_x$), nitrous oxides ($NO_x$), and particulate matter (PM) are produced from the combustion of biomass; the World Health Organisation estimates that 7 million premature deaths are caused each year by air pollution.[76] Biomass combustion is a major contributor

## 1.2.6 Power Systems

With economic growth and development the need for electricity also increases. An estimation presented during the Event Horizon 2017 in Vienna (Energy Blockchain Conference) predicted that in 30 years the existing levels of energy will only suffice to maintain the existing infrastructure, given the population growth and the fact that more and more products use electricity as their primary source. And when energy storage and the automobile industry reach the break-through point, electricity consumption is expected to double.

The increasing demand for electric vehicles will be a significant factor in increased demand. There is no question about demand: it is huge. Over and above that, there is the gap between retail price and bulk price of electricity to consider as well. On average, only one third of the retail electricity price is on the energy itself, the rest are different charges that increase your bill, such as; distribution charge, customer charge, state tax adjustment charge, consumer education charge and some others. Introducing blockchain into the energy market means reducing costs by diminishing the effect and cost of the middleman. It also means optimizing consumption and production of electricity and thus transferring the added value directly to consumers and producers.

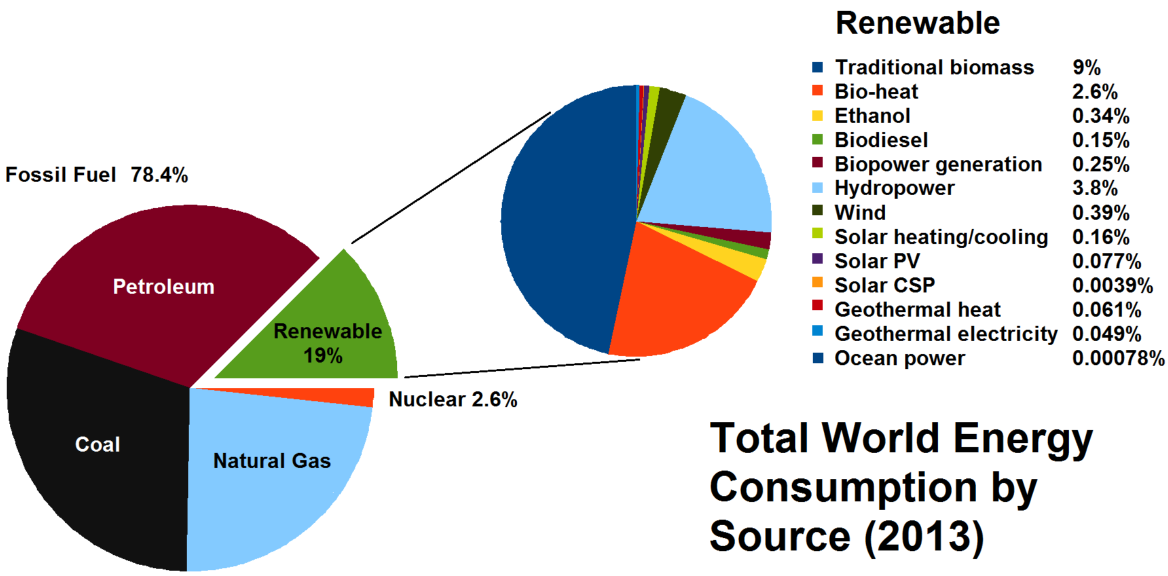



**Figure 10:** World energy consumption, up to 2012. [17]

| Selected renewable energy global indicators | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 |
|---|---|---|---|---|---|---|---|---|---|
| Investment in new renewable capacity (annual) ($10^9$ USD)[88] | 182 | 178 | 237 | 279 | 256 | 232 | 270 | 285 | 241 |
| Renewables power capacity (existing) (GWe) | 1,140 | 1,230 | 1,320 | 1,360 | 1,470 | 1,578 | 1,712 | 1,849 | 2,017 |
| Hydropower capacity (existing) (GWe) | 885 | 915 | 945 | 970 | 990 | 1,018 | 1,055 | 1,064 | 1,096 |
| Wind power capacity (existing) (GWe) | 121 | 159 | 198 | 238 | 283 | 319 | 370 | 433 | 487 |
| Solar PV capacity (grid-connected) (GWe) | 16 | 23 | 40 | 70 | 100 | 138 | 177 | 227 | 303 |
| Solar hot water capacity (existing) (GWth) | 130 | 160 | 185 | 232 | 255 | 373 | 406 | 435 | 456 |
| Ethanol production (annual) ($10^9$ litres) | 67 | 76 | 86 | 86 | 83 | 87 | 94 | 98 | 98.6 |
| Biodiesel production (annual) ($10^9$ litres) | 12 | 17.8 | 18.5 | 21.4 | 22.5 | 26 | 29.7 | 30 | 30.8 |
| Countries with policy targets for renewable energy use | 79 | 89 | 98 | 118 | 138 | 144 | 164 | 173 | 176 |

**Figure 11:** depicting trends in renewable energy sector investments.

### 1.2.7 The BijliCoin

We propose “BijliCoin”, as a reward coin for generating renewable energy.

Blockchain technology is driving innovation and scale-up in the energy, climate, and environmental sectors. BijloCoin, a blockchain-based digital asset and currency, will be designed to accelerate the transition to a clean energy economy. As a blockchain based technology, BijliCoin will be global and decentralized, with similarities to crypto currencies like Bitcoin.

BijliCoin stands out, because unlike these, BijliCoin ‘binds’ the disbursement of digital coins to useful economic and environmental activity - verifiably produced renewable energy.

BijliCoin uses the unique characteristics of blockchain technology to create a phenomenon that is:

- A free, additional reward for renewable energy producers
- The first digital currency to protect natural capital

We plan to reward renewable energy producers with blockchain-based digital tokens at the rate of 1 BijliCoin (BIC) per 1 kWh of solar energy produced.

*But how does BijliCoin work?*

1. Users produce renewable energy

2. Users register renewable energy system to the BijliCoin Foundation/Economy

   a. Provides power purchase agreement

   b. And the amount of renewable kWhs generated

3. Claim is approved by the BijliCoin foundation/economy

4. BijliCoins are disbursed to the user

5. User can then spend / exchange for money / use Bijlicoins

### 1.2.8 Integrating BlockChain and BijliCoin

All transactions would be transparent, and can be viewed by anyone using a blockchain explorer.

Because of their distributed nature, blockchains can reduce transaction costs and time. More importantly, blockchain technology is poised to revolutionize the exchange of value, replacing the need for trusted intermediaries such as banks or currency exchanges.

BijliCoins will be blockchain-based tokens issued into circulation when renewable energy would be verifiably produced. BijliCoins can be transacted among peers. Transactions are collected, verified and summarized in blocks - creating the BijliCoin blockchain.

The BijliCoin blockchain is the high integrity data foundation of our project: a decentralized, incorruptible and auditable record of renewable energy produced.

## 1.2.9 More on the Hyperledger Fabric

Hyperledger Fabric is a platform for distributed ledger solutions, underpinned by a modular architecture delivering high degrees of confidentiality, resiliency, flexibility and scalability. It is designed to support pluggable implementations of different components, and accommodate the complexity and intricacies that exist across the economic ecosystem.

Hyperledger Fabric delivers a uniquely elastic and extensible architecture, distinguishing it from alternative blockchain solutions. Planning for the future of enterprise blockchain requires building on top of a fully vetted, open source architecture; Hyperledger Fabric is a really good starting point. Some of the key design features woven into the Hyperledger Fabric are:

- Assets - Asset definitions enable the exchange of almost anything with monetary value over the network, from whole foods to antique cars to currency futures.
- Chaincode - Chaincode execution is partitioned from transaction ordering, limiting the required levels of trust and verification across node types, and optimizing network scalability and performance.
- Ledger Features - The immutable, shared ledger encodes the entire transaction history for each channel, and includes SQL like query capability for efficient auditing and dispute resolution.
- Privacy through Channels - Channels enable multi-lateral transactions with the high degrees of privacy and confidentiality required by competing businesses and regulated industries that exchange assets on a common network.
- Security & Membership Services - Permissioned membership provides a trusted blockchain network, where participants know that all transactions can be detected and traced by authorized regulators and auditors.
- Consensus - a unique approach to consensus enables the flexibility and scalability needed for the enterprise.

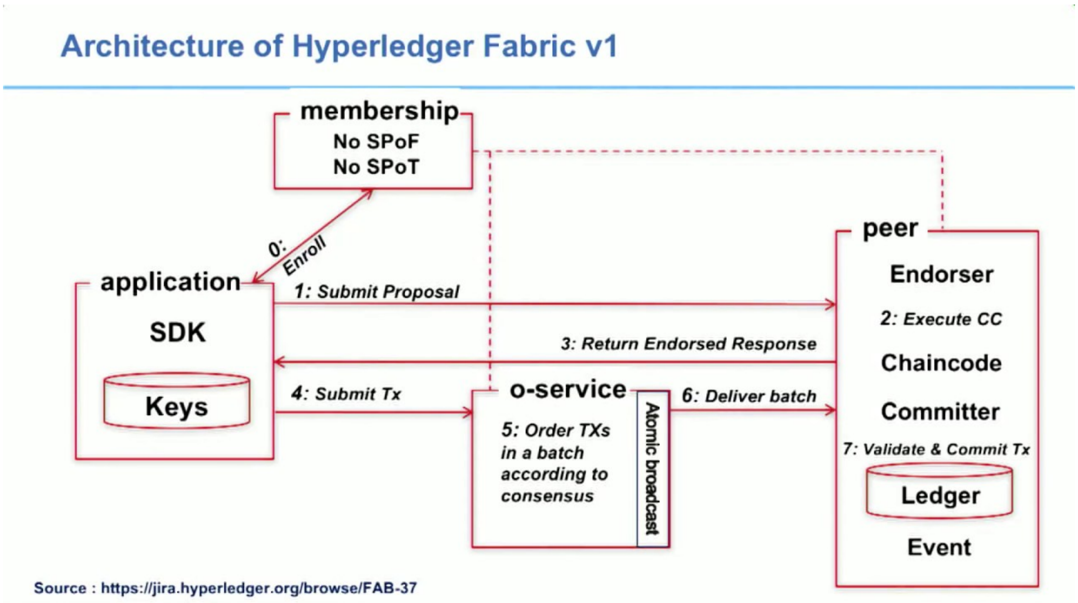



**Figure 12:** Architecture of Hyperledger Fabric v1

### 1.2.10 Smart Contracts

Nick Szabo introduced this concept in 1994 and defined a smart contract as “a computerized transaction protocol that executes the terms of a contract” [18]. Szabo suggested translating contractual clauses (collateral, bonding, etc.) into code, and embedding them into property (hardware, or software) that can self-enforce them, so as to minimize the need for trusted intermediaries between transacting parties, and the occurrence of malicious or accidental exceptions.

Within the blockchain context, smart contracts are scripts stored on the blockchain. (They can be thought of as roughly analogous to stored procedures in relational database management systems [19].) Since they reside on the chain, they have a unique address. We trigger a smart contract by addressing a transaction to it. It then executes independently and automatically in a prescribed manner on every node in the net- work, according to the data that was included in the triggering transaction. (This implies that every node in a smart contract-enabled blockchain is running a virtual machine (VM), and that the blockchain network acts as a distributed VM.)

Smart contracts allow us to have general-purpose computations occur on the chain. Where they excel however, is when they are tasked with managing data-driven interactions [20] between entities on the network. We can unpack this statement with an example. Consider a blockchain network where Alice, Bob, and Carol participate, and where digital assets of type X and Y are being traded. Bob deploys a smart contract on the network that defines: (a) a “deposit” function allowing him to deposit units of X into the contract, (b) a “trade” function that sends back 1 unit of X (from the contract’s own deposits) for every 5 units of Y it receives, and (c) a “withdraw” function that allows Bob to withdraw all the assets that the contract holds.

It should be noted that the “deposit” and “withdraw” functions are written so that only Bob (via his key) can call them, because this is what Bob decided, and also what makes sense for our example; they could have been written so that any user on the network can call them successfully.

Bob sends a transaction to that smart contract’s address, calling its “deposit” function and moving 3 units of X to the contract. This transaction is recorded on the blockchain. Alice, who owns 12 units of Y, then sends a transaction that moves 10 units of Y to the contract’s “trade” function, and gets back 2 units of X. This transaction is also recorded on the blockchain. Bob then sends a signed transaction to the contract’s “withdraw” function. The contract checks the signature to make sure the withdrawal is initiated by the contract’s owner, and transfers all of its deposits (1 unit of X, and 10 units of Y) back to Bob.

Chaincode is the Hyperledger fabric’s interpretation of the smart contract method/algorithm, with additional features. A chaincode is programmatic code deployed on the network, where it is executed and validated by chain validators together during the consensus process. Developers can use chain codes to develop business contracts, asset definitions, and collectively managed decentralized applications.

There are generally two ways to develop business contracts: the first way is to code individual contracts into standalone instances of chaincode; the second way, and probably the more efficient way, is to use chaincode to create decentralized applications that manage the life cycle of one or multiple types of business contracts, and let end users instantiate instances of contracts within these applications. Users can use chaincode (for business rules) and membership service (for digital tokens) to design assets, as well as the logic that manages them.

There are two popular approaches to defining assets in most blockchain solutions: the stateless UTXO model, where account balances are encoded into past transaction records; and the account model, where account balances are kept in state storage space on the ledger. Each approach carries its own benefits and drawbacks. This blockchain fabric does not advocate either one over the other. Instead, one of our first requirements was to ensure that both approaches can be easily implemented with tools available in the fabric.

Chaincode can be written in any programming language and executed in containers inside the fabric context layer. The fabric's first fully supported chaincode language is Golang, and support for JavaScript and Java was released in 2016. Support for additional languages and the development of a fabric-specific templating language are being discussed, and more details will be released in the near future.

# CHAPTER 2: DESCRIPTION & WORKING OF PROJECT

We present Bijli- The distributed energy economy. This is a vision for the smartgrid of the future. Just like any economy, we have residents, who can trade power. We have banks from where power and BijliCoins, i.e. BIC can be exchanged, and there are utilities companies from whom one can trade power, but they're special companies who produce power.

The project is powered by the hyperledger blockchain, which serves as a computationally powerful tool for such applications. The app basically allows user to send power to someone else. Ideally, there will be smart meters as a part of the blockchainand they'll be in sync with the clockchain at regular intervals.

Another feature would be the cost of the coin – cost depends on the demand and supply. If demand is more than supply, the cost would be higher. Else, lower. The standard fixed cost would be initially $10 per coin.

We have build a blockchain powered application. The user interacts with our system via a web app. The application's UI is built on Angular framework, providing the user a friendly and easy-to-use interface.

- The administrator interacts with distributed energy economy UI comprising of the Angular framework.
- The web app then processes the user requests to the network through a REST API.
- Then, it further implements requests to the Blockchain state database on Hyperledger fabric.
- The REST API is used to retrieve the state of the database.
- The Angular framework gets the data through GET calls to the REST API.

The application demonstrates energy sharing through a network between the 'Residents' of an economy. In order to make this possible, real Smart-meters will have to be linked with the blockchain. The users can then interact with the web-app to directly transfer power to someone else in exchange of BIC.

The actual price (in rupees) of BIC changes dynamically based on the demand and supply fundamentals, i.e. when the demand is more than supply, the coin will be worth higher, thus the producers will be paid more, and in case when the supply exceeds the demand of renewable energy, the consequent value of BIC falls.

## 2.1 Setting up the fabric

Firstly, the administrator has to kill and remove all previously running containers, and should remove all previously created Hyperledger Fabric chaincode images. This can be achieved by running the following command:

> *docker kill $(docker ps -q) docker rm $(docker ps -aq) docker rmi $(docker images dev-* -q)*

Then, the Hyperledger Fabric version needs to be set to version 1.0, using the following command:

> *export FABRIC_VERSION=hlfv1*

The fabric can then be started and peer admin card can be generated, using:

> *cd fabric-tools/ ./downloadFabric.sh ./startFabric.sh ./createPeerAdminCard.sh*

## 2.2 Generating the BNA – Business Network Archive & Deploying to the fabric

Then, the Business Network Archive (BNA) file can be generated from the root directory by using:

> *cd ../ npm install*

The composer archive create command in package.json would have now created a file called "decentralized-energy-network.bna" in the dist folder. Now, the network needs to be deployed on the Hyperledger Fabric.

This requires the Hyperledger Composer chaincode to be installed on peer, then the business network archive (.bna) to be sent to the peer, and a new participant, identity, and associated card to be created to act as the network administrator. Following that, the network administrator business network card must be imported for use, and the network then be pinged to check it is responding.

The composer runtime can be installed using:

> *cd dist/ composer runtime install --card PeerAdmin@hlfv1 --businessNetworkName decentralized- energy-network*

Then, the business network can be deployed using the following comand:

> *composer network start --card PeerAdmin@hlfv1 --networkAdmin admin --networkAdminEnrollSecret adminpw --archiveFile decentralized-energy-network.bna --file networkadmin.card*

Then, the network administrator identity can be imported as a usable business network card, using the following command:

> *composer card import --file networkadmin.card*

After checking for the successful deployment of the business network, check for the ping using:

*composer network ping --card admin@decentralized-energy-network*

## 2.3 Executing the application, Creating Participants and Executing Transactions

A dependency will need to be installed, and then the app can be started.

*cd ../angular-app/ npm install npm start*

The application would now be running at: *http://localhost:4200*

The REST server to communicate with network would be available at: *http://localhost:3000/explorer/*

In the application, participants, residents, banks and utility companies can be created and data can be filled in. Execute transactions manually between Residents, Resident and Bank, and Resident and Utility Company. After executing transactions, participants account values would be updated. At the end of session, the fabric can be stopped using:

*cd ~/fabric-tools ./stopFabric.sh ./teardownFabric.sh*

Finally, the costs of the BIC would fluctuate based on demand and supply of the energy. The users can choose from whom do they want to purchase the energy, thus providing them a fair, and equal say in deciding the distribution authority for himself or herself, hence eradicating the unnecessary price hikes faced in case of present day monopoly prone ecosystems.

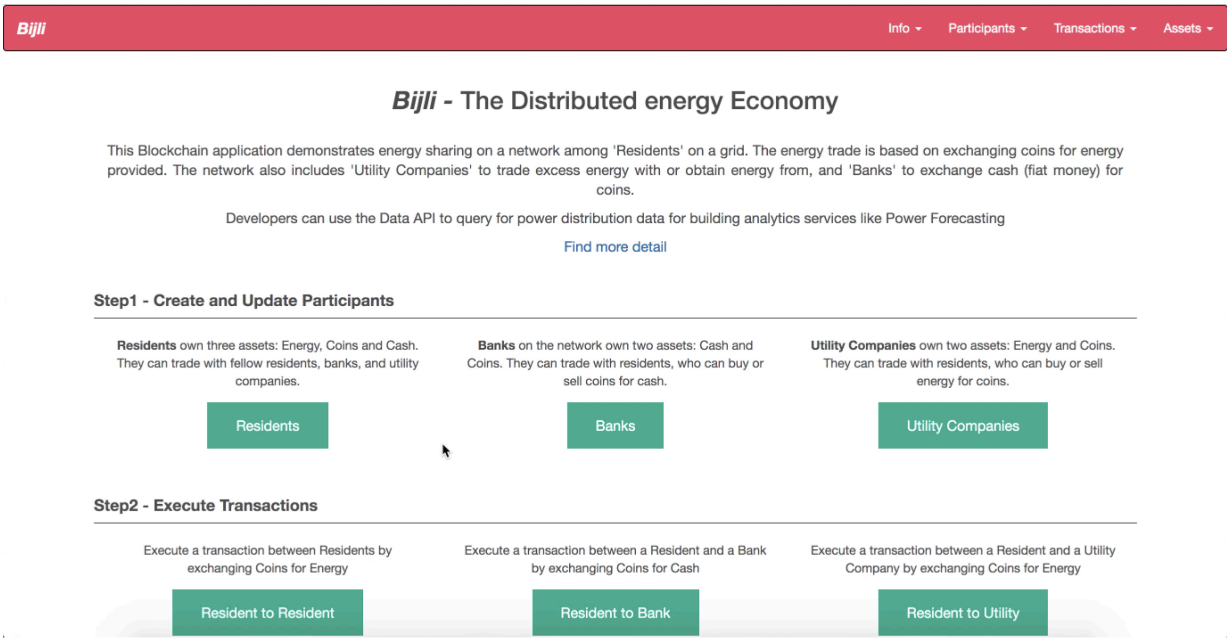
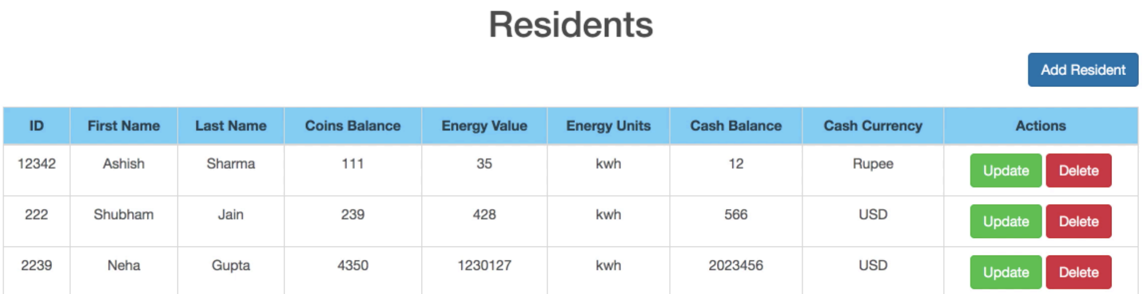



**Figure 13:** Showing the main Graphical User Interface of the Web App Includes the main header, having four different drop down menus. The main page contains the description of the application, and then allows the user to explore through 6 different options based on who they are, and what they want to do.




| ID | First Name | Last Name | Coins Balance | Energy Value | Energy Units | Cash Balance | Cash Currency | Actions |
|---|---|---|---|---|---|---|---|---|
| 12342 | Ashish | Sharma | 111 | 35 | kwh | 12 | Rupee | Update Delete |
| 222 | Shubham | Jain | 239 | 428 | kwh | 566 | USD | Update Delete |
| 2239 | Neha | Gupta | 4350 | 1230127 | kwh | 2023456 | USD | Update Delete |

**Figure 14:** Showing sample records of the residents of the network. New residents can be added, and the residents already added in the network can be deleted or their records can be updated based on whatever the need be

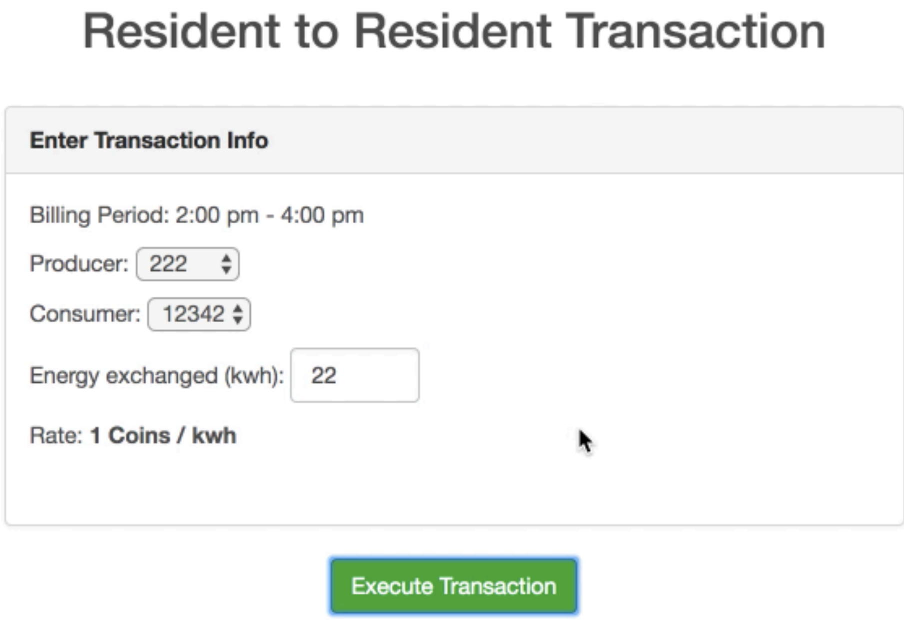



**Figure 15:** Depicting the main resident to resident transaction, including the Billing time period, the producer's unique ID, consumer's unique ID, the amount of energy to be exchanged and the Rate of the BijliCoin during the time of the transaction.

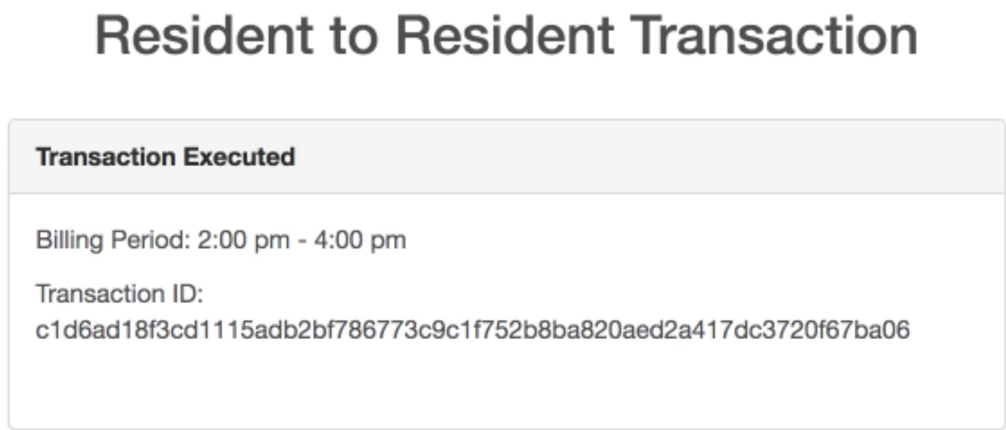



**Figure 16:** Snapshot depicting the transaction executed, it includes the billing time period, the unqiue Transaction ID that can later be used to track/identify as a proof of the transactions executed.

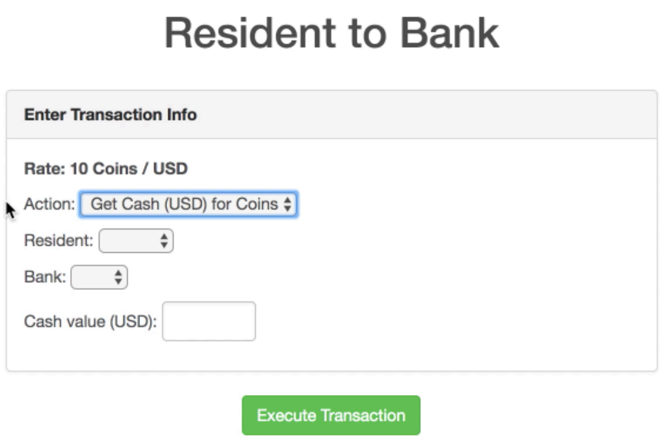



**Figure 17:** Snapshot showing the feature of the app that allows the user to exchange their cash in return of BijliCoins from the Banks

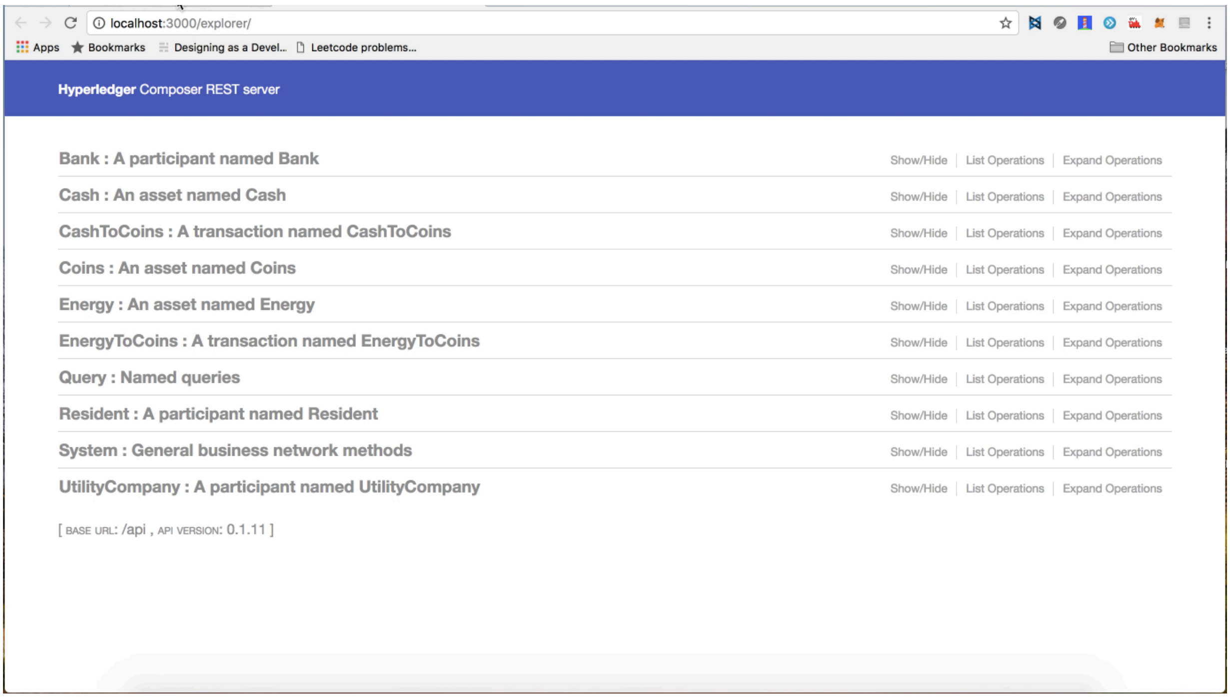



**Figure 18:** Hyperledger Composer REST Server: depicting the functionalities offer by our app that allows it to be integrated into other major applications as an extended service, mainly via APIs.

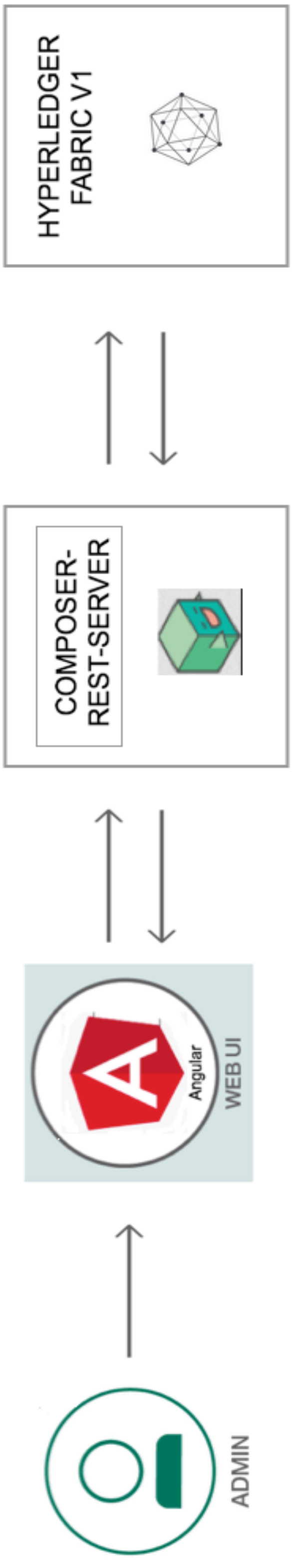



**Figure 19:** Overall Architecture of our Application - BIJLI

# CHAPTER 3: CONCLUSION OF THIS PROJECT

We believe that blockchain is a highly promising innovation for trusted estimation and monitoring of energy related resources, which are required to be more decentralized as self generation and small scale business scenario is spreading widely, both amongst people in general, governmental and and private divisions. Our application, BIJLI can be incorporated into electricity grids at a large scale in urban cities, where users have access to latest technologies. We believe that the need for such a system to exist will become inevitable in the coming times due to a rise in the wide-scale generation and adoption of renewable energy production. There will be a need to govern the consumption, production, management and storage of such energy; and it would be best done without monopoly of single parties at power. The users would ultimately benefit more if they would directly be involved in deciding the sources of their power needs from a variety of producers. BIJLI application demonstrates a basic idea of a decentralized energy network using Blockchain and can be ameliorated in certain ways, some of which include the addition of specific permissions and participant access; setting up real time transactions among participants; integrating with IoT to read from power meter and distribute energy accordingly according to needs.

Blockchains give us resilient, truly distributed peer-to-peer systems and the ability to inter- act with peers in a trustless, auditable manner. Smart con- tracts allow us to automate complex multi-step processes. The devices in the IoT ecosystem are the points of contact with the physical world. When all of them are combined we get to automate time-consuming workflows in new and unique ways, achieving cryptographic verifiability, as well as significant cost and time savings in the process. We also believe that the continued integration of blockchains in the IoT domain will cause significant transformations across several industries, bringing about new business models and having us reconsider how existing systems and processes are implemented.

# CHAPTER 4: FUTURE SCOPE

The real power of a home battery and solar panel setup will come from combining it with the blockchain, a distributed ledger system capable of recording, tracking and verifying transactions across a peer-to-peer network. Instead of only having to option to sell excess energy back to the grid through a local utility provider, this system would allow anyone to sell it to the highest bidder – e.g. their neighbour, local school, shop etc. – via a local smart grid. It's a system that will benefit the planet, our wallets and our outdated energy grids, and with the decentralized energy revolution.

# CHAPTER 5: REFERENCES


[1] F. Imbault, M. Swiatek, R. de Beaufort, R. Plana, “The green blockchain Managing decentralized energy production and consumption”, IEEE International Conference on Environment and Electrical Engineering and IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe), 2017

[2] Kenji Tanaka, Kosuke Nagakubo and Rikiya Abe, “Blockchain-based electricity trading with Digitalgrid router”, IEEE International Conference on Consumer Electronics – Taiwan (ICCE-TW), 2017

[3] Pietro Danzi, Marko Angjelichinoski, Čedomir Stefanović and Petar Popovski, “Distributed Proportional-Fairness Control in MicroGrids via Blockchain Smart Contracts”, arXiv:1705.01453, 2017.

[4] José Horta, Daniel Kofman and David Menga, “Novel paradigms for advanced distribution grid energy management”, arXiv:1712.05841, 2017.

[5] M. Mihaylov, I. Razo-Zapata, R. Rădulescu, S. Jurado, N. Avellana and A. Nowé, “Smart Grid Demonstration Platform for Renewable Energy Exchange”, Lecture Notes in Computer Science: Advances in Practical Applications of Scalable Multi-agent Systems, vol. 9662, pp 277-280, DOI: 10.1007/978-3-319-39324-7_30, 2016

[6] Tai Xue, Sun Hongbin and Guo Qinglai, “Electricity Transactions and Congestion Management Based on Blockchain in Energy Internet”.

[7] Roy Thomas Fielding, "Architectural Styles and the Design of Network-based Software Architectures", University of California, Irvine, 2000.

[8] M. Parol and L. Rokicki, “Voltage stability in low voltage microgrids in aspects of active and reactive power demand”, Archives of Electrical Engineering, vol. 65, 2016.

[9] Bloomberg New Energy Finance: https://www.bnef.com/dataview/new-energy-outlook- 2016/index.html

[10] J. Kelly and A. Williams. (2016). Forty Big Banks Test Blockchain-Based Bond Trading System. [Online]. Available: http://www.nytimes.com/reuters/2016/03/02/ business/02reuters-banking-blockchain-bonds.html

[11] Double-Spending, Bitcoin Wiki, accessed on Mar. 15, 2018, Available: https://en.bitcoin.it/wiki/Double-spending

[12] S. Nakamoto. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[13] M. Castro and B. Liskov, “Practical Byzantine fault tolerance,” in Proc. OSDI, vol. 99. 1999, pp. 173–186.

[14] MultiChain—Open Source Private Blockchain Platform, accessed on Mar. 17, 2018. [Online]. Available: http://www.multichain.com/

[15] Roy Thomas Fielding, "Architectural Styles and the Design of Network-based Software Architectures", University of California, Irvine, 2000.

[16] Bloomberg New Energy Finance, accessed on Apr. 17, 2018, Available: https://www.bnef.com/dataview/new-energy-outlook-2016/index.html

[17] Total world energy consumption by source 2013, from REN21 Renewables 2014 Global Status Report, accessed on Mar. 18, 2018. Available: http://www.ren21.net/status-of-renewables/global-status-report/

[18] N. Szabo. (1994). Smart Contracts. [Online]. Available: http://szabo.best.vwh.net/smart.contracts.html

[19] MySQL Reference Manual—Using Stored Routines (Procedures and Functions), accessed on Apr. 17, 2018. [Online]. Available: http://dev.mysql.com/doc/refman/5.7/en/stored-routines.html

[20] Eris Industries Documentation—Smart Contracts, accessed on Mar. 15, 2016. [Online]. Available: https://docs.erisindustries.com/ explainers/smart_contracts/