

Energy Online

J. Shah, M. Jordi, M. Stemmle, S. Oeschger, S. Zemljic

March 4, 2018

1 Introduction - Michael J. - around 600 words

1.1 Context

The BIOTS program exists to push education for innovative new technologies into universities to help Switzerland to become one of the leading hubs in Blockchain and IoT space.¹

Since 2016, every year the BIOTS program offers three major verticals for the students to build their ideas and use-cases. In 2018, the focus was on sustainability issues, which are currently on the agenda of the UN or hot topics in industry. Challenges from the sponsors Elektrizitätswerk der Stadt Zürich (EWZ), UBS Group AG (UBS), the European founded project FuturICT 2.0 (<https://fudurict2.eu>) across the following verticals were presented and solved in the program:



Figure 1: Major verticals of BIOTS 2018 , Source <http://biots.org/the-biotsphere/>

¹Source: <http://biots.org/about/>

This report was created as a result of the challenge 3 “virtual energy storage” from EWZ.

1.2 Background

In a cantonal vote 2017 it has been decided, that EWZ needs to sell all its holding in the nuclear sites of Gösgen and Leibstadt until 2034. In addition, EWZ will not be allowed to obtain nuclear energy anymore as well.²

In 2016 32.8% of the electricity in Switzerland is provided by nuclear power plants.³ To be able to cover those losses, massive investments into renewable energy are needed. Large power plants alone will not be enough to cover the demands, thus small-scale energy production, like private PV's will play a huge role.

Currently, the entry barriers for the installation of PV are high, battery storage for surplus energy is expensive and overall incentives are missing. For example, energy surplus from private power plants (mainly PV) can only be sold to local energy provider for a very low price of 8.50 Rp./kWh (high tariff) or 4.45 Rp./kWh (low tariff).⁴ To incentivize the installation of PV, promotional money is lacking.⁵

To reach the goal of 0% nuclear energy it is important to make private power plants more lucrative. Therefore, the challenge 3 of EWZ deals with the topic “Virtual Energy Storage”. Virtual Energy Storage allows private household to store their surplus energy virtually and withdraw it later when needed.

This concept has been already adapted Companies like eOn or SENEK already offer services like this to their customers:

eOn (<https://www.eon-solar.de/eon-solarcloud>)

- Service Name: E.ON SolarCloud
- Value Proposition: Genießen Sie jetzt Ihre Sonnenenergie 365 Tage, und Nächte

²Source: <https://www.nzz.ch/zuerich/aktuell/abstimmungssonntag-zuerich-atomausstieg-zuerich-ld.130738>

³Source: Schweizerische Elektrizitätsstatistik 2016, p. 5: http://www.bfe.admin.ch/php/modules/publikationen/stream.php?extlang=de&name=de_306571764.pdf

⁴Source: <https://www.ewz.ch/content/dam/ewz/services/dokumentencenter/energie-produzieren/dokumente/verguetung-stromruecklieferung-zh-2016-18.pdf>

⁵Source: <https://www.srf.ch/news/schweiz/abstellgleis-solarenergie-wer-foerdergelder-will-muss-jahre-warten>

- Value Chain: E.ON -> End customer
- Revenue Model: Monthly Fee
- Customer Segment: Private Households with PV installations

Senec-ies (<https://www.senec-ies.com/tarife-services/senec-cloud/>)

- Service Name: SENECloud 2.0
- Value Proposition: Strom im Sommer einfrieren und im Winter wieder auftauen.
- Value Chain: SENECloud -> End customer
- Revenue Model: Monthly Fee
- Customer Segment: Private Households with PV installations

Due to the companies and customers are trusted parties in those two services offerings, the necessity of the blockchain technology is not given. According to Karl Wüst and Arthur Gervais, “Blockchain is being praised as a technological innovation which allows to revolutionize how society trades and interacts. This reputation is in particular attributable to its properties of allowing mutually mistrusting entities to exchange financial value and interact without relying on a trusted third party. A blockchain moreover provides an integrity protected data storage and allows to provide process transparency.”⁶

1.3 Problem Statement

The problem statement given by the EWZ Challenge 3 “Virtual Energy Storage” is: “How might the blockchain technology help to develop a reliable and highly efficient virtual energy solution that gives anyone the possibility to produce, store and use his/her own energy anywhere and anytime?”⁷

2 Literature Review - Jig - around 1100 words

- Business Model <https://www.eon-solar.de/eon-solarcloud>
- Business Model <https://www.senec-ies.com/tarife-services/senec-cloud/>

⁶“Do you need a Blockchain?”, Karl Wüst, Arthur Gervais, <https://eprint.iacr.org/2017/375.pdf>

⁷Source: “BIOTS_ewz_challenges.pdf”

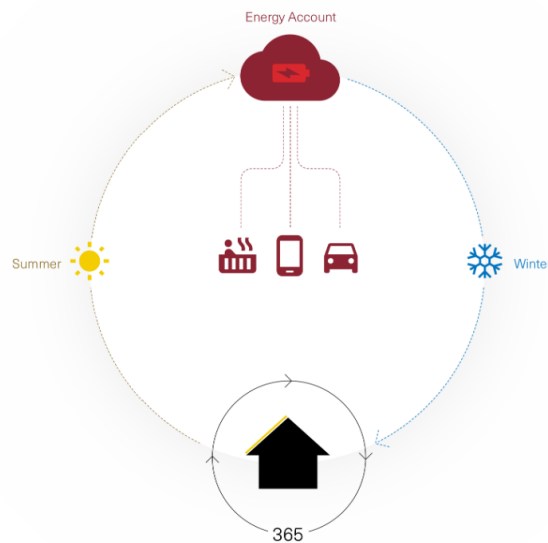


Figure 2: EWZ Challenge 3 - "Virtual Energy Storage", Source: BIOTS_ewz_challenges.pdf

- Certificats: <http://www.bfe.admin.ch/themen/00612/00614/index.html?lang=en>
- Blockchain and Energy <https://hbr.org/2017/03/how-utilities-are-using-blockchain-to-modernize-the-grid>
- Blockchain and Energy <https://www.etla.fi/wp-content/uploads/ETLA-Working-Papers-43.pdf>
- Blockchain and Energy <http://ieeexplore.ieee.org/abstract/document/7589035/>
-

3 Conceptual Model - Simon around 1100 words - Michael J. around 500 words

Our aim is to create a Decentralized Autonomous Organization (DAO). The DAO's behaviour is completely controlled by smart contracts and it fulfils functions similar to the ones of an energy provider today:

- Energy Management: The DAO makes sure that there is sufficient energy available

for all the participants of the systems and enables all parties to trade energy freely by using the energy token.

- Trading at the energy market: The DAO sells surplus energy and buys energy if there is a shortage in the network on the traditional energy market.
- Profit distribution: The DAO uses profits realized by trading to subsidize others to install renewable energy production possibilities and participate in the system. This enlarges the current system and creates additional surplus.

Within the DAO, energy is traded freely amongst all participants by using energy tokens. Energy tokens represent the right to obtain energy from a participant of the system.

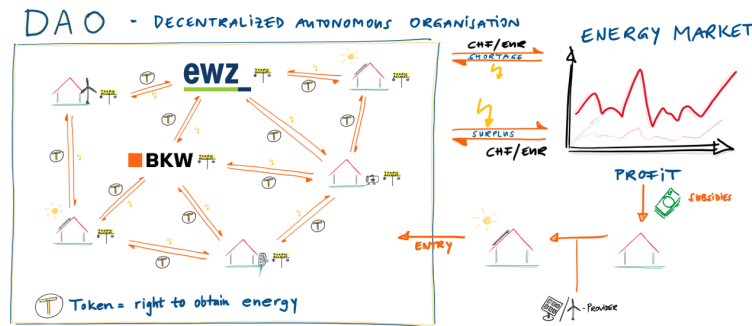


Figure 3: Conceptual Design, Source: this report

For other energy providers, the DAO can act as a virtual power plant which generates renewable energy. Due to the energy would be generated within Switzerland, the cost and lost of transportation would be lower compared to the purchase of energy abroad.

With this concept, the monopolistic Swiss energy market could be disrupted massively. Electricity could be freely traded peer to peer. The limitations through cantonal borders, would fall away, even trade across national borders could be possible at a later point. Current energy providers would take the role of a hardware provider, ensuring the maintenance of the energy grid and maybe the smart meters financed through fees for the usage of the infrastructure.

4 Solution Design - Stefan - around 1100 words

If we draw attention to the code behind our idea we can divide it into two parts: backend (solidity) and frontend (html, javascript, metamask). The purpose of the frontend is to give a simple but hopefully not less secure interface to the user so he can easily access the blockchain. The backend consists of the wallet contract (managing the tokens) and an example implementation of how some of the transfers could work. It is still not fully realistic because the complexity would increase heavily if you make the model more realistic. The focus of the following chapter is on the backend part because the frontend is quite simple. Frontend is just about letting MetaMask call some prepared functions of certain contracts on the blockchain, and about listening to events of the contracts (and showing these in a stylish and simple way to the user).

We will start with the abstract model of what we tried to build. Then the main part is about the implementation (in solidity) and it's difficulties. The last part of the solution design will be about some problems with this implementation.

4.1 Idea

The main target of our implementation is, that it has to be secure. So we try to do it as simple as possible (then there are less sources of errors). Every user should be able to get a wallet where his tokens will be stored. Nobody else should have the right to take tokens out of that 'bag'. But there should be some automation for the transactions (e.g when a user produces for or buys energy from the connecting network at his home).

To create the connection between the real world and the blockchain we use the concept of SmartMeters. Any organization that got the trust of people in its environment could offer this service. SmartMeters should measure the produced and consumed energy of users and report it to the blockchain, where SmartContracts between the user and the connecting network will transfer tokens dependant on the individual implementation of these SmartContracts.

4.2 Our Model Implementation

Wallet This contract is responsible for the ICO and it holds the balances of tokens. Therefore it should be really really simple. If there is a bug that can be exploited in this contract the whole concept could not work. We explain shortly the functions of this contract:

- When the wallet is constructed it will initialize the amount of coins on the market to $1e20$. We did not spend a lot research in how much this should be, so this could be changed. The owner is the creator of this contract. The owners address will get all the Ether that is given to this contract (the ether from the ICO).
- ‘transfer’ transfers an amount of tokens from the callers account to a destination account. It’s quite simple. It only works if there are enough tokens on the callers account and if the transaction happened it will fire an event, there is no need for overflow check because there are not enough tokens initialized to create an overflow. Frontend programs can listen on the fired events. In the ethereum blockchain there is an extra section where fired events are listed, so everyone could check or listen to them.
- ‘buyTokens’ is the only payable function of this contract. It’s used for the ICO. For simplicity it just does a normal transfer of tokens from the wallets account to the callers account. The amount of tokens is equal to the amount of wei paid. Obviously the transaction gets cancelled if there are not enough tokens on the wallets account.

4.3 Model Scenario

The Wallet is the only part that will be the same for all transactions in different regions with different producers and consumers. The other three contracts just show a possible scenario how SmartContracts could do the transactions automatically.

SmartMeterBackend contains functions that can be called by the real (hardware) SmartMeter that measures electricity in and outflow of a producer/consumer. In our implementation it has just one function that reports energy production to the network operator.

The OperatorBackend will need some more explanation. There are functions (that are called by the owner, as example EWZ) that set the price for electricity from this network and the reward for feeding energy into it. Aside from that there is a list of trusted SmartMeterBackends. Only trusted SmartMeterBackends will be able to trigger a payout of tokens to the producer (they can call the rewardProduction function). Now, if a user wants to buy energy, he can call the function buyEnergy with a callback that sends tokens to to the operator if and only if the operator fires the event that reports the transaction as Event.

The User is an example contract of how a producer/consumer could implement his contracts. There is a function to ‘pay for energy’ and a function to transfer tokens to other users (as example if Bob wants to sell produced electricity to Alice). The energy supply will be done by the operators of the networks but if both operators (the one of Bob and the one of Alice) allow to pay for energy with tokens, it will be as if Bob sold his energy to Alice. The ‘pay for energy’ function calls the ‘buyEnergy’ function of the linked operator and sets up all required variables and functions for the transaction (it sends the callback function and sets the price, that the callback function will expect).

4.4 Problems and Possible Solutions

One problem could be the privacy. There may be users that don’t want, that everyone could know how much energy they consume and produce. This is a hard to solve problem because one of the main features of the blockchain is transparency. An approach for a solution could be, that the operator and user handle the transaction between them outside the blockchain (this requires trust from the user side). Still some transactions would be visible on the blockchain.

Another issue is, that in the implementation of the scenario (not in the implementation of the wallet) are still features, that could be exploited by the operator or the user. As example the operator could not trust the SmartMeter. Then the token transfer would not get triggered even if the consumer produces energy. This case is not that much a problem because all actions get logged in the blockchain and the trust in the operator would decrease much, if such things happen.

Another possible weakness of the system could be the SmartMeter itself. A user could cheat if he manages to ‘hack’ the SmartMeter. A possible solution for this would be a SmartMeter that has some features that prevent hacking (as example it destroys its private key, when it notifies an attempt).

5 Conclusion/Outlook - Michael S. - around 600 words

A very important part are reliable measurements of energy production and consumption. It would be ideal, if smart meters could provide the data directly to the network. But it’s highly possible that this does not work yet. As the trust in ewz is very high, and they already check energy consumption manually, this is a service EWZ might provide.

Another problem might be on the legal front. The DAO is still a young concept with no laws existing yet. There is also the question, of how the certificates are handled in such a system.