Liber Acta

Blockchain Marketing Engine

White Paper
November 2018
LiberActa

Contents

Introduction

The market of couponing, gifting and loyalty promotions today is ruled by closed solutions that leave little room for interoperability and in which few actors control most of the volumes.

On one side there are services provided by proprietary platforms implemented by a handset of specialized companies that offer their expertise to help create promotional campaigns.

On the other side we can find in-house solutions developed typically by big retailers and important brands that want to have complete control over all the steps of their promotions.

The solution proposed by BME represents a third way that exploits decentralized technologies to offer a promotional platform that let the client to define their own campaigns and at the same time guarantees fairness and transparency for the consumers. The rules of the promotions are enforced by smart contracts and that offers additional benefits such as automation and real time accounting. The specific design of the BME smart contracts facilitates the interoperability between promotions, simplifying the coordination of campaigns issued by different players. It provides also an innovative approach that enforces the transparency towards the consumers increasing their trust end involvement.

Summary

Loyalty programs are more about gaining customer insights than offering discounts or granting rewards.

Despite the technology behind BME can be applied to different domains, the first business cases implemented and validated concerned promotions issued by retail operators, restaurants, food delivery.

A KPMG survey revealed how 90% of millennials applies to at least a loyalty program and their preference is for programs at department and grocery stores, restaurants, food delivery, entertainment and gym.

The trust of consumers on brands and institutions today can't be taken for granted, and it's a trend expected to increase in the near future. Famous names or expensive advertising alone are not enough to involve consumers in strong loyalty programs. On the other hand a reward system based on smart contracts, intrinsically transparent and fair, can increase customer confidence regardless the brand or the size of the connected companies. This shift in perspective, in which the confidence is enforcer by internal, tamper free rules rather than on brands or on certified third parties, is an element of democratization that allows small stores to share the same benefits of big retail operators.

This new approach has another remarkable effect: marketing campaigns today can't overlook the impacts of the reputation systems managed by platforms such as Amazon and eBay or by social networks . of this new approach has a This new approach leads is a remarkable side effect …

Personalized and targeted programs

Big names that want to maximize their return of investment on promotion and loyalty projects

The problem

According to a KPMG report [https://assets.kpmg.com/content/dam/kpmg/be/pdf/Markets/is-it-time-to-rethink-your-loyalty-program.pdf], “38% of consumers reported a problem with a loyalty program in the last 6 months” and the most common issues were the redemption phase and the calculation of the reward.

Errors in a promotional campaign is counterproductive, causing disaffection and distrust among the consumers.

The same paper cites a survey stating that 19% of consumers would never return to a brand that has been hacked and the 53% is not happy that brands store personal information.

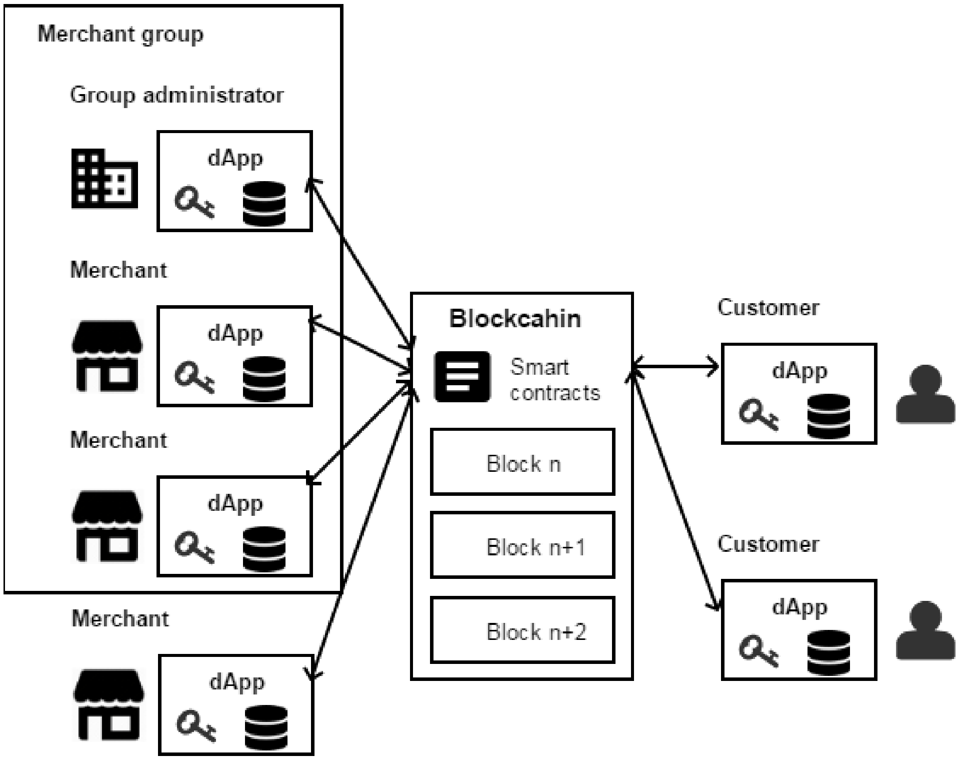
Introduction

Liberacta has based the decentralized features of its BME platform on InterPlanetary File System (IPFS) and the public Ethereum blockchain and strongly believes in the future of these two technologies. Some important topics are still under research by the Ethereum community, such as overcoming its scalability issues, increase the throughput and enhance the privacy. It's not realistic to expect a stable solution to these problems before a couple of years. Anyway Ethereum offers many strong features that can be used already today: immutability, real decentralization and a solid resiliency. Another important element to consider are the transactions fees. Today they represent a prohibitive cost for many applications. Some of the proposed solution suchs as Casper (the Proof of Stake based consensus algorithm) and the sharding of the blockchain will have a huge impact on the dynamics of the costs, however it's difficult today to predict how and to what extent they will change.

A platform with a customer oriented vocation that wants to provide added value for its users from the first day should start from real world use cases rather than from the technology. LiberActa has chosen to use the Ethereum features available today, overcoming the drawbacks of the technology with an architecture that combines blockchain and cloud based services. Clients have the possibility to invoke an Application Programming Interface (API) exposed by BME. It's a more conventional solution that smoothen the transition towards decentralized business models. Even if Ethereum is undoubtedly one of the most mature blockchains, it's still a platform in evolution and BME can mitigate the impacts of its changes. Introducing an API layer between the blockchain and the users involves some degree of centralization however, as Ethereum will evolve and get more mature, the number of use cases that can be implemented directly on chain without invoking additional services will increase. We expect to gradually remove in the future cloud based services conceived to improve the response time and decrease the transaction fees and that BME will be used as a real decentralized solution.

Decentralized architecture

If the use case involves high value - low frequency transactions, then BME can be used in a pure decentralized setting involving only smart contracts and dApps (Distributed Applications) and all the operations are done on-chain. The picture below shows the architecture of BME under this scenario.

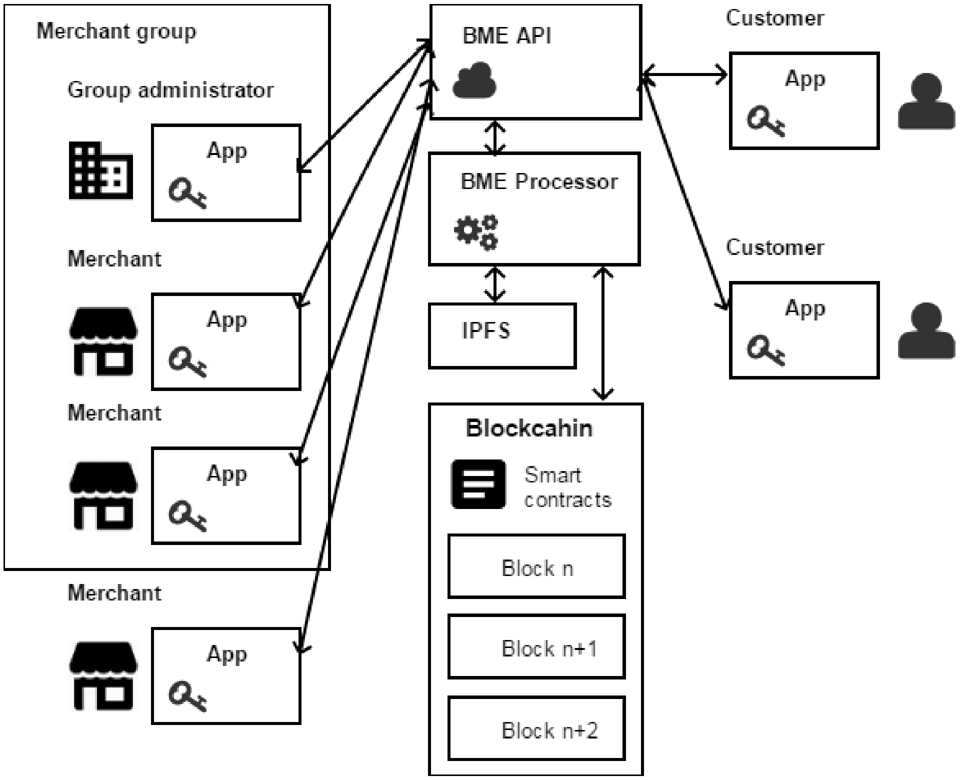


BME provides a set of smart contract templates covering the most common scenarios. The merchant chooses the ones that fit better with her promotion and customizes the behavior setting initialization parameters, such as deadline and the maximum number of coupons. Merchants can also be managed in groups or small hierarchies. In this case a group administrator instantiates the smart contract and set the rules of the campaign. For example a coupon issued by a merchant of the group can be redeemed by another merchant of the same group. The merchant that received the payment from the customer should compensate the other merchant according to the reimbursement policies of the group. With a smart contract the merchants need to trust neither each other, nor the group administrator that instantiated the contract. They have just to evaluate if the rules of the contract are good for them, then they can be confident that nobody will cheat. Many campaigns promote the active involvements of the customers in actions that can increase the customer base. For example they can incentivize the transfer or the sharing of coupons between users. That intruces an additional complexity for the clearing and the accounting of the promotions that s mart contract manages pretty well and in almost real time. The merchant and customer dApps can send signed transactions directly to Ethereum.

The redemption of the coupons in most cases involves off-chain operations, for example in case the merchant is a restaurant, it could represent a dinner. If the merchant could link the coupons to be redeemed to her reservations software, she can provide a better service. She would also be able to do a reconciliation between the payments and the redemptions in her accounting. The integration of the dApp with external software is sometimes seen as a marginal feature in many blockchain projects, however it represents a decisive aspect for highly structured organizations. Here the merchant is responsible for managing sensitive information of the customer. If there isn't third party involved, she's both data controller and data processor according to the GDPR rules. In such architecture the BME services will focus more on these added value features rather than on the management of the transactions that will be delegated to the blockchain.

Mixed architecture

In case of low value - high frequency transactions (for example coupons issued by a coffee shop) an on chain solution involving an Ethereum transaction for each user operation would be today too slow and expensive. In such scenario BME can shorten the response time and lower the Ethereum fees putting a cloud based service that accepts and validate in almost real time the operations of the user, accumulates them in batch of operations and registers the batch in the blockchain. The writing on Ethereum doesn't affect the response time of the users, however there are some implications. Bigger batch sizes imply lower fees but also longer delays before an operation has a trace in the blockchain. During this delay the solution is neither better nor worse than a conventional cloud based architecture. In fact the validation of the operation, including the check of eventual double spending attempts, is done in the service and not directly by the blockchain.

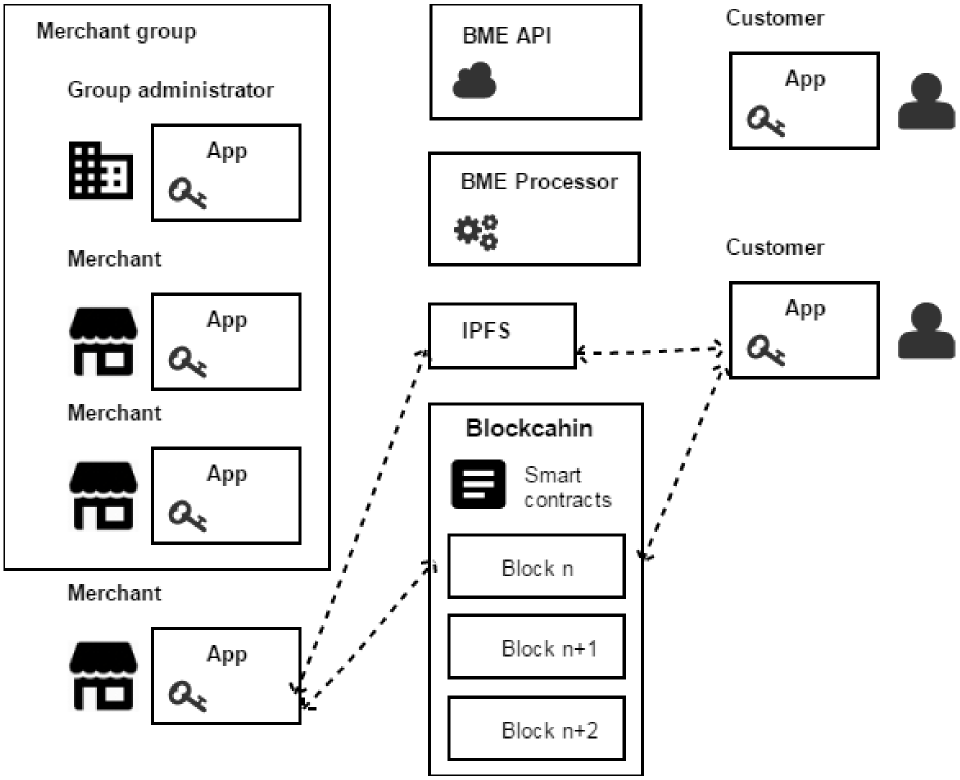


In order to explain the architecture let's consider a simple use case: a restaurant issues a coupon for a dinner to a client that later redeems it supposing the smart contract has been already instantiated by the merchant.

1. The client displays on her smartphone the QR code corresponding to her public key
2. The merchant reads the QR code with her app and assign the coupon. When she confirms the operation, the coupon along with the public address of the client are signed with the merchant's private key. The signed transaction is sent to the BME API on the cloud. This operation is not affected by the response time of Ethereum and is in almost real time. The current implementation uses Google Cloud platform so is highly scalable and resilient.
3. The client can verify on her app that the coupon has been assigned. At this phase the assignment is guaranteed by the BEM APIs, since nothing is still written in Ethereum, however

the transaction received by the client is signed by the merchant and by the operator (in this case BME), so later she will be able to proof that the merchant has created the coupon and that BME has taken in charge the operation..

4. Transactions are organized by BME in a simple Merkle Tree. Once the hash of the Tree root is stored in Ethereum, each user can verify that the transactions are tracked in the blockchain.
5. BME Processor gets the transactions Merkle Tree from BME APIs and publishes it on IPFS that returns an hash that uniquely identifies the Tree.
6. BME Processor stores in an Ethereum smart contract the hash provided by IPFS.
7. From this moment everyone can verify the transactions tracked on the blockchain. The transaction that issues the coupon contains a reference to the address of the smart contract. The client that received the coupon so can get from the smart contract the IPFS address of the Merkle last Tree processed by BME. The picture below shows how with the introduction of IPFS the verification phase can be independent of BME.



The drawback of this approach is that the validation of the transactions is not on-chain but must be done by the cloud based service of BME. It implies that the parties need to trust the platform at least for the time interval between the transaction and the commit to the blockchain, however there are two points that must be remarked:

- The signing of the transaction by the operator (in this case BME) in step 2 is not a feature common to other similar mixed architectures. Here it is used to smooth the transition to a decentralized solution and to allow the operator (BME) to be responsible of the validation. If a user receives a transaction signed by the operator she know that the operation will be responsible of its correct execution.
- In order to have a fraudulent activity it’s not enough that one of the users is a cheater, but she has also to collude with the operator. Anyway after the Merkle Tree is published the fraudulent activity can be easily proved. A common

approach is to introduce cryptoeconomics rules to discourage malicious actions, such as a deposit stored by a smart contract.

Few considerations about the future improvements:

- Step 6 can be optimized for some specific cases. For example for non fungible tokens such as coupons, the simple Markle Tree can be replaced by a Sparse Merkle Tree working as a UTXO. In this case it's not strictly necessary to publish the Merkle Tree on IPFS. This optimization however wouldn't fit well with gift and loyalty cards, so at the moment, so at the moment the priority has be given to the stabilization of the whole platform.
- One can easily find some similarities with Plasma, specifically with its variant based on the Proof of Authority. However the main differences here is that the exits are not present and no deposits are asked to the customers. From a technological point of view a Plasma implementation could fit well with this architecture, but at the moment the drawbacks that it imposes on the user experience block its adoption in many of the user cases. In fact it must be remarked that generally coupons don't have many transactions and a deposit and an exit could have huge impacts on the whole process. It makes more sense for loyalty cards where a same card can be used for many transactions, so it's an option still open.
- For the mixed architecture BME provides a smartphone app or alternatively a progressive web app. They can't be strictly considered dApps because they need to connect to the BME API to interact. The progressive web app is proposed to simplify the adoption of BME. It doesn't need to be installed and can be easily integrated with third party applications, however the security is enforced entirely by the BME APIs so it doesn't differ much from a conventional cloud based application. It still have the commit on the blockchain and the immutability of historical data. The native app is preferred because it can handle user's private key and can sign the

transactions, so the non-repudiation is cryptographically enforced.