Blockchainizator

Libra and new tokens

1 New tokens, why?

Descriptions and papers about the Libra blockchain and its Move language were released last month. Big actors are supporting it. The main goal of the crypto-currency Libra coin is an affordable, broad, & fair money transfer and payment system for everyone.

As often with blockchain and DLT (decentralized ledger technology), the 1st implementations are around finance, followed by B2B and organisation areas. We, the people active in blockchain for gaming & digital collectibles, are used to provide blockchain products for a large number of B2C users, mainly consumers in the media & entertainment areas. It seems that we share with Libra our common mission to deliver blockchain service "for everyone".

So, is Libra a good opportunity for gaming & collectibles? Can we provide new powerful tokens for games and collections?

2 The structure of this analysis

To answer previous questions, I read the papers, and kept as excerpts the parts related 'in general' to our activities around gaming & collectibles, and the ones more focused on "tokens", the needed element for digital assets ownership and apps around it. This part is available in the table with some comments at the end of this article.

Additionally, I summarized key points and related thoughts in the next chapters.

[For more detailed info about 'blockchain for gaming & collectibles', read The Early Evolution of (Some) Art On The Blockchain, What the heck are cryptocollectibles, anyway?, my checklist Blockchain tokens for gaming & collectibles.]

3 The key points of Libra for gaming & collectibles

From previous experience of gaming & collectibles products on other blockchains and the analysis of Libra with its June 2019 publications (with excerpts and comments hereafter) key points are emerging that should support games and collections.

(**a**) Libra 'resource' (token) and 'module' (smart contract) should allow to produce and manage efficiently robust and powerful tokens for simple collectibles to more complex games and features around digital assets.

(**b**) Libra and Move should allow the creation of complex applications suited for some part of "game design mechanics" (parts of the game's application that run as features on the blockchain)

(**c**) The announced priority of payments and financial applications on the "mainchain" could deprioritize, delay, or block the creation and publication of collectible tokens and blockchain games.

(**d**) Libra and Move are designed to evolve in the future for a massive amounts of quick & cheap transactions, but not all on its main-chain. The alternatives are around sidechain, concurrent computing, parallelization, sharding, etc.

Three other aspects to keep in mind, as identified barriers against broad blockchain gaming & collectibles usages:

- The key role of wallet ease of use: potential wallet within chat apps already used by hundreds of millions should foster broad acceptance, other initiatives for UX-friendly wallet (similar to ZenGo) would be well suited for Libra.
- One flexible token format avoiding fragmentation that is difficult to manage for wallets and apps: early definition and implementation of common token standard with planned evolution potential is the way to follow.

- Tokens trade liquidity to bring enough buy & sell orders together: a DEX (Decentralized EXchange) at the same deep blockchain level like the tokens format, having only the core swap features, but allowing anyway other transaction services like marketplaces, auction, etc. to be provided on it by 3rd-parties.

4 The recommendation

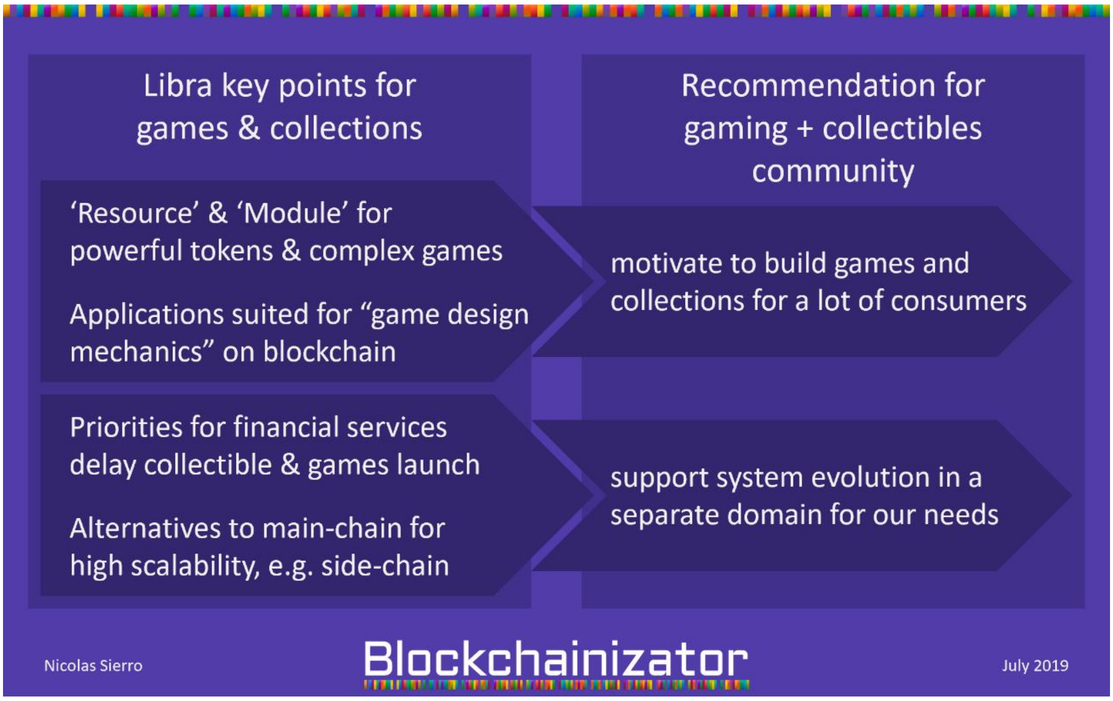
Let's review our four previous points in a simple way.

Libra tokens (**a**) and Move features (**b**) clearly motivate to build games and collections on this blockchain having the potential to reach a lot of consumers.

Gaming and collectible actors do not desire to delay or jeopardize their projects, due to the different interests and priorities for the launch of the core initial Libra financial applications for money transfer and payment (**c**). We find a way to progress by supporting it in a separate domain, and so not disturbing it.

There are solutions to build games and collections products on another domains that the main Libra blockchain networks (**d**). Triple advantages:

- We have specific blockchain network areas for our needs (powerful tokens features, massive quick & cheap transactions for game design mechanics): the gaming + collectibles Libra domain.
- As the main Libra blockchain and our gaming + collectibles domain share the same foundations, it will be easy for provider and enduser to transfer partially some of our specific tokens to the main Libra blockchain for (temporary) broader tokens exposure and trading.
- The gaming + collectibles community supports Libra in its evolution by defining, prototyping, and running their initial implementations of some of their off-chain features, e.g. side-chain, parallelization, sharding, concurrent or local computing and consensus. Looking at the history of blockchain games and collections, this will bring qualitative (games complexity) and quantitative (high number of features, transactions, and users) feedback and experience on the Libra blockchain running several such apps.



Libra key points for games & collections

'Resource' & 'Module' for powerful tokens & complex games

Applications suited for "game design mechanics" on blockchain

Priorities for financial services delay collectible & games launch

Alternatives to main-chain for high scalability, e.g. side-chain

Recommendation for gaming + collectibles community

motivate to build games and collections for a lot of consumers

support system evolution in a separate domain for our needs

Nicolas Sierro

Blockchainizator

July 2019

5 Next steps

Let's start the discussion. Potential online channels are

- Libra community: Community Forum, Telegram group, Discord group, (community on Discord)
- Gaming & collectibles community Telegram: Crypto Gamers Community, Blockchain Games egamers.io, Crypto Bit Games -(for blockchain game developer Blockchain Game Alliance , then join on discord & Telegram)

Depending on feedback and interests that I will anyway collect and summarize, a follow-up post could be released this summer.

6 Libra papers excerpts

, and comments relevant … | 🟨 for token | 🟩 in general

page	Libra White Paper [LibraWhitePaper_en_US.pdf]	
5	Specifically, Move is designed to prevent assets from being cloned. It enables "resource types" that constrain digital assets to the same properties as physical assets: a resource has a single owner, it can only be spent once, and the creation of new resources is restricted. The Move language also facilitates automatic proofs that transactions satisfy certain properties, such as payment transactions only changing the account balances of the payer and receiver.	🟨 Libra token is a 'resource' with the core features needed for this type of digital asset.
6	The Libra Blockchain is a single data structure that records the history of transactions and states over time. The Libra Blockchain is pseudonymous and allows users to hold one or more addresses that are not linked to their real-world identity	🟩
9	An additional goal of the association is to develop and promote an open identity standard.	🟩
page	Libra Blockchain [the-libra-blockchain.pdf]	
1	Transactions are based on predefined and, in future versions, user-defined smart contracts in a new programming language called Move.	🟩 Smart contract (in Libra: 'module') allow to add rules to token ('resource') or to run game design mechanics & collectibles behaviours
3	A client can optionally create a replica of the entire database. Other clients can read from a client that holds a replica in the same way they would read from a validator to verify the authenticity of the response.	🟩 Interesting for gaming & collection analytics
4	The user can also rotate the key used to sign transactions from the account without changing its address	🟩
5	The rules for mutating, deleting, and publishing a resource are encoded in the module that created the resource and declared its type. A module must be uniquely named within an account. On the other hand, the account at address 0x34 could declare a module named Currency.	🟨 Publishing and managing resource (token) is done by an account and the related module
6	In the current version of the Libra protocol, modules are immutable. Once a module has been declared under an account address, it cannot be modified or deleted, except via a hard fork. We are researching options for a scheme to enable safe module updates in future versions. At a high level, a transaction consists of a transaction script (written in Move bytecode) and arguments to the transaction script. The event list is a set of side effects produced by executing the transaction. (allow sequence of	🟨

	triggered actions). Transactions can only generate events — they cannot read events.	
7	A client can check that the ledger state is correct by re-executing each transaction Ti in the history allows the Move language and toolchain to mature before being exposed to users.	Important of the agenda before more features are added, e.g. for games & collectibles, by external providers
10	Transactions that do not proceed beyond step (2) are not appended to the ledger history. The fact that these transactions were considered for execution is never recorded. The key feature of Move is the ability to define custom resource types, which have semantics inspired by linear logic. Resource types are used to encode programmable assets that behave like ordinary program values: resources can be stored in data structures, passed as arguments to procedures, and so on. However, the Move type system provides special safety guarantees for resources. A resource can never be copied, only moved. In addition, a resource type can only be created or destroyed by the module that declares the type.	? potential to review the potential anyway for unimportant [non-critical] events, even re-use it as ‘non-certainty’ actions for game Promising features for several uses cases Perhaps a restriction for assets’ “true ownership”, or the ability to transmit the control/ownership of an assets family. (could be solved by transferring the modules’ rights, but note really elegant. Alternative: plan within the module itself, such specific use cases. (see next excerpts of page 12)
11	No data field can be accessed outside of its declaring module. Modules contain code values, and resources contain data values.	
12	A module is a recipe for creating resources, but it can create an arbitrary number of resources that can be published under different account addresses.	Way to simulate a kind of “delegation” of publishing rights
12	types and values: booleans, unsigned 64-bit integers, 256-bit addresses, fixed-size byte arrays, structs (including resources), and references.	
14	and multiple replicas can be created to handle read queries in parallel.	(similar to excerpt of page 3 of ‘Libra Blockchain’)
16	Move encourages each user to hold resources in their own account. In the initial release of Libra, we optimize for small accounts.	What is a definition of a small accounts? Few tokens?
16	We are assessing a wide range of approaches for a rent-based mechanism that best suits the ecosystem	Naturally very interesting for the ability to rent token as gaming assets (wanna play with my powerful dragon?)
21	Clients may also submit read requests to query validators for the content of an account in the decentralized database. Read requests do not mutate state and can be processed locally without going through consensus.	Another indication (same direction than “Replica”) about efficient query & read, by separating ‘action’ (write, broad consensus) vs. ‘inquiry’ (read, local quick response). Similar distinction for players vs. spectators (think e-sports)
22	We also select elements of the protocol with parallelization and sharding in mind.	Promise of concurrent computing and data handing: opportunities to implement more game design mechanics on (modules of) the blockchain.
22	The mission of the Libra protocol is to support a global financial infrastructure. We anticipate that many payment transactions will occur off-chain, for	For small amount of gaming & collectibles transactions off-chain payment could be suited. Could be compatible (even with synergy) for game

	example, within a custodial wallet or by using payment channels	“off the main Libra chain”, e.g. a side-chain for that (see previous “parallelization & sharding”)
page	Move: A Language With Programmable Resources	
3	Scarcity. The supply of assets in the system should be controlled. Duplicating existing assets should be prohibited, and creating new assets should be a privileged operation. Access control. A participant in the system should be able to protect her assets with access control policies.	Fully in the sprit of collectibles. (as already tested by markets/users from current games & collections on other blockchain types)
6	a resource can never be copied or implicitly discarded, only moved between program storage locations. … they can be stored in data structures, passed as arguments to procedures, and so on.	
6	… allow resources to be created, modified, and destroyed . Move programmers can protect access to these critical operations with modules. Move modules are similar to smart contracts in other blockchain languages. A module declares resource types and procedures that encode the rules for creating, destroying, and updating its declared resources. Modules can invoke procedures declared by other modules and use types declared by other modules. Critical operations on a resource type T may only be performed inside the module that defines T. Each Libra transaction includes a transaction script that is effectively the main procedure of the transaction. A transaction script is a single procedure that contains arbitrary Move code, which allows customizable transactions. A script can invoke multiple procedures of modules published in the blockchain and perform local computation on the results. This means that scripts can perform either expressive one-off behaviors (such as paying a specific set of recipients) or reusable behaviors (by invoking a single procedure that encapsulates the reusable logic)	Again, promising features for several uses cases Beware: modules & transaction scripts flexibility allows to cover very complex use cases, but this diversity could lead to fragmentation, e.g. for token exchange where common domain for trading ensures the needed liquidity for successful sell and buy orders.
20	Support third-party Move modules. We will develop a path to third-party module publishing. Creating a good experience for both Libra users and third-party developers is a significant challenge. First, opening the door to general applications must not affect the usability of the system for core payment scenarios and associated financial applications. Second, we want to avoid the reputational risk that scams, speculation, and buggy software present. Building an open system while encouraging high software quality is a difficult problem. Steps such as creating a marketplace for high-assurance modules and providing effective tools for verifying Move code will help.	Open for quality 3rd-party development, but with priority for the core goal: payment and financial applications. This aspect could favour the use of a side-chain or another Libra chain (to be off-chain of the main chain) for gaming application, but with the option to switch/swap back to the main chain just to trade or auction the digital assets as token against Libra coin.

Nicolas Sierro - July 2019

Originally published in : blockchainizator.com - Libra and new tokens