Telos worker proposal:
Nodejs module for Chronicle

cc32d9@gmail.com

January 2nd 2020

Telos account: `cc32dnineexp`

Mission

Chronicle is a software package that is interpreting the output of nodeos state history plugin and exporting JSON data to the consumer. The software has been in production for more than 9 months, and a number of third-party projects are implementing the consumer functionality in their own way.

What is missing is a simple way for an application developer to set up the environment for receiving blockchain notifications and start processing the data with minimum effort.

This worker proposal aims to build a nodejs JavaScript library and accompany it with documentation, in order to simplify the process of blockchain integration.

Target

The project targets all Telos application developers. Almost every dApp needs to receive updates from the blockchain.

Solution

The solution will be an NPM module, easy to set up in a standard way, accompanied with a complete documentation for setting up a Telos node and starting to collect the data.

The NPM module will implement a Chronicle data consumer, and emit events so that the application can process transaction traces, table deltas, and microforks.

The documentation will include complete details on topics like the following:

- Preparing a server for a Telos node.
- Setting up nodeos with state history. Two options will be documented: start from genesis or from a snapshot.
- Setting up Chronicle.
- Setting up a nodejs application with the module that will be implemented in this WP.

Also the module will be accompanied with examples, as follows:

- Catching all WordProof events and storing them in a MySQL database (Wordproof is a timestamping application, and its initial development was sponsored by Telos WPS).
- Catching incoming TLOS transfers for an account from head block, then confirming them when they become irreversible, and printing all events to the console.
- Telos UK is building “hash-it” service (https://telosuk.io/hash-it/) where transfers to “publicdomain” accounts contain SHA256 hashes linking content together. The example will catch all such transactions and store the hashes in a MySQL database.

All resulting software will be published as open source under MIT license.

The big picture

Today a number of projects are focusing around processing state history data:

- Hyperion by EOS RIO aims to replace the deprecated nodeos history module;
- IRIS by CALEOS is processing Chronicle data and lets applications subscribe to specific events;
- Spectrum by EOS Tribe is building a history database and providing streaming API for querying the history of EOSIO accounts.
- TELOS DreamStack by Gabo Esquivel aims to build a new solution for history querying and data streaming.

Too many dApp developers are unaware that the legacy history plugin is deprecated by Block One, and alternative solutions exist. Also they keep trying to poll the API nodes instead of subscribing to a stream of events, which would be much more efficient and scalable.

The solution as proposed in this WP will give the dApp developers a new opportunity to use the blockchain in a simple manner. Potentially the new module will also be usable as a base for services like IRIS.

Roadmap

Once it is clear that the WP is supported by the community I will start developing and writing the documentation. It will take about 2 weeks to finalize the work. The first release should be ready within 2 weeks after the WP approval.

Human capital

I’m a senior engineer with over 20 years in the industry of software and network engineering. I started working on various EOSIO related projects in June 2018, and I tried to share and publish as much as possible of anything useful for the community. You may find details of my work in my Medium blog and on GitHub:

https://medium.com/@cc32d9

https://github.com/cc32d9

https://github.com/eos-geneva/escrowescrow

https://github.com/eosio-standards-wg

https://github.com/eosio-ecosystem

https://github.com/EOSChronicleProject

The nickname “cc32d9” is the beginning of SHA256 hash from my real name.

You can reach me in Telegram at @cc32d9 or via email: cc32d9@gmail.com

My previous Telos work proposals:

https://chainspector.io/dashboard/worker-proposals/8

https://chainspector.io/dashboard/worker-proposals/28

Finances

Total requested budget is US$5000, or 100k TLOS in current prices. At $150 per hour, that covers 30 hours plus hosting expenses.

Effort estimation:

Task	**Hours**
Development and testing of nodejs module	15
Development of examples	15
Documentation	10

Working time exceeding the budget is expected to be covered from other commercial projects.

KPI

The project will be considered successful once dApps start adopting and actively using the software in production.