

Libre

en.wikipedia.org/wiki/Libre

Свобода — возможность проявления субъектом своей воли на основе осознания законов развития природы и общества.

ru.wikipedia.org/wiki/Свобода

LibreKnowledge

en.wikipedia.org/wiki/Libre_knowledge

LibreSoftware

en.wikipedia.org/wiki/Libre_software

— LibreOffice

— LibreSSL

Libre.fm

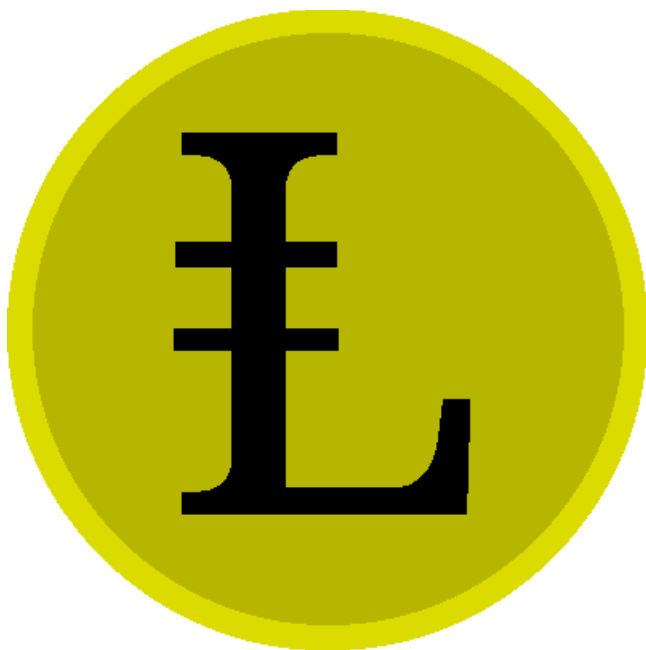
en.wikipedia.org/wiki/Libre_fm

LibreSociety

LibreMoney

LibreUnion

LibreMoney



1. Введение

- 1.1. Термины и определения
- 1.2. ФРС, Bitcoin, валюты, их недостатки и обеспеченность денег
- 1.3. Что такое деньги. Трудовая теория денег

2. Основная часть

- 2.1. Описание
- 2.2. Составляющие элементы LibreMoney
- 2.3. Предварительные правила проектов в системе LibreMoney
- 2.4. Группы и сообщества
- 2.5. Защита от создания пустых денег
- 2.6. Протокол
- 2.7. Программная реализация
- 2.8. API (RestAPI)
- 2.9. Структура управления сообществом разработчиков LibreMoney

3. Заключение

- 3.1. История появления LibreMoney
- 3.2. Немного про Bitcoin, NXT и альткоины
- 3.3. История появления криптовалют
- 3.4. КriptoНотариус- блокчейн
- 3.5. Видео
- 3.6. Что дальше?

LibreMoney

Введение

1.1. Термины и определения

1.1.1. Почему название LibreMoney — LibreMoney — Свободные деньги

1.1.2. Либертарианцы

1.1.3. Что такое Свободные Деньги?

1.1.4. Денежная иллюзия

1.1.5. Экономическая свобода

1.2. ФРС, Bitcoin, валюты, их недостатки и обеспеченность денег

1.2.1. История доллара и ФРС, истоки кризиса

1.2.2. Bitcoin — это крипто-актив

1.2.3. Недостатки существующей системы

1.2.4. Свободные деньги Гезеля

1.2.5. Про ФРС, Bitcoin и обеспеченность денег

1.2.6. Обеспеченность денег

1.3. Что такое деньги. Трудовая теория денег

1.3.1. Трудовая теория денег и прибавочная стоимость

1.3.2. Мой ответ на критику

1.3.3. Теория денег (Маркс) (видео)

1.3.4. Что такое деньги

LibreMoney

Введение

Термины и определения

Почему название LibreMoney (Свободные деньги)

Название состоит из двух слов лат. Libre (свобода) и англ. Money (деньги).

Что означает слово Libre

en.wiktionary.org/wiki/libre

ru.wiktionary.org/wiki/libre

Либертарианцы

ru.wikipedia.org/wiki/Либертарианство

en.wikipedia.org/wiki/Libertarianism



Что такое Свободные Деньги?

Свободные деньги (нем. Freigeld) — понятие введенное немецким экономистом Сильвио Гезелем, для того, чтобы обозначить деньги, используемые только как инструмент обмена (мера стоимости и средство

обмена), но свободные от процентов.

Денежная иллюзия

Денежная иллюзия — термин, описывающий склонность людей воспринимать номинальную стоимость денег, а не их реальную стоимость, выражающуюся в покупательной способности. Иначе говоря, большинство людей больше обращают внимание на цифровой номинал денег, хотя важны количественные соотношения при покупке товаров. Это заблуждение вызвано отсутствием самостоятельной ценности фиатных денег, реальной ценностью которых есть их возможность обмена на товары и услуги, а также возможность платить налоги.

ru.wikipedia.org/wiki/Денежная_иллюзия

Экономическая свобода

Экономическая свобода — это свобода любой деятельности, включающая право выбора и сопряжённые с этим риск и ответственность. (Фридрих фон Хайек)

ru.wikipedia.org/wiki/Экономическая_свобода
en.wikipedia.org/wiki/Economic_freedom

LibreMoney

Введение

ФРС, Bitcoin, валюты, их недостатки и обеспеченность денег

История доллара и ФРС, истоки кризиса

Сейчас самая мощная система свободных денег — швейцарский WIR (нем. Wirtschaftsring-Genossenschaft, Кооператив экономического круга), насчитывающий 62 тысячи участников и обеспечивающий ежегодный оборот в эквиваленте 1 млрд 650 млн швейцарских франков. Эта система была основана в 1934 году, как механизм преодоления платежного кризиса, сложившегося под влиянием великой депрессии. Однако уже в 1952 году они были вынуждены отказаться от теории Гезеля «свободных денег» и сейчас используют ссудный процент.

Логичным продолжением идей Гезелля являются разные варианты «торговых систем местного обмена» (LETS), существующие сегодня как в США, так и в Европе.

Поскольку существует множество систем, альтернативных необеспеченным и кредитным деньгам, а значит, теоретически есть повод избежать катастрофы, все гадают, каким способом эту катастрофу можно предотвратить — мирным или насильственным. Романтики-экологи вроде Маргрит Кеннеди, чья книга «Деньги без процентов и инфляции» стала библией сторонников теории свободных денег Сильвио Гезелля в России, ратует за мирный переход и терпеливое убеждение финансовой элиты в необходимости добровольно отказаться от главной кормушки — кредитных денег. Трудно не то что согласиться, но даже представить себе меру ботанического идеализма, потребную для воспитания в себе веры в саму возможность подобного развития событий.

В равной мере нереальным представляется и насильственное разрешение ситуации, ведь аппарат подавления, экипированный современными технологиями и находящийся на службе «старых денег», настолько превосходит шансы любой оппозиции, что исключает даже намек на осмысленное вооруженное противостояние. Не говоря уж о том, что эффективная система тотального промывания мозгов через СМИ никогда не допустит зарождения «неправильных мыслей» в количестве, достаточном для массовой конфронтации.

Таким образом, нравится это или нет, но наиболее вероятное развитие событий

представляется в образе клюющего жареного петуха — той самой финансовой катастрофы, которой, похоже, не избежать. В подобных обстоятельствах велик соблазн предположить, что рассказ об альтернативных денежных системах — не более чем праздное знание, лишенное практической подоплеки.

Своеобразие темы, однако, заключается в том, что все модели свободных денег изначально создавались не в качестве альтернативы глобальной финансовой системе, а как локальная инициатива, способная облегчить жизнь мелкой общине, поселку, городу, самое большое — области или графству. Существует даже синоним для обозначения разновидностей Freigeld Сильвио Гезелля — community currencies, местные валюты. Идея же полной замены национальных валют свободными деньгами возникла гораздо позже — в период окончательного отлета денежной фантазии от реальности товаров и услуг (в начале 80-х годов прошлого века).

Продолжение:

libremoney.org/article/23 — Теория свободных денег как противовес долговым деньгам

libremoney.org/article/47 — Естественный экономический порядок

Bitcoin — это крипто-актив

Bitcoin и его аналоги выполняют некоторые функции денег. В понимании многих людей Bitcoin является крипто-валютой. Но на самом деле Bitcoin — это крипто-актив с децентрализованной эмиссией. Правильно рассматривать Bitcoin и аналогичные крипто-валюты как акции распределенной корпорации BITCOIN.

Недостатки существующей системы

Недостатки системы

- 1) необеспеченность денег конкретным гарантом
- 2) война валют, раскачка курсов
- 3) мировая централизация эмиссии в ФРС США
- 4) подавление автоэмиссии, подавление альтернативных эмитирующих центров
- 5) вырождение международной распределенной системы долговых расписок

Свободные деньги Гезеля

Гезелль считал, что «свободные деньги» (Freigeld) воплощают идею о «хороших» («правильных») деньгах, которые должны быть «инструментом обмена и больше ничем». По мнению Гезелля, традиционные формы денег неэффективны, так как «исчезают из обращения всякий раз, как возникает

повышенная в них потребность, и затапливают рынок в моменты, когда их количество и без того избыточно». Гезелль считал, что подобные формы денег «могут служить лишь инструментом мошенничества и ростовщичества и не должны признаваться годными к употреблению, сколь бы привлекательными ни казались их физические качества».» — ru.wikipedia.org/wiki/Свободные_деньги

Про ФРС, Bitcoin и обеспеченность денег

Лично я считаю, что ФРС ещё долго будет контролировать глобальные экономики мира. Как минимум 10-20 лет. Но инструментом будет не доллар, а некая другая единица расчета. Роль доллара в будущем — это локальная валюта США, не более того. Одновременно с системой ФРС будут развиваться фиатные системы Китая и альянса БРИКС (Россия+Индия+Бразилия+Китай+...).

А кроме этого, параллельно будут развиваться свободные системы (криптовалюты). Самые интересные на данный момент — это Bitcoin и NXT. Это очень хорошие и перспективные финансовые инструменты. Очень большая вероятность, что курс Bitcoin будет расти и дальше. По прогнозам через пару лет его курс достигнет \$10 000 за 1 BTC. После чего скорее всего немного скорректируется. Это вполне вероятное развитие событий.

Но лично меня не устраивают ни фиатные деньги, ни существующие криптовалюты. В первую очередь из-за не обеспеченности ни тех ни других. Поэтому я смотрю такие инструменты как недвижимость и металлы, а так же на акции и облигации. Но необходимо обеспеченное ликвидное платежное средство и средство накопления.

Обеспеченность денег

Деньги должны быть обеспечены ресурсами (товарами) или другой собственностью, принадлежащей эмитенту. С обычными ресурсами (нефть, газ, золото) и товарами — всё понятно — они находятся на определенной территории, подконтрольной определенным правителям. Но, во первых, государственные эмитенты, по идее, должны выпускать ровно столько бумажек, сколько есть ресурсов в обеспечении. А, во вторых, это не относится к интеллектуальным ресурсам, которые легко скопировать.

Возникает вопрос: а кому принадлежат интеллектуальные ресурсы (технологии, фильмы, музыка)? Есть некие «правообладатели», но по факту любой человек может скопировать фильм, музыку или программу и пользоваться им. Это просто файлы с данными и фактически они принадлежат тому, у кого они хранятся. Поэтому финансирование таких продуктов становится все более затруднительным. Вводятся всякие санкции против пиратов и торрентов, но это

временная мера.

Так вот, модель создания интеллектуальных ресурсов постепенно смещается в сторону совместного финансирования — **краудинвестинга/краудфандинга**. Это мировая тенденция. Каждый такой проект привносит новые ценности в этот мир. Но почему-то под эти созданные ценности производит эмиссию ФРС и/или какой-то ЦБ. Это справедливо? Нет — это не справедливо. Фактически, **тот кто создал новую ценность — тот и должен эмитировать новые деньги**. И реализовать это вполне возможно.

LibreMoney

Введение

Что такое деньги. Трудовая теория денег

Трудовая теория денег и прибавочная стоимость

Публикую сокращенный вариант лекций «История экономической мысли» для более быстрого просмотра. Эти лекции, в частности, хорошо объясняют теорию проекта LibreMoney. Я выделил, самые главные мысли, которые наиболее понятно отражают суть проекта.

youtu.be/F5hcZ_91yIU — Система Кантильона — Предприниматель

youtu.be/gPy7vE-GGA — Адам Смит — Труд

youtu.be/RhFz6wgaf7c — Карл Маркс

libremoney.org/article/510

Мой ответ на критику

Иногда меня упрекают, что я не знаю матчасть. Поэтому добавляю этот текст.

Как оказалось, есть два принципиально разных взгляда на жизнь и на мир денег. Это мир американский и мир советский, есть капитализм и есть социализм. В капиталистической теории куча всевозможных формул, графиков и гипотез и прочего.

Хочу сказать, что я изучал Кейнсианскую теорию, по которой живет западный мир, изучал как происходит эмиссия акций в компаниях, изучал принципы товаро-денежных отношений в древнем мире и в современных реалиях. Когда узнаёшь как было и как стало сейчас, то на ум приходит только один вывод: запад занимается надуванием финансовых пузырей!

В западной теории обязательно наличие инфляции, которое развивает экономику. Но позвольте, при советской власти не было инфляции, а технологии развивались очень и очень большими темпами. Деньги не были ценностью как таковой. Ценности — это то, что создают люди. Деньги лишь количественное отражение ценностей.

Я человек простой, финансовых универов не заканчивал, мозги мне никто своими теориями не запудривал. Поэтому я чётко понимаю: **есть новая ценность — есть эмиссия; нет новой ценности — нет эмиссии**. Все теории про деньги, про эмиссию в западных странах, про эмиссию акций и облигаций — всё это пусть существует, но лично меня не устраивает принципы эмиссии

ни Биткоинов, ни долларов, ни эмиссии акций с размыванием и прочими премудростями. Всё это игрушки.

Эмиссия должна быть только на основе реального труда. Думаю это понятно всем кто мыслит самостоятельно, а не основывает свои рассуждения на западной пропаганде раздувания пузырей.

Теория денег (Маркс) (видео)

Рекомендую посмотреть видеолекции «Теории денег. От ракушки до биткоин» проекта Лекториум (Денежная теория Маркса). Теория прибавочной стоимости и прибавочной ценности лежит в основе проекта LibreMoney.

youtu.be/uXyISpTU8iI — Потребительная полезность товара

youtu.be/o_2gvIkZ9c4 — Функции денег

youtu.be/iyX9LxFXgOQ — Банковские институты

libremoney.org/article/463

Что такое деньги



Деньги не являются ценностью. Деньги — это лишь численное отражение ценностей.

Что такое ценности

Обычно под ценностями понимаются следующее. Ценности — это:

- 1) товары (например хлеб, недвижимость),
- 2) услуги (например подстрижка и т.п).

Так или иначе товары и услуги материальны. Но существуют еще такие объекты как фильмы, музыка, OpenSource программы, технологии. Всё это — не материальные свободные ценности.

Ценность — важность, значимость, польза, полезность чего-либо.

Ценность — как характеристика предмета, обозначающая признание его значимости. Разделяют «Материальные ценности» и «Духовные ценности».

ru.wikipedia.org/wiki/Ценность

Как появляются ценности

Разберем на примере. Предположим мы делаем глиняные горшки. У нас есть кусок глины стоимостью 100 руб и на изготовление горшка мы затрачиваем час своего времени. Предположим час времени стоит 500 руб. Сумма затрат равна 600 руб. Но готовый горшок мы продадим по рыночной цене (предположим это 1000 руб). Есть доход 1000, есть затраты 600, прибыль получается 400 руб. Всё просто.

Но давайте задумаемся. А откуда появились эти 400 руб? Вы ответите, что это очевидно. 400 руб — это прибавочная стоимость. Но какой физический смысл у этих 400 руб? Час труда (500 руб) — это услуга, глина (100 руб) — это товар. А 400 руб — это ни товар ни услуга. Эти 400 руб — ничто иное как прибавочная ценность.

Другими словами, мы не просто делаем горшки, а привносим ценности в этот мир. Её величина называется «прибавочная ценность».

Как появляются деньги

Теперь давайте рассмотрим как появляются деньги. И здесь вы ответите, что всё очевидно. Деньги выпускает государство в лице Центрального банка. Новых денег, по идее, выпускается столько сколько появляется новых товаров и услуг на территории этого государства. Если новых денег выпускается больше, чем появляется новых товаров и услуг, то происходит инфляция. При этом, по Кейсианской теории, считается очень хорошо выпускать как можно больше денег. Это позволяет «развивать экономику». Хотя, на самом деле это называется «надувание пузыря». Но мы сейчас не об этом.

И так. Деньги выпускает государство, под обеспечение товаров и услуг, которые произведены на территории (или под юрисдикцией) этого государства. Это является официальным платежным средством на территории этого государства. Такие деньги называются национальными валютами или фиатными.

Фиатные деньги (от лат. fiat — декрет, указание, «да будет так») — деньги, законные платёжные средства, номинальная стоимость которых устанавливается, обеспечивается и гарантируется государством посредством его авторитета и власти.

ru.wikipedia.org/wiki/Фиатные_деньги

С этим всё понятно. Товары находятся на территории какого-то конкретного государства и услуги предоставляются на территории конкретного государства. И под эти товары и услуги государство эмитирует новые деньги. А как быть со свободными ценностями, которые распространяются свободно в интернете?

Какому государству принадлежат не материальные ценности?

Не материальные свободные ценности

Ведь, де-факто как только фильм или музыка попадает в сеть, то она уже перестаёт принадлежать кому-либо. Да, юридически права кому-то принадлежат. Но мы говорим про то как происходит на самом деле. Фильмы, музыка, технологии и все остальные цифровые «товары» которые опубликованы хотя бы один раз в сети — де-факто принадлежат всему миру.

Очевидно, что разработка цифровых «товаров» — штука ценная. Технологии и алгоритмы, порой, являются очень ценными. Возникает вопрос Какое государство, при этом, должно производить эмиссию новых? Да, очевидно, что если компания и её сотрудники находятся на территории и в юрисдикции определенного государства, то это государство и будет производить эмиссию. Тут всё просто. Но если компания зарегистрирована в одном государстве, разработчики физически находятся в другом государстве, а работу ведут на серверах, расположенных в третьем государстве. А если команда распределенная, работа производится децентрализованно и без образования юрлица (OpenSource разработки)? Что тогда? Где именно произведены ценности и какое государство выпускает новые «обеспеченные» деньги? Все государства сразу? Или разработка не является ценной? Например ядро Linux и другие OpenSource проекты являются ценными разработками или нет? 😊

Эмиссия за не материальные свободные ценности

Мы утверждаем, что за все не материальные свободные ценности эмиссию должен получать их создатель. Под свободными ценностями здесь мы понимаем все ценности, созданные свободными людьми в рамках свободного пространства интернета.

Кто такие свободные люди

Это люди, которые трудятся в интернете. При этом физически человек может находиться в одном государстве, работать в компании зарегистрированной в другом государстве, на сервере, расположенном в третьем государстве. Главное — это то, что результат труда должен распространяться в интернете свободно, без территориальных ограничений. Другими словами — OpenSource.

Свободная валюта

Проанализировав всё это, мы пришли к выводу, что в интернете необходима свободная валюта (криптовалюта). Тем более, что технологии для создания такой валюты уже существуют и успешно используются многими людьми. Это

Bitcoin и его многочисленные форки. Но Bitcoin не является деньгами в нашем понимании. Т.к. эмиссия Bitcoin производится не за полезный труд, а распределяется случайным образом за подбор хеша. По сути вся система Bitcoin и его аналоги попусту сжигают электричество, не производя полезных (научных) вычислений. Поэтому мы говорим, что Bitcoin — это прообраз валюты будущего. Существуют проекты криптовалют, в которых предлагается сделать генерацию блока на основе полезных вычислений. Это правильно, но эти криптовалюты пока на стадии идеи.

Мы решили создать первую свободную криптовалюту, где эмиссию будут получать только авторы интересных и полезных разработок (фильмы, музыкальные произведения, OpenSource программы, изобретатели, разработчики новых технологий). Авторы будут получать эмиссию в качестве премии за полезный труд, за созданные ценности.

LibreMoney

2. Основная часть

2.1. Описание

2.1.1. Зачем?

2.1.2. Что это?

2.1.3. Для кого?

2.1.4. Манифест

2.1.5. Чем обеспечена?

2.1.6. Как это работает

2.1.7. Эмиссия

2.1.8. Механизм краудфандинга в LibreMoney

2.1.9. Механизм венчурного краудинвестирования в LibreMoney

2.2. Составляющие элементы LibreMoney

2.2.1. Активы LibreMoney

2.2.2. Псевдонимы и сообщения LibreMoney

2.2.3. Мульти-шлюз

2.2.4. Почему децентрализованная биржа?

2.2.5. Как работает биржа активов?

2.2.6. Децентрализованный краудфандинг

2.2.7. Безопасная система с открытым исходным кодом

2.2.8. Концепция цветных монет

2.3. Предварительные правила проектов в системе LibreMoney

2.4. Группы и сообщества

2.5. Защита от создания пустых денег

2.6. Протокол

2.7. Программная реализация

2.8. API (RestAPI)

2.9. Структура управления сообществом разработчиков LibreMoney

LibreMoney

Основная часть

2.1. Описание

Зачем?

Цели

- Создать удобную и надежную систему для развития открытых технологий и интернета (OpenSource, фильмы, музыка, наука).
- Создать полноценную финансовую систему.

Основная задача проекта LibreMoney — мотивация людей (стартаперов, предпринимателей) создавать и реализовывать технологические (ИТ) проекты. Это необходимо для дальнейшего развития свободного общества, создания свободного государства без границ, без расовых и языковых расприй.

LibreMoney

2. Основная часть

2.1. Описание

2.1.2. Что это?

LibreMoney — это децентрализованная система краудинвестинга с начислением репутации и Lm-денег (криптовалюта) за успешно выполненные крауд проекты.

LibreMoney — first secured cryptocurrency and crowdfunding system with obtain reputation and Lm-money for successfully completed crowd projects.

LibreMoney — это свободная децентрализованная финансовая система, основанная на существующих протоколах криптовалют с распределенной эмиссией на основе crowdfunding/crowdfunding.

LibreMoney — это децентрализованная краудинвестинговая система и первая обеспеченная криптовалюта с эмиссией на основе реального труда людей.

LibreMoney — это первая и единственная обеспеченная криптовалюта не подверженная ни инфляции, ни дефляции.

Принципы

- Открытость. Полностью открытая система.
- Децентрализация. Децентрализованная система! Нет единого сервера, нет единого механизма передачи данных.
- Устойчивость. Максимальная криптографическая защита.
- Анонимность. Исключить возможность отслеживания реального местоположения пользователя.
- Масштабируемость. Система должна иметь возможность масштабироваться до больших размеров. Вплоть до межпланетарных масштабов.
- Надежность. Четкое исполнение всех задач вне зависимости от желания третьих лиц.
- Однозначность. Система едина. Все действия однозначны. Не допускается параллельных разных подсетей одной системы, не допускается фрагментация системы.

LibreMoney — Возможности

- Совместное финансирование проектов (краудфандинг и краудинвестинг).
- Инвестирование.
- Венчурное финансирование.
- Накопление.

- Торговля.
- Фриланс.
- и прочее.

LibreMoney — Особенности

- До тех пор пока в системе хотя бы три компьютера система продолжает функционирование.
- При временной изоляции подсети система не нарушает своего функционирования. Происходит лишь ограничение круга пользователей, доступных для финансовых расчетов и заключения контрактов.
- Появление и движение денежных средств подчиняются только открытым алгоритмам.
- Сумма денежных средств системы отражает совокупность реальных ценностей в системе.
- Эмиссия производится только при создании новых реальных ценностей.

LibreMoney

Основная часть

2.1. Описание

2.1.3. Для кого?

Для кого разрабатывается LibreMoney

- для венчурных инвесторов: удобный поиск стартапов на ранних стадиях с проверенным спросом;
- для частных инвесторов: возможность вложения в интересные проекты за небольшую долю в проекте (~0.1%);
- для стартаперов (предпринимателе): привлечение крауд-инвестиций на ранней стадии проекта (стадия идеи и прототипа);
- для инвесторов в криптовалюты;
- для OpenSource сообщества — возможность зарабатывать деньги (Donate + эмиссия в качестве премии за хороший проект);
- для всех — полноценная денежная и финансовая система XXI века. Включая такие возможность как децентрализованные компании, умные самоисполняющиеся контракты, электронный нотариус и прочие возможности, которые предлагает технология blockchain.

LibreMoney

Основная часть

2.1. Описание

2.1.4. Манифест

1. Право эмиссии принадлежит всем пользователям.
2. Эмитируемые новые деньги распределяются только за выполненную общественно полезную работу.
3. Для получения эмиссии нужно успешно реализовать общественно полезный проект.
4. Полезность того или иного проекта определяется открытым голосованием.
5. Голосование проводится после завершения работы над проектом в течение фиксированного промежутка времени.
6. В голосовании могут принимать участие все пользователи сообщества.

LibreMoney

2. Основная часть

2.1. Описание

2.1.5. Чем обеспечена?

У LibreMoney обеспечение за счет создания общественных ценностей. Например автор создает фильм, музыку или OpenSource программу и выкладывает свое произведение в сеть. Автор привнес новую ценность в этот мир и под эту новую ценность выпускаются новые деньги (производится эмиссия). Так как эмиссия в системе производится только за реальный труд, за созданные ценности -то мы утверждаем, что все деньги в системе обеспечены реальным полезным трудом. Таким образом мы получаем первую обеспеченную криптовалюту.

Как именно определить величину ценности, и следовательно размер эмиссии? Лично я предлагаю определять на основе собранной суммы при краудфандинге и от суммы голосов при оценке выполненного проекта.

Другими словами, **все новые Lm-деньги обеспечены произведениями авторов крауд-проектов.**

Есть один важный момент. Не все краудрсинговые проекты могут получать эмиссию. Проект должен быть: 1) полезным, 2) востребованным, 3) общественным.

Замечу, что результат работы автора должен быть в итоге доступен каждому человеку. Поэтому эмиссия должна производиться за такие проекты как создание **фильмов, музыки, изображений (картин), технологий, OpenSource-программ и т.п.** То есть результаты таких проектов должны быть доступны в цифровом виде с возможностью свободного копирования и распространения. Предполагается что в транзакциях создания крауд-проекта в системе LibreMoney обязательно должна быть указана ссылка где можно скачать результат (git репозиторий, magnet, torrent и т.п.). Кроме этого автор крауд-проекта должен публиковать результат работы под свободной лицензией (предпочтительно CC0 или как общественное достояние). Результат работы должен быть свободным как от патентных претензий правообладателей, от территориальных и иных ограничений.

Крауд-проекты, которые собирают деньги на лечение конкретного человека, на «поддержку голодающих африканцев», на создание материальных товаров —

такие проекты не могут являться общественно полезными. Даже если деньги собираются на такие полезные дела как очистка территории или создание памятника и т.п. Дело в том, что товар, памятник или территория не могут быть свободно скопированы — значит далеко не все могут пользоваться этим произведением. Это является ограничением свободы. А LibreMoney — это свободная валюта, для свободных людей. 😊

Кстати говоря, первый шаг к свободному государству — это свободная валюта. Об этом еще рано говорить, но кто знает. Возможно скоро мы будем наблюдать появление новых государств, в которых будет своя валюта, свои законы, своя система власти (на основе децентрализованной системы голосования). У такого государства не будет территории и границ, но будут люди, готовые защищать свое государство от внешних врагов. Кто тут сказал «криптоанархия»?)))

LibreMoney

Основная часть

Как это работает

Как определить полезный труд или нет и на сколько



Мы предлагаем авторам работ (предпринимателям), до того как начать разработку сначала провести сбор средств методом краудфандинга. Таким образом можно еще до создания оценить нужность решения/продукта.

То есть производить оценку той или иной работы будем в два этапа.

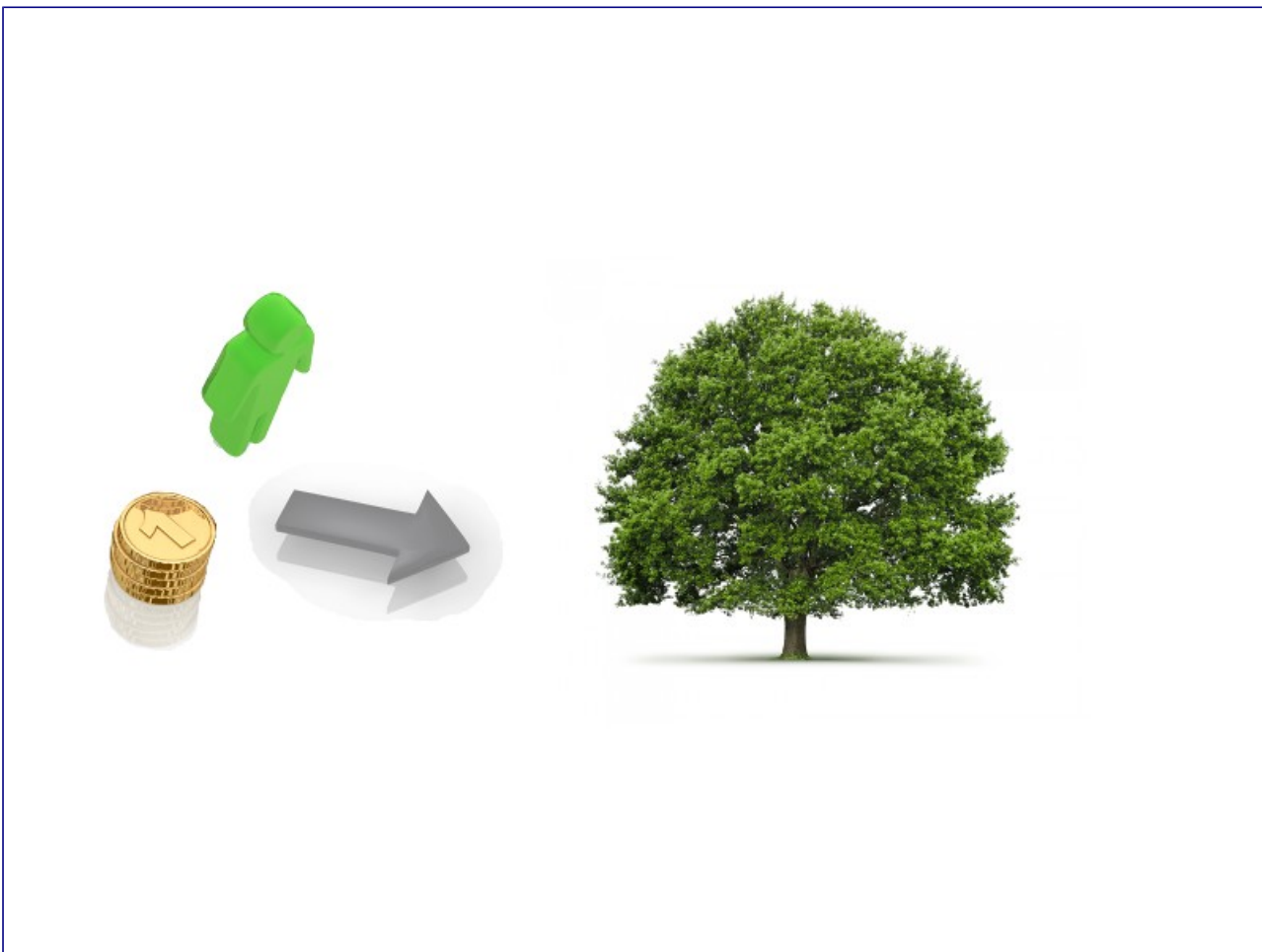
1. До создания, с помощью краудфандинга, определить нужен продукт или нет и на сколько нужен (какова собранная сумма финансирования). Если необходимая сумма на проект собрана, то далее деньги автоматически передаются автору проекта и работа начинается.
2. После реализации, с помощью голосования, определить на сколько хорошо реализован проект. Если проект сделан хорошо, то автор автоматически получает эмиссию в качестве премии. Если проект выполнен плохо, то автор эмиссию не получает и теряет свою репутацию.



Как это работает в LibreMoney

Описание работы системы наиболее понятно получается объяснить на примере. Предположим, что в системе есть несколько участников и есть сообщества. Один из участников предлагает реализовать Crowdfunding проект. Проект в LibreMoney обязательно реализуется в рамках какого-либо сообщества. При этом автор проекта должен являться членом того сообщества в рамках которого планирует проводить сбор средств.

Сообщества в системе необходимы для того, чтобы иметь возможность экспертной оценки проекта при начислении эмиссии. При этом другие участники оценивают самих экспертов. Для этого есть коэффициент лояльности пользователей к сообществу. Поэтому экспертам выгодно объективно оценивать проекты и не допускать в свой круг случайных людей. Для этого размер сообщества жестко ограничен и не может превышать 16 участников.



LibreMoney

2. Основная часть

2.1. Описание

2.1.7. Эмиссия

Если эмитировать деньги за любой труд (как полезный, так и без полезный), то стоимость таких «денег» будет уменьшаться. Именно поэтому в **LibreMoney** **новые деньги эмитируются только после оценки работы автора проекта.** Оценивают работу как вкладчики проекта и эксперты сообщества, так и все остальные пользователи системы. Остальные пользователи системы участвуют в оценке проекта косвенно — оценивая сообщество (влияя на параметр ВесСообщества).

Формула эмиссии простая:

$$\text{Эмиссия} = \text{СобраннаяСуммаНаПроект} * \text{ОценкаВкладчиков} * \text{ОценкаЭкспертов} * \text{ВесСообщества}$$

Где значения коэффициентов ОценкаВкладчиков, ОценкаЭкспертов, ВесСообщества = от 0.0 до 1.0

- 1) Если вкладчикам не понравился продукт, который произвел автор, то эмиссия не производится.
- 2) Если эксперты (члены сообщества) считают что проект технологически выполнен плохо, то эмиссия не производится. Это защищает от проектов-пустышек.
- 3) Все пользователи системы могут оценивать сообщества (+1,0,-1). Если эксперты плохо оценивают проекты, то ВесСообщества будет нулевой и эмиссия за любые проекты производится не будет.

Чтобы экспертное мнение было адекватным размер каждого сообщества ограничен. Прием новых членов в сообщество и исключение из сообщества проводится голосованием. Как и выборы лидера каждого сообщества. Так мы получим много разных тематических сообществ, которые будут конкурировать между собой. Поэтому качество оценки проектов со стороны экспертов будет высоким. Технически реализовать все это вполне реально.

Таким образом, **только полезный труд привносит новые деньги.** И, поэтому, мы говорим, что все созданные деньги в LibreMoney обеспечены общественно-полезным трудом. При этом у каждого общественно полезного труда есть какой-то результат: материальный товар или не материальный (музыкальное произведение, видео, созданная программа, и др). Поэтому, можно сказать, что

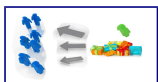
новые деньги обеспечиваются теми общественно-полезными произведениями (товарами), которые создал автор.

libremoney.org/article/189

Пример

Разберем пример эмиссии в системе. Предположим что есть автор, который предлагает собрать дистрибутив Linux. Автор ранее хорошо себя зарекомендовал, есть сообщество, есть желающие помогать как советами, так и деньгами. Но необходимо финансирование. Поэтому автор решает собрать необходимую сумму методом краудфандинга. И так, что нужно сделать:

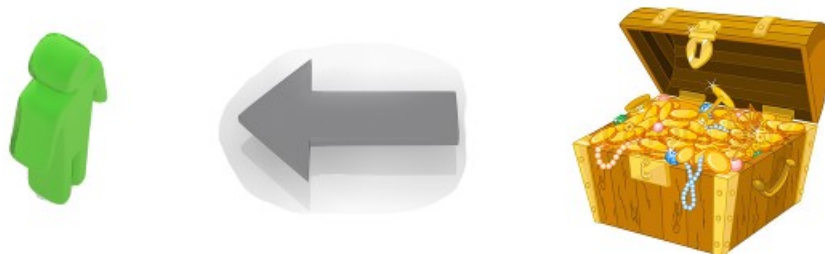
1. Регистрация. Зарегистрироваться в системе, завести кошелек.
2. Сообщество. Если подходящего сообщества еще нет, то необходимо создать сообщество. Либо подать заявку на вступление в имеющееся сообщество и подождать одобрения от экспертов (от членов сообщества).
3. Проект. Создать проект «сборка дистрибутива Linux», задать необходимую сумму, задать сроки сбора средств и сроки выполнения проектов.
5. Сбор средств. После регистрации проекта, все желающие могут профинансировать проект (стать вкладчиками).
6. День окончания сбора средств. Если проект не наберет нужной суммы, то собранные средства автоматически возвращаются.



7. Выполнение проекта. После сбора средств автор получает деньги в свое распоряжение.



8. Час X окончания проекта (DeadLine). После назначенного времени, когда проект должен быть готов, производится оценка. В течение нескольких дней проходит голосование. При этом оценивают проект как вкладчики, так и эксперты (члены сообщества).



9. По результатам голосования определяется коэффициент эмиссии, начисляется эмиссия, а так же прибавляется или уменьшается показатель репутации.

Эмиссия зависит от нескольких коэффициентов и от собранной суммы. В частности от интереса к сообществу, от результатов голосования вкладчиков и результатов голосования экспертов.

libremoney.org/article/201

Репутация

В системе есть численный показатель репутации, который показывает на сколько много и хорошо автор выполнял прошлые проекты. Репутацию нельзя купить, её можно только заработать. Репутация является аналогом показателя Кармы на Хабре.

LibreMoney

2. Основная часть

2.1. Описание

2.1.8. Механизм венчурного краудинвестирования в LibreMoney

Краудинвестинг — составная часть краудфандинга. Главная особенность краудинвестинга — возможность получения вознаграждения за спонсорство, если проект добивается успеха (аналогично инвестированию).

ru.wikipedia.org/wiki/Краудинвестинг

В системе реализована возможность инвестировать в потенциально выгодные стартапы — венчурное инвестирование на ранних стадиях. Автор проекта организует компании по сбору средств с целью начальной реализации проекта с дальнейшим привлечением средств от инвестиционного фонда. При этом фонд «выкупает» доли у первоначальных инвесторов. Это хорошая возможность для авторов собрать посевные инвестиции и проверить свою бизнес-идею. И хорошая возможность для инвесторов-ангелов заработать. А главное возможность инвестировать в любые проекты, без привязки к территории. Естественно есть и риски. Показатель репутации, отчасти, будет являться гарантией.

LibreMoney создается как система для привлечения финансирования для «стартапов» на начальных этапах. Это называется краудфандинг. Но результат краудфандинга обычно — это некие нематериальные блага наподобие постера с личной подписью автора и т.п. Это интересно, но для многих намного более актуально инвестирование с целью получения денежной прибыли в будущем. Это называется краудинвестинг. Система LibreMoney предназначена и для этого. Люди могут вкладывать деньги (Lm) в крауд-проект за долю в этом проекте.

Но это еще не всё. Люди могут помогать не только деньгами, но и вкладывая (инвестируя) свой труд в крауд-проект. Для этого в LibreMoney тоже есть механизмы. Реализовано это следующим образом.

1. Инвестировать можно как деньги (Lm), так и свой труд.
2. В обмен каждый инвестор получает свою долю в проекте в соответствии с внесенным вкладом.
3. Система строится на доверии и на системе репутаций (как автора, так и сообщества).
4. Основная цель венчурного краудинвестирования — это первичный этап

реализации до момента «продажи» проекта стратегическому венчурному инвестору (фонду).

5. Максимальный процент от проекта распределяемый при краудинвестировании должен быть не большим (скажем, 25%).

6. По итогам, когда проект реализован и цель выполнена (получены деньги от инвест-фонда), то каждый участник (инвестор) получает свою часть прибыли.

7. Далее проект начинает жить обычной жизнью стартапа с наращиванием функционала, развитием, привлечением второго и последующих раундов инвестирования.

Другими словами в системе LibreMoney реализована система краудинвестирования на начальном этапе развития проекта (pre-seed). Этот этап является как самым рискованным, так и самым прибыльным.

Как определить долю инвестора, который вкладывает свой труд? Очень просто. Автор крауд-проекта размещает список работ с указанием суммы вознаграждения (доли в проекте). Если находятся желающие реализовать, то они связываются с автором проекта и договариваются. По окончании автор начисляет заработанную долю участнику. Если возникают споры, то они решаются через арбитраж, методом голосования, с привлечением сообщества. Если желающих выполнить задачу за указанное вознаграждение не нашлось, то автор проекта может либо нанять сотрудников за деньги, полученные от инвесторов, либо выполнить работу самостоятельно.

libremoney.org/article/398

LibreMoney

2. Основная часть

2.1. Описание

2.1.9. Механизм краудфандинга в LibreMoney

Система оценки следующая, распишу по пунктам.

1. Автор крауд-проекта делает анонс, объявляет необходимую сумму, устанавливает сроки реализации, указывает за какой вклад кто что получит от этого автора (открытку, постер, имя в титрах и т.д. и т.п).
2. Далее в течение ограниченного срока производится сбор средств на реализацию проекта. Это первый этап оценки проекта. Если проект не интересен или автор не вызывает доверия, то проект, вероятно, не наберет необходимой суммы и деньги вернутся вкладчикам. Это происходит автоматически сама система, выполняя скрипты в транзакции (контракте).
3. После успешного сбора средств, в указанный срок, деньги поступают распоряжение автора крауд-проекта.
4. Далее идет выполнение проекта.
5. В назначенный момент окончания работы над проектом (deadline) автоматически открывается голосование. Голосуют как вкладчики, так и эксперты сообщества. Если к времени голосования проект еще не реализован, то пользователи оценивают такой проект отрицательно и эмиссия не производится.
6. Если проект выполнен хорошо, вкладчики и эксперты довольны, то коэффициент эмиссии будет положительный и, по окончании срока голосования, производится эмиссия в пользу автора крауд-проекта.

В модели начисления присутствуют такие сущности как:

- 1) автор крауд-проекта,
- 2) проект,
- 3) сообщество,
- 4) пользователи сообщества (эксперты),
- 5) вкладчики (спонсоры проекта).

— Автор проекта обязательно должен быть членом сообщества в котором размещается проект.

— У каждого пользователя (эксперта) есть свой вес (репутация эксперта).

— У каждого сообщества так же есть свой вес (репутация сообщества).

— Репутация эксперта определяется по той эмиссии, какую он заработал в

прошлом.

— Чем больше полезных проектов создал эксперт, тем значимее его оценка.

— Репутация сообщества определяется суммой голосов всех пользователей системы.

— Если сообщество не интересно другим и если есть подозрения, что эксперты какого-то сообщества занимаются накрутками (созданием проектов-пустышек), то остальные пользователи системы ставят такому сообществу отрицательные оценки.

— При оценке вес голоса эксперта определяется от его репутации, вес голоса спонсора определяется его вкладом.

В простом виде формула начисления новых денег за проект выглядит так:

**Эмиссия = СобраннаяСуммаПроекта * ОценкаЭкспертов *
ОценкаСпонсоров * ВесСообщества**

Если хотя бы один из коэффициентов нулевой или меньше нуля, то эмиссия не производится.

Кроме этого, размер сообщества жёстко ограничен. А вступление в сообщество и исключение производится общим голосованием участников (экспертов).

Очень важно, чтобы ВесСообщества был как можно больше. Поэтому необходимо раскручивать его, рекламировать, выполнять проекты качественно и в срок. Иначе такое сообщество не будут оценивать положительно. Таким образом экспертам выгодно в свое сообщество допускать только ответственных и надёжных людей. И, в итоге, мы получаем само-регулируемую систему.

P.S. Да, кстати. Автором крауд-проектов в LibreMoney может стать не каждый, а только по приглашению (инвайт). Как на Хабре. Но, при этом, создавать кошельки и пользоваться как платёжной системой может кто угодно без ограничений.

LibreMoney

2. Основная часть

2.2. Составляющие элементы LibreMoney

Активы LibreMoney

Система биржи LibreMoney предлагает почти мгновенную передачу ресурсов, также известную как цветные монеты (colored coins). Цветные монеты являются активами, которые могут быть выпущены в обращение и торговаться с хорошей скоростью в рамках удобного пользовательского интерфейса. Сохраняя биржу активов простой, насколько это возможно, активы LibreMoney могут быть предназначены для представления любой идеи или объекта, включая криптовалюты, фиатные деньги, драгоценные металлы, акции, земли и т.д. Например, в новой системе активов можно было бы торговать ваучерами Amazon или игровой валютой.

Псевдонимы и сообщения LibreMoney

Биржа активов LibreMoney поддерживает регистрацию псевдонимов, что позволяет пользователю связывать информацию по всему Интернету, используя имя, привязанное к своему счету LibreMoney. Хотя система псевдонимов предлагает уникальные перспективы в будущем, разработчики LibreMoney предоставляют интегрированную систему сообщений для совместного использования с псевдонимами. Отправка сообщений позволяет торговцам информировать своих клиентов о скидках или ваучерах и обо всем, о чем захочет знать пользователь. По существу, псевдоним в сочетании с системой обмена сообщениями предлагает широкий спектр функций и оказывает поддержку как потребителям, так и поставщикам.

Скажем, Бетти хочет продать Litecoin (LTC) за Bitcoin (BTC). Она посылает лайткоин на «Hank Litecoin gateway» с сообщением, в котором указан адрес: «Bettys Lm address: 123456789». Хэнк формирует «X hankLTCTokens» и посылает его на адрес Бэтти. Бетти использует его для заполнения самого высокого предложения в очереди заявок и таким образом получает Lm-деньги. Затем Бетти использует свои Lm-деньги для заполнения самого низкого спроса в очереди заявок, чтобы получить lmskysBTCTokens. Она посылает свой lmskysBTCTokens на адрес lmskys Lm с вложенным сообщением «Betty's BTC address:btc123» и ждет пока поступит BTC.

Мульти-шлюз

С запуском биржи активов LibreMoney одним из важных моментов является

контроль за другой электронной валютой, проходящей через систему. Пользователи будут иметь возможность торговать практически всеми существующими электронными валютами посредством Lm, делая Lm стандартом, на основании которого оцениваются все другие активы, по крайней мере, в теории. Валютный интерфейс уже успешно протестирован. На примере Dogecoin разработчики послали его и вывели из биржи активов без каких-либо проблем.

Шлюз мульти-подписей (multi-signature gateway) запрограммирован так, чтобы достичь трех основных целей: эффективность, прозрачность и безопасность. Для защиты и стабильности операций, для системы интерфейса будет организован фонд страхования вкладов Lm.

Почему децентрализованная биржа?

Часто у бирж, инвестиционных фирм, даже при банковских переводах защита инвесторов от потерь невелика, или ее совсем нет. Децентрализованная биржа по сути является торговой платформой, где вы можете торговать множеством различных товаров без необходимости регистрироваться на централизованных биржах, таких как Bitstamp или BTC-E. С децентрализованной одноранговой (p2p) экосистемой LimbeMoney вы будете готовы к работе немедленно.

Как работает биржа активов?

Рассмотрим гипотетический сценарий.

1. Допустим, я хочу запустить бизнес. Сначала я создаю актив (назовем его Krusty Krab tokens: KKT). Я создам лишь 1000 KKT за все время. Чтобы заработать на своем недавно заявленном активе, я объявляю общественности (или пользователям), что собираюсь продать KKT за 20 Lm каждый, и собираюсь выплачивать ежемесячные дивиденды тем, кто купит у меня KKT.
2. Для того, чтобы приобрести эти KKT, вы посылаете 500 Lm на мой счет. Теперь что именно вы будете делать с вашими 25 KKT – это ваше дело (кстати, вы можете их продать), но каждый месяц я буду проверять реестр и смотреть, кто владеет моими KKT. Затем я вышлю 5 Lm за каждый KKT тем пользователям, кто первоначально купил их или купил через биржу. Все это делается автоматически через функцию управления счетом.

Стоимость выпуска актива составляет 1000 Lm или 0,07 BTC (по текущему курсу). В конечном счете, достаточно высокая цена должна препятствовать замусориванию биржи активами, которые могут вызвать неразбериху, или которые не относятся ни к чему ценному.

Мошенники и злоумышленники будут существовать всегда, но с помощью механизма блокировки вы можете уменьшить количество рисков для конечного пользователя. Резонно предположить, что совершенно легитимные пользователи могут создать столько активов, сколько они захотят, и все с одним именем.

Децентрализованный краудфандинг

Lm AE можно также использовать в качестве платформы децентрализованного краудфандинга аналогично www.Kickstarter.com. Благотворительные организации могут использовать биржи активов для пожертвований и других проектов, на которые можно собирать деньги, продавая свои активы. Активы могут представлять собой игровую валюту, серийные номера к программам, электронные книги, прибыль компании или все что угодно, что желает создатель актива.

Безопасная система с открытым исходным кодом

Разработчики LibreMoney устранили проблему пластичности транзакций (transaction malleability), которая преследует протокол биткойна в течение многих лет. Система биржи активов LibreMoney гарантирует участникам, что они никогда не падут жертвой бреши в системе безопасности, вызванной «пластичностью транзакций».

Почти все цифровые валюты, основанные на принципе Proof of Work, страдают недостатком, заключающемся в том, что они требуют увеличения количества энергии в геометрической прогрессии для обеспечения безопасности своих сетей. Экосистема криптовалюты Lm основана на Proof of Stake и имеет конкурентное преимущество в плане расходов на содержание сети перед всеми Proof of Work крипто-валютами, включая биткойн и устаревшие банковские системы. Данные всегда будут идти туда, где дешевле.

Концепция цветных монет

Такие биржи, как LibreMoney, Ripple и Mastercoin работают над идеей под названием «цветные монеты» (colored coins). Что касается биткойна, то один любой биткойн – это то же самое, что и другой биткойн. Они взаимозаменяемы. Цветные монеты целенаправленно не взаимозаменяемы. Они заведомо различны и отличаются друг от друга. Цветные монеты имеют значение, которое к ним прикреплено. Так, например, говоря абстрактно, цветной Lm может иметь прикрепленное к нему значение «Lmsky обещает заплатить 1 BTC тому, кто посылает ему этот маркер, подписанный lmsky». Теперь этим маркером можно торговать на рынке. «Цветная монета» может представлять

собственность, акции/облигации, сырье или даже концепции.

libremoney.org/article/341

LibreMoney

2. Основная часть

2.3. Предварительные правила проектов в системе LibreMoney

LibreMoney создан для привлечения финансирования в высокорискованные (венчурные) OpenSource проекты в области информационных технологий и создания иных нематериальных ценностей (музыкальных произведений, видео, и пр).

LibreMoney не является площадкой для привлечения инвестиций в традиционные бизнесы с просчитываемой моделью роста.

Что не разрешено на LibreMoney?

- LibreMoney не создан для финансирования благотворительных проектов.
- LibreMoney не создан для инвестирования в объекты недвижимости.
- Запрещено использование оскорбительных материалов и материалов, разжигающих ненависть.
- Запрещены проекты, связанные с табаком, алкоголем и наркотическими средствами.
- Запрещены проекты, пропагандирующие какие либо политические взгляды, одобряющие или направленные против каких либо политических организаций или политиков.
- Запрещены проекты, использующие порнографические материалы.

libremoney.org/article/369

LibreMoney

2. Основная часть

2.4. Группы и сообщества

- В системе LibreMoney проект размещается в определенной группе.
- Для того чтобы разместить проект в группе автор должен быть членом этой группы.
- Кол-во членов в группе ограничено (MaxGroupUserCount).
- У каждой группы есть действующий лидер (представитель), роль которого обеспечивать коммуникацию между участниками группы.
- Лидер выбирается на определенное время (GroupLeaderLifetime)
- Вход в группу осуществляется по приглашению от лидера группы и решается голосованием остальных членов группы.
- Выход из группы осуществляется
- У каждой группы есть ВесГруппы — это коэффициент отражающий интерес других людей к деятельности данной группы.
- От коэффициент ВесГруппы зависит размер эмиссии.
- Каждый может создать новую группу. Для вновь созданных групп ВесГруппы равен нулю.
- Пользователь может быть членом нескольких групп.
- Кол-во групп не ограничено.
- Группы объединяются в сообщества на свободных условиях.

MaxGroupUserCount = 16 пользователей

GroupLeaderLifetime = 33554432 sec

LibreMoney

2. Основная часть

2.5. Защита от создания пустых денег

Совокупность репутации каждого автора, и репутация (вес) каждой группы внутри системы даст защиту от накруток. Все транзакции и все проекты является открытыми. Как только появляется подозрение о мошенничестве — то пользователи системы «заминусуют» как автора, так и сообщество в котором он состоит. Кстати, именно по этой причине эксперты (члены сообщества) будут выставлять оценки адекватно, не завышая их.

LibreMoney

2. Основная часть

2.6. Протокол

Система LibreMoney с одной стороны позволяет сохранить анонимность платежей. С другой стороны, созволяет гарантированно учитывать кол-во собранных средств на тот или иной крауд-проект и четко определять какой кошелек должен генерировать блок в данный момент (это необходимо для обеспечения функционирования принципа Proof-of-Stake).

Это реализуется с помощью принципа, похожего на BitSharesX (TITAN+DPOS). Но отличается от него.

Что такое ТИТАН?

ТИТАН — это русская калька с английской аббревиатуры TITAN, с дословным переводом/написанием: «транзакция, невидимая для любого имени / transfer invisibility to any name », и обозначающая тот факт, что операции в сети DPOS анонимны по умолчанию. Другими словами: вы видите сделки и объём, но не можете определить фигурантов сделки. Только отправитель и получатель могут правильно идентифицировать искомую транзакцию.

Каковы особенности?

- Нет необходимости отправлять друг другу уродливые адреса. Вместо этого, в цепочке блоков можно зарегистрировать имя, чего будет достаточно для приёма платежей.
- Получатель анонимен для всех, кроме отправителя.
- Упрощает учёт в кошельке, так как вы можете зарегистрировать произвольные имена и «никто» не может увидеть ваш баланс (кроме вас, конечно)!

Создание расчётного счёта

Вы можете создать принимающий аккаунт, выбрав уникальное (еще не использованное) имя учетной записи, например, BogdanPohlovanM1918. Почти как при регистрации своей первой учетной записи электронной почты. Вам не нужно запоминать адрес — необходимо запомнить только имя. После регистрации вашей учетной записи в цепочке блоков вам нет необходимости использовать адрес аккаунта.

После регистрации своего аккаунта в цепочке блоков, всё что необходимо знать отправляющим вам средства — это имя аккаунта. Больше нет необходимости в

уродливых адресах или открытых ключах, нужно только имя учётной записи. Вы можете зарегистрировать имя вашей учётной записи в цепочки блоков за очень небольшую плату. Чтобы иметь возможность оплатить услугу, вам нужно иметь счёт с достаточным количеством Lm в нём или друга, готовых прийти на помощь, отправив вам некоторое количество Lm (используя адрес, который вы только что создали). После того, как ваш трансфер получен, вы можете зарегистрировать свой аккаунт на цепочке блоков, указав имя аккаунта, который вы хотите зарегистрировать и имя аккаунта, с которого вы хотите оплатить услугу.

Технические особенности регистрации в цепи блоков

Все операции являются анонимными в цепочке блоков по умолчанию!

Механика

Для каждого платежа генерируется временный одноразовый адрес

Подробности действий ТИТАНа можно резюмировать как:

```
dan => DANS_EXT_PUBLIC_KEY
scott => SCOTTS_EXT_PUBLIC_KEY

scott: Generate OneTimePrivateKey & OneTimePublicKey Pair
scott: OneTimePrivateKey * DANS_EXT_PUBLIC_KEY => SECRET
      DANS_EXT_PUBLIC_KEY.child(SECRET) => RECEIVE_PUBLIC_KEY =>
RECEIVE_ADDRESS
scott: RECEIVE_PUBLIC_KEY * SCOTTS_EXT_PRIVATE_KEY => CHECK_SECRET
      => SHORT_HASH(CHECK_SECRET) == SHORT_SIGNATURE
scott-broadcast: OneTimePublicKey + RECEIVE_ADDRESS + ENCRYPT(from scott +
SHORT_SIGNATURE, SECRET)
dan: OneTimePublicKey * DANS_EXT_PRIVATE_KEY => SECRET
      DANS_EXT_PRIVATE_KEY.child(SECRET) => RECEIVE_PRIVATE_KEY =>
      => RECEIVE_PUBLIC_KEY => RECEIVE_ADDRESS

dan: DECRYPT(data, SECRET) => "from scott" + SHORT_SIGNATURE
dan: SCOTT_EXT_PUBLIC_KEY * RECEIVE_PRIVATE_KEY => CHECK_SECRET =>
SHORT_HASH(CHECK_SECRET)
```

Разве это не то же самое что и стелс-адреса в Bitcoin?

Это вариация стелс-адресов, которая использует систему имён, чтобы решить основную проблему со стелс-адресами (общий открытый ключ).

Доступны ли с ТИТАНом ключи «только для чтения» /«только для просмотра» (read-only/watch only keys)?

Да. Основная поддержка уже готова. Мы имплементируем так называемые ключи наблюдателя (observer keys).

Какими дополнительными возможностями он обладает?

Умеет кодировать в данные транзакции сообщение с примечанием, чаще всего используется для указания назначения платежа, сделки.

Источники

bitsharestalk.org/index.php?topic=4887.msg64091#msg64091

Транзакции как доказательство доли

Даниэль Лаример

dlarimer@invictus-innovations.com

28 ноября 2013

Протокол доказательства доли Proof-of-Stake, основывается на концепции, подразумевающей, что безопасность цепочки блоков должна обеспечиваться теми, кто имеет в существовании данной цепи финансовую заинтересованность. В данной статье будет продемонстрирован новый подход к протоколу Proof-of-Stake, в котором разрушение монета-дней в каждой транзакции заменяет подавляющее большинство предоставляемых в рамках протокола доказательства работы Proof-of-Work аспектов защиты сети. В отличие от предшествующих систем Proof-of-Stake, в которых только определённое количество узлов участвовали в расчётах в рамках алгоритма POS, мы представим новое видение функционирования протокола доказательства доли, при котором все узлы сети, совершающие транзакции, будут участвовать в обеспечении безопасности системы. Мы полагаем, что такая сеть будет обладать иммунитетом к тем типам атак, которые могут быть применены против систем Bitcoin или Peercoin.

В этой статье предоставлен упрощенный алгоритм доказательства доли Proof-of-Stake, который позволяет всем пользователям сети внести свой вклад в защиту системы от атак. В рамках этой системы задача создания секретной цепи блоков, которая содержит больше уничтоженных монета-дней, чем общественная цепь сделок, представляется экономически неосуществимой для любого злоумышленника. Представлены методы, позволяющие защитить сеть от 51% атаки, атаки майнинга группой лиц в эгоистических целях и атаки с двойной тратой, осуществимые без майнинга вообще.

Так как протокол доказательства доли Proof-of-Stake устраняет необходимость в выплате вознаграждений майнерам, он так же устраняет необходимость в инфляции, а также устраняет необходимость в затратном потреблении электроэнергии, свойственном протоколу proof-of-work. В мире, с постоянно увеличивающимся количеством цепочек блоков, обеспечение безопасности с помощью протокола доказательства работы Proof-of-Work становится фрагментированным без объединенного майнинга, у которого в свою очередь также имеются недостатки. С переходом на алгоритм доказательства доли Proof-of-Stake, объединённый майнинг больше не является необходимым и

любое количество цепочек блоков может быть произведено без ущерба для безопасности каждой отдельной цепочки блоков.

Подробнее: libremoney.org/article/774

Делегированное доказательство доли или протокол DPOS

Даниэль Лаример | 3 апреля 2014

Эта статья знакомит с новой реализацией протокола доказательства доли, в рамках которого подтверждение сделок занимает секунды, обеспечивая большую безопасность за более короткий период времени, чем все существующие системы, функционирующие под протоколом доказательства доли PoS. К тому времени как в системе Bitcoin будет произведён лишь один блок, ваша транзакция будет заверена уже 20% акционеров, а к тому времени, Bitcoin утвердит сделку и сделает её почти необратимой (6 блоков, 1 час), ваша транзакция под протоколом DPOS уже будет заверена 100% акционеров через своих делегатов.

Процесс применения протокола DPOS, в сочетании с TaPOS, образует систему, которая в три или более раз превосходит системы Bitcoin, Peercoin и NXT, в плане обеспечения доказанного общественного соглашения. Система под протоколом DPOS способна достигать такого консенсуса быстрее, устраняя возможность мелких нарушений со стороны случайных миноритарных акционеров. Экономические стимулы гарантируют, что делегаты будут работать, доказывая своё соответствие занимаемой должности, и, скорее всего, примут алгоритм консенсуса, похожий на используемый Ripple. DPOS является, по сути, средством для получения уникального списка узлов для Ripple в децентрализованном порядке, без опасности раскола сети.

Примечание: В NXT реализован механизм передачи своих средств в «лизинг».

Для этого существует специальный тип транзакции

TYPE_ACCOUNT_CONTROL —

SUBTYPE_ACCOUNT_CONTROL_EFFECTIVE_BALANCE_LEASING. Деньги могут быть востребованы обратно в любой момент. Деньги автоматически возвращаются в колёшек владельца по истечении определенного времени. Этот механизм в NXT работает аналогично делегированию своих денег в BitSharesX.

Подробнее: libremoney.org/article/789

LibreMoney

2. Основная часть

2.7. Программная реализация

Криптовалюты, основанные на PoW не подходят, необходим 100% PoS. За основу мы взяли исходный код криптовалюты NXT (NextCoin). Исходный код полностью переписан на JavaScript (Node.js) и в него добавлен механизм эмиссии.

NXT — это 100% PoS криптовалюта второго поколения. Подробнее:

Что делает NXT криптоплатформой второго поколения

— habrahabr.ru/sandbox/77572

Новые веяния в криптовалютах: 100% proof-of-stake и Nxt

— habrahabr.ru/post/207120

LibreMoney

2. Основная часть

2.8. API (RestAPI)

1. Описание
2. Общие сведения
3. Операции со счетом
4. Действия с Альясами
5. Системные функции для работы с произвольными сообщениями
6. Операции для работы с блоками
7. Операции на Бирже Активов
8. Операции с опросами
9. Операции для работы с серверной частью
10. Операции с клеймами (hallmarks)
11. Операции с токенами
12. Операции с транзакциями
13. Коды ошибок

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Описание

LibreMoney API позволяет взаимодействовать с нодами LibreMoney посредством HTTP GET/POST запросов. Ответы возвращаются в виде JSON объектов.

Все API вызовы делаются посредством общения с нодами LibreMoney по порту 1400. Во всех примерах, приведенных здесь, нода представлена, как "localhost". Все примеры и ответы отформатированы для комфортного чтения, поэтому надо помнить, что переносы строк и пробелы не используются в действительности в запросах и ответах.

Многие вызовы API ссылаются на генезис блок. Для справки, ID генезисного блока: [XXXXXXXX](#)

Со всеми API функциями можно ознакомиться а также протестировать их по адресу <http://localhost:1400/test> Для определенных API функций, для ознакомления и тестирования можно использовать запрос вида <http://localhost:1400/test?requestType=<specificRequestType>> где <specificRequestType> это имя определенной API функции которую вы хотите вызвать

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Общие сведения

Все http API запросы которые создают новую транзакцию, работают с publicKey параметром (подпись осуществляется на стороне клиента, в браузере).

publicKey параметр в виде закодированного байтового массива, то операция будет подготовлена сервером и возвращена в JSON ответе как transactionBytes. Этот байтовый массив может теперь быть подписан клиентом, и затем передан обратно на сервер для широковещательной передачи используя API broadcastTransaction.

У всех вызовов API, которые требуют номер аккаунта, могут принимать номер аккаунта как в виде идентификатор учетной записи (цифровой номер аккаунта) или как адрес в формате Reed-Solomon

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции со счетом

Get Account

Возвращает информацию о аккаунте, ассоциированным с предоставленным номером аккаунта.

Запрос

```
http://localhost:1400/api?  
requestType=getAccount&  
account=ACCOUNTNUM
```

Where:

- ACCOUNTNUM Это ID аккаунта, по которому вы запрашиваете информацию.

Ответ

```
{  
  "publicKey": "PUBKEY",  
  "assetBalances": [  
    {  
      [ARRAY_OF_ASSETS]  
    }  
  ],  
  "balance": "BALANCE",  
  "accountRS": "ACCOUNTRS",  
  "unconfirmedAssetBalances": [  
    {  
      [ARRAY_OF_UNCONFIRMED_ASSETS]  
    }  
  ],  
  "account": "ACCOUNT",  
  "effectiveBalance": EFFBALANCE,  
  "unconfirmedBalance": "UNCONFBALANCE",  
  "forgedBalance": "FORGEDBAL"  
}
```

Where:

- PUBKEY 64-битный публичный ключ ассоциированный с аккаунтом.
- ACCOUNTRS адрес аккаунта в формате Reed-Solomon.
- BALANCE: количество MilliLm которым вы владеете независимо от числа подтвержденных транзакций, связанные с этим аккаунтом
- ACCOUNT учетная запись, создавшая актив

- EFFBALANCE**: количество MilliLm которым вы владеете и которое подтверждено более чем 1440 раз. Это эффективный баланс , который может использоваться для форжинга.
- UNCONFBALANCE**: баланс (в MilliLm) за вычетом всех неподтвержденных исходящих транзакций. Также не включаются неподтвержденные входящие транзакции и double-spending. Пользователь видит неподтвержденный баланс в клиенте.
- FORGEDBAL** баланс который используется для форжинга, в MilliLm

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAccount&
  account=7898870630272254992
```

Ответ:

```
{
  "publicKey":
    "ce0fc0f2fed0645f7018e251493a8af1e1ecfb444a4603902e71bfd2a4b41558",
  "assetBalances": [
    {
      "asset": "1013693125509851736",
      "balanceQNT": "48"
    }
  ],
  "balance": "10007000000000",
  "accountRS": "LMA-BK2J-ZMY4-93UY-8EM9V",
  "unconfirmedAssetBalances": [
    {
      "unconfirmedBalanceQNT": "48",
      "asset": "1013693125509851736"
    }
  ],
  "account": "7898870630272254992",
  "effectiveBalance": 9932,
  "unconfirmedBalance": "10007000000000",
  "forgedBalance": "0"
}
```

Verified 5/12/14

Get Account ID

Возвращает номер счета, связанный с переданной парольной фразой.

Запрос

```
http://localhost:1400/api?
  requestType=getAccountId&
  secretPhrase=PASSPHRASE
```

Где:

- PASSPHRASE — парольная фраза того счета, ID которого вы хотите получить

Ответ

```
{
  "accountId":      "ACCOUNTNUM"
  "accountRS":      "ACCONTRRS"
}
```

Where:

- ACCOUNTNUM ассоциированный с аккаунтом ID.
- ACCONTRRS адрес аккаунта в кодировке Reed-Solomon.

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAccountId&
  secretPhrase=mySuperSecretPassphraseThatYouCanNeverGuess
```

Ответ:

```
{
  "accountId":      "4357314498768237104"
  "accountRS":      "LMA-UWKJ-GFEV-AGY4-5C4YS"
}
```

Внимание: Вы получите ERRORCODE:1, если вы делаете этот запрос не через метод POST

Проверено 14/05/14

Get Account Block IDs

Возвращает ID всех блоков, созданных данным аккаунтом. Результаты отсортированы в хронологическом порядке.

Запрос

```
http://localhost:1400/api?
  requestType=getAccountBlockIds&
  account=ACCOUNT
  timestamp=TIME
```

Где:

- ACCOUNT — номер счета, который вы хотите проверить
- TIME — значение в секундах начиная с создания генезис блока, определяющее время самой ранней генерации блока, который вы бы хотели получить. Установка параметра в 0 даст ID всех блоков, сгенерированных данным аккаунтом.

Ответ

```
{
  "blockIds":
    [ARRAY_OF_BLOCK_IDS]
}
```

Примечание: результаты отсортированы в хронологическом порядке (самый ранний блок отображается первым)

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAccountBlockIds&
  account=387977119245692429
  timestamp=0
```

Ответ:

```
{
  "blockIds":
    [
      "932115710792099523"
    ]
}
```

Проверено 15/05/2014

Get Account Public Key

Возвращает публичный ключ, связанный с переданным номером счета.

Запрос

```
http://localhost:1400/api?
  requestType=getAccountPublicKey&
  account=ACCOUNTNUM
```

Где:

- ACCOUNTNUM — номер счета, который вы хотите проверить

Ответ

```
{
  "publicKey":      "PUBKEY"
}
```

Где:

- PUBKEY — 64-битный публичный ключ, связанный с данным счетом.

Примечание: если переданный номер счета не известен, вы получите ошибку с кодом 5. Если счет существует, но с него не посылались транзакции, вы получите пустое значение: {}

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=getAccountPublicKey&  
account=6483973064837388393
```

Ответ:

```
{  
  "publicKey":  
  "29592749cd26f5c1c4bf0225f8ab225f289449b4978495e6fec09d486ae2d217"  
}
```

Проверено 14/05/14

Get Account Transaction IDs

Возвращает ID транзакций, связанных с переданным номером счета. Параметр timestamp определяет насколько далеко в прошлое вам нужно заглянуть в вашем списке транзакций; нулевое значение позволит вам получить все транзакции данного счета, начиная с генезис блока.

Запрос

```
http://localhost:1400/api?  
requestType=getAccountTransactionIds&  
account=ACCOUNT&  
timestamp=TIME&  
type=TYPE&  
subtype=SUBTYPE
```

Где:

- ACCOUNT — номер счета, который вы хотите проверить
- TIME — значение в секундах начиная с создания генезис блока, определяющее время самой ранней транзакции, которую вы бы хотели получить. Установка параметра в 0 даст ID всех транзакций данного аккаунта.
- TYPE (Optional) тип транзакции которую вы запрашиваете
- SUBTYPE (Optional) тип суб-транзакции которую вы запрашиваете

Ответ

```
{  
  "transactionIds":  
    [ARRAY_OF_IDS]  
}
```

Где:

- ARRAY_OF_IDS — массив ID транзакций.

Примечание: результаты отсортированы в хронологическом порядке (самая ранняя транзакция отображается первой)

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAccountTransactionIds&
  account=25940420118693217452&
  timestamp=0
```

Ответ:

```
{
  "transactionIds":
    [
      "17484500871708980634",
      "51938275284739120529"
    ]
}
```

Проверено 15/05/14

Get Balance

Возвращает баланс аккаунта.

Запрос

```
http://localhost:1400/api?
  requestType=getBalance&
  account=ACCOUNT
```

Где:

- ACCOUNT — номер счета LibreMoney

Ответ

```
{
  "guaranteedBalance": "GUARANTEED_BALANCE",
  "balance": "BALANCE",
  "effectiveBalance": "EFFBALANCE",
  "unconfirmedBalance": "UNCONFBALANCE",
  "forgedBalance": "0"
}
```

Где:

- GUARANTEED_BALANCE общий баланс аккаунта соответствующего критериям. Результат выражается в MilliLm
- BALANCE количество MilliLm которым вы владеете отображается в MilliLm, независимо от числа подтвержденных транзакций связанных с этими имеющимся аккаунтом.

- **EFFBALANCE** количество которым вы владеете и которые имеют минимум 1440 подтверждений. Этот баланс Это - баланс, который используется для форжинга.
- **UNCONFBALANCE** баланс (в MilliLm) за вычетом всех неподтвержденных исходящих транзакций. Он также не включает не подтвержденные входящие транзакции и double-spending. Пользователь видит этот неподтвержденный балансА в клиенте.
- **FORGEDBAL** баланс который используется аккаунтом для форжинга, выражается в MilliLm

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getBalance&
  account=7898870630272254992
```

Ответ:

```
{
  "guaranteedBalance": "0",
  "balance": "10007000000000",
  "effectiveBalance": 9932,
  "unconfirmedBalance": "10007000000000",
  "forgedBalance": "0"
}
```

Проверено 14/05/14

Get Guaranteed Balance

Возвращает баланс счета, который подтвержден определенное число раз.
Реализовано, начиная с версии 0.5.5

Запрос

```
http://localhost:1400/api?
  requestType=getGuaranteedBalance&
  account=ACCOUNT&
  numberOfConfirmations=CONFIRMATIONS
```

Где:

- **ACCOUNT** — номер LibreMoney аккаунта
- **CONFIRMATIONS** — минимальное число подтверждений транзакции, позволяющее ей быть видимой при подсчете гарантированного баланса

Ответ

```
{
  "guaranteedBalance":          GUARANTEED_BALANCE
```

}

Где:

•**GUARANTEED_BALANCE** — общий баланс счета, удовлетворяющий критериям **CONFIRMATIONS**. Выражается в MilliLm.

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getGuaranteedBalance&
account=3847997711292454692&
numberOfConfirmations=15
```

Ответ:

```
{
  "guaranteedBalance":      1303400000000
}
```

Проверено 14/05/14

Lease Balance

Передача вашего баланса в аренду другому аккаунту.

Запрос

```
http://localhost:1400/api?
requestType=leaseBalance&
period=PERIOD&
recipient=REC&
secretPhrase=SECRET&
fee=FEE&
deadline=DEADLINE&
referencedTransaction=REFTXID&
broadcast=BROAD
```

Где:

- PERIOD** продолжительность аренды вашего баланса, продолжительность указывается в количестве блоков.
- REC** аккаунт которому вы передаете в аренду свои средства
- SECRET** секретная фраза аккаунта передающего в аренду свои средства
- FEE** комиссия которую вы платите за транзакцию (в MilliLm)
- DEADLINE** дедлайн транзакции до получения подтверждения, определяется в минутах
- REFTXID** ID предыдущей транзакции на которую вы ссылаетесь. Это создает зависимые транзакции (цепочка транзакций), означая, что текущая транзакция не может быть подтверждена, пока транзакция на

которую вы ссылаетесь, не подтверждена. (опционально)

•BROADCAST если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

Ответ

```
{
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "transactionBytes": "TXBYTES",
  "hash": "HASH",
  "transaction": "TX",
  "broadcasted": BROADCAST,
  "unsignedTransactionBytes": "UNTXBYTES"
}
```

Где:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хеш всех полей транзакции, за исключением подписи
- TX ID транзакции
- BROADCAST статус - транзакция была передана или нет
- UNTXBYTES байты транзакции с частью подписи, заменяемой нулями

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=leaseBalance&
  period=2880&
  recipient=1562462127635514638&
  secretPhrase=123&
  fee=1000000000&
  deadline=1440
```

Ответ:

```
{
  "fullHash":
    "e4d1f83e1cf26ee71d8566d00077510c4a641e04174e7ec3893d7e5aaafc82e65",
  "signatureHash":
    "48cab5351b4c08ac2816b5e13e9d238fa910085a0d07a16e1c7d7986222fc444",
  "transactionBytes":
    "0400401adb00a005668683e673274c671802492d6f5...",
  "hash":
    "9b64d859cecdf63773d0696777c2783892616f3a793794544531e4fd7b8a66b2",
  "transaction": "16676532673329025508",
  "broadcasted": true,
  "unsignedTransactionBytes":
    "0400401adb00a005668683e673274c671802492d6f54cdc..."
}
```


}

Проверено 14/05/14

Start / Stop Forging

Запуск или остановка форжинга аккаунтом, или проверка, чтобы увидеть, форжит ли учетная запись. Учетная запись форжит, когда ее секретная фраза (закрытый ключ) предоставлена серверу/узлу LibreMoney.

Запрос

```
http://localhost:1400/api?
  requestType=START_OR_STOP_OR_GET&
  secretPhrase=SECRET
```

Где:

- START_OR_STOP startForging или stopForging
- SECRET секретная фраза (личный ключ) для аккаунта, форжинг на котором вы хотите начать или остановить

Ответ

```
{
  "deadline":      DEADLINE
}
```

Где:

- DEADLINE оценочное число секунд до генерации аккаунтом нового блока

Для "stopForging":

```
{
  "foundAndStopped":  TRUE_OR_FALSE
}
```

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=startForging&
```

```
secretPhrase=MySuperSecretPasswordThatIsNotActuallyPostedHereOrIsEvenRealSoDoNot
EvenTryIt
```

Ответ:

```
{
  "deadline":      1036742
}
```

}

Проверено 14/05/14

Set Account Information

Установка сведений об аккаунте

Request

```
http://localhost:1400/api?
requestType=setAccountInfo&
secretPhrase=SECRET&
name=NAME&
description=DESCRIPT&
publicKey=PUBKEY&
fee=FEE&
deadline=DEADLINE&
referencedTransactionFullHash=REFTXHASH&
broadcast=BROAD
```

Где:

- SECRET секретная фраза (приватный ключ) аккаунта
- DEADLINE дедлайн для транзакции, задается в минутах
- FEE комиссия за транзакцию (в MilliLm)
- PUBKEY открытый ключ передающего счета для транзакции (опционально)
- NAME имя которое вы хотите установить для аккаунта(опционально)
- DESCRIPT описание которое вы хотите дать для аккаунта(опционально)
- REFTXHASH ID транзакции, на которую ссылается эта транзакция(опционально)
- BROAD true или false если это должно быть передано широкоевещательно (опционально)

Ответ

```
{
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "transactionBytes": "TXBYTES",
  "hash": "HASH",
  "transaction": "TX",
  "broadcasted": BROAD,
  "unsignedTransactionBytes": "UNTXBYTES"
}
```

Where:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций

условного депонирования

- TXBYTES байткод транзакции
- HASH хэш всех полей транзакции, за исключение подписи
- TX ID транзакции
- BROAD указывает была транзакция передана или нет
- UNTXBYTES байты транзакции с подписью, частично замененной нулями

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=setAccountInfo&
```

```
secretPhrase=MySuperSecretPasswordThatIsNotActuallyPostedHereOrIsEvenRealSoDoNot  
EvenTryIt&  
fee=1000000000&  
deadline=1440
```

Ответ:

```
{  
  "fullHash":  
    "93e8bba270eed8693f62576e13b45b45a848dd475e510fdb1e4026712829b734",  
  "signatureHash":  
    "5487681f5a828d379864cbbdba9b1f4f2c6f7035c8cf2f272dc323147aea5eac",  
  "transactionBytes":  
    "0105c84ad700a005668683e673274c671802492d6f54cdc33d247aa426e7...",  
  "hash":  
    "6b6ec6ad2f06119493caca546b6561069cd5341b41884d5be068e0b1d9ce7748",  
  "transaction":  
    "7627108136485906579",  
  "broadcasted":  
    true,  
  "unsignedTransactionBytes":  
    "0105c84ad700a005668683e673274c671802492d6f54cdc33d247aa4..."  
}
```

Проверено 14/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Действия с Альясами

Создание/присвоение альяса

Позволяет вам создать альяс в один запрос. Если данный альяс уже существует , этот запрос присвоит новый URI псевдониму.

Запрос

```
http://localhost:1400/api?
  requestType=assignAlias&
  secretPhrase=SECRET&
  alias=ALIAS&
  uri=URI&
  fee=FEE&
  deadline=DEADLINE&
  publicKey=PUBKEY&
  referencedTransactionFullHash=TRANSFULLHASH&
  broadcast=BROAD
```

Где:

- SECRET секретная фраза аккаунта
- ALIAS альяс, который вы хотите создать
- URI текст альяса (на пример <http://www.google.com/>)
- FEE комиссия (в MilliLm) за транзакцию
- DEADLINE дедлайн для транзакции, задается в минутах
- PUBKEY 64-битный публичный ключ ассоциированный с данным аккаунтом. (опционально)
- TRANSFULLHASH Если Вы хотите сослаться на другую транзакцию(опционально)
- BROAD true или false если это должно быть опубликовано широкоэщательно (опционально)

'Ответ'

```
{
  "transactionId":      "TXID"
}
{
  "fullHash":           "FULLHASH",
  "signatureHash":      "SIGHASH",
  "transactionBytes":   "TXBYTES",
  "hash":               "HASH",
  "transaction":        "TX",
```

```
    "broadcasted":          BROAD,  
    "unsignedTransactionBytes": "UNTXBYTES"  
}
```

Проверено 15/05/14

Пример

Запрос:

```
http://localhost:1400/api?  
  requestType=setAlias&  
  secretPhrase=123&  
  aliasName=Google&  
  aliasURI=http://google.com&  
  fee=10000000000&  
  deadline=1440
```

Результат:

```
{  
  "fullHash":  
    "645853ddc55daa2c948f46520b0630d964ce0f40c0ee03515faffce6259d0bae",  
  "signatureHash":  
    "1e622a64292f8b1fcb1ccf155a42a37a99b89cbcaa725f1200ab55b5ec5f1ca8",  
  "transactionBytes":  
    "0101fe21d700a005668683e673274c671802492d6f54cdc33d247aa426e7ce...",  
  "hash":  
    "580c1bb70d7bcb1eb967c4e86c6c9bfc481641dcce84300a579caf35840e662a",  
  "transaction":          "3218487988112808036",  
  "broadcasted":          true,  
  "unsignedTransactionBytes":  
    "0101fe21d700a005668683e673274c671802492d6f54cdc33d247a..."  
}
```

Исправлено 22/05/14

Get alias

Возвращает информацию о заданном альясе

Запрос

```
https://localhost:1400/api?  
  requestType=getAlias&  
  alias=ALIAS_ID
```

Где:

- ALIAS_ID альяс который вас интересует
- ALIAS_NAME имя альяса который вас интересует

Ответ

```
{  
  "timestamp":    TIMESTAMP,  
  "aliasName":    "ALIAS",  
  "account":      "ACCOUNT",  
  "accountRS":    "ACCOUNTRS",  
  "aliasURI":     "URI",
```

```
}    "alias":      "ALIAS_ID"
```

Где:

- TIMESTAMP метка времени, когда был создан альяс. Задается в секундах, от даты создания генезисного блока
- ALIAS название альяса
- ACCOUNT аккаунт, который владеет альсом
- ACCONTRRS адрес аккаунта в кодировке Reed-Solomon
- URI адрес, на который указывает псевдоним
- ALIAS_ID идентификатор альяса (id)

Пример

Запрос:

```
https://localhost:1400/api?  
requestType=getAlias&  
alias=15515279700680480368
```

Ответ:

```
{  
  "timestamp": 2409343,  
  "alias":     "google",  
  "account":   "5629477397208681336",  
  "accountRS": "LMA-FLVS-VRBV-LDPD-6DZ9W",  
  "uri":       "http://google.com"  
}
```

Проверено 15/05/14

Get aliases

Возвращает информацию о всех альясах для указанного аккаунта (начиная с указанной метки времени, если задано)

Запрос

```
https://localhost:1400/api?  
requestType=getAliases&  
account=ACCOUNT&  
timestamp=TIMESTAMP
```

Где:

- ACCOUNT аккаунт, список альясов которого вас интересует
- TIMESTAMP время начиная с которого были созданы альясы. Задается в секундах начиная с времени создания генезисного блока.

Ответ

```
{
  "aliases": [
    {
      "timestamp":  TIMESTAMP,
      "aliasName":   "ALIAS",
      "account":     "ACCOUNT",
      "accountRS":   "ACCOUNTRS",
      "aliasURI":    "URI",
      "alias":       "ALIAS_ID"
    },
    {
      ...
    }
  ]
}
```

Где:

- TIMESTAMP метка времени, когда был создан альяс. Задается в секундах, от даты создания генезисного блока
- ALIAS название альяса
- ACCOUNT аккаунт, который владеет альсом
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon
- URI адрес, на который указывает псевдоним
- ALIAS_ID идентификатор альяса (id)

Пример

Запрос:

```
https://localhost:1400/api?
  requestType=getAliases&
  account=5629477397208681336
```

Ответ:

```
{
  "aliases": [
    {
      "timestamp": 2409602,
      "aliasName": "promotion",
      "alias": "11086300057073013586",
      "aliasURI": "",
      "accountRS": "LMA-FLVS-VRBV-LDPD-6DZ9W",
      "account": "5629477397208681336"
    },
    {
      ...
    }
  ]
}
```

Проверено 26/05/14

Get alias ID (Не используется)

Возвращает ID определенного алиаса. Реализовано начиная с версии 0.5.4e.
Функция не используется с версии клиента 1.1.4.

Запрос

```
https://localhost:1400/api?  
  requestType=getAliasId&  
  alias=ALIAS
```

Где:

- ALIAS — альяс, ID которого вы хотите получить

Ответ

```
{  
  "id":      "ALIAS_ID"  
}
```

'Пример'

Запрос:

```
https://localhost:1400/api?  
  requestType=getAliasId&  
  alias=google
```

Ответ:

```
{  
  "id":      "15515279700680480368"  
}
```

Проверено 15/05/14

Get alias IDs (Не используется)

Возвращает все ID транзакций алиасов после определенного времени в блокчейне. Функция не используется, начиная с клиента версии 1.1.4

Запрос

```
https://localhost:1400/api?  
  requestType=getAliasIds&  
  timestamp=TIMESTAMP
```

Где:

- TIMESTAMP — время, начиная с которого вы хотели бы видеть определенные алиасы. Выражается в секундах начиная с времени генезис блока

'Ответ'

```
{
  "aliasIds":      [ARRAY_OF_ALIAS_IDS]
}
```

Проверено 14/05/14

Пример

Запрос:

```
https://localhost:1400/api?
  requestType=getAliasIds&
  timestamp=30000
```

Результат:

```
{
  "aliasIds":
    ["12219820415480193342",
     "2284413548488744428",
     "6882945328050210195"...]
}
```

Исправлено 22/05/14

Get alias URI (Не используемый)

Возвращает целевой URI альяса. Функция не используется, начиная с клиента версии 1.1.4

Запрос

```
http://localhost:1400/api?
  requestType=getAliasURI&
  alias=ALIAS
```

Где:

- ALIAS альяс, целевой URI которого Вы хотели бы видеть

'Ответ'

```
{
  "uri":      "URI_ASSIGNED_TO_ALIAS"
}
```

Внимание: Возвращенный URI будет ограничен! *Проверено 14/05/14*

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAliasURI&
  alias=Google
```

Ответ:

```
{
  "uri": "http:\\\\google.com"
}
```

Исправлено 22/05/14

List Aliases (Не используется)

Возвращает список алиасов, принадлежащих аккаунту. Функция не используется начиная с клиента версии 1.1.4

Запрос

```
http://localhost:1400/api?
  requestType=listAccountAliases&
  account=ACCOUNT
```

Где:

- ACCOUNT — номер аккаунта

'Ответ'

```
{
  "aliases":
    [
      {
        "alias": "ALIAS",
        "uri": "URI",
        "account": "ACCOUNT",
        "timestamp": "TIMESTAMP"
      }
    ]
}
```

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=listAccountAliases&
  account=17157779854388545959
```

Ответ:

```
{
  "aliases":
    [
      {
        "alias": "testalias",
        "uri": "1337",
        "account": "17157779854388545959",
        "timestamp": 2920016
      }
    ]
}
```

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Системные функции для работы с произвольными сообщениями

Send Message

Создает транзакцию с исходящим произвольным сообщением.

Запрос

```
http://localhost:1400/api?
  requestType=sendMessage&
  secretPhrase=SECRET&
  recipient=RECIP_ACCOUNT&
  fee=FEE&
  message=HEX_STRING&
  deadline=DEADLINE&
  referencedTransaction=REFTXID&
  broadcast=BROAD
```

Где:

- SECRET секретная фраза аккаунта, который отправляет сообщение
- RECIP_ACCOUNT номер аккаунта, которому отправляется сообщение (получателя)
- HEX_STRING сообщение в hex кодировке. Механизмы шифрования сообщения не предусмотрены, оставлено для разработчиков. Длина сообщения ограничена 1000 байтами.
- FEE комиссия (в MilliLm) за транзакцию
- DEADLINE дедлайн транзакции, указывается в минутах
- REFTXID ID предыдущей транзакции на которую вы ссылаетесь. Это создает зависимые транзакции (цепочка транзакций), означая, что текущая транзакция не может быть подтверждена, пока транзакция на которую вы ссылаетесь, не будет подтверждена. (опционально)
- BROAD если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

Ответ

```
{
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "transactionBytes": "TXBYTES",
  "hash": "HASH",
  "transaction": "TX",
  "broadcasted": BROAD,
```

```
}    "unsignedTransactionBytes": "UNTXBYTES"
```

Где:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хэш всех полей транзакции, за исключение подписи
- TX ID транзакции
- BROADCAST указывает была транзакция передана или нет
- UNTXBYTES байты транзакции с подписью, частично замененной нулями

Проверено 15/05/14

Пример

Запрос:

```
http://localhost:1400/api?
requestType=sendMessage&
secretPhrase=MySuperSecretPasswordThatYouDoNotKnow&
recipient=501860284368&
message= 6602b1611f921092bdf2a89fc97148ec9684066776b83fd3&
deadline=720&
referencedTransaction=8815617645011985536
```

Ответ:

```
{
  "fullHash":
"d56ade2c322083d76ebcdb9f9ef18d7d52c6d105f67be040cc423c7a41a47aea",
  "signatureHash":
"458514e72f4272c789f99201c27cff611460de10fe484b474166c1199c23e754",
  "transactionBytes":
"0100471ddb00a005668683e673274c671802492d6f54cdc33d247aa426e7cef...",
  "hash":
"36a0dc90736d95ad5302f078d5db6fc48e5c087b76da65944f7fe2836b73abcd",
  "transaction":
"15529291339976829653",
  "broadcasted":
true,
  "unsignedTransactionBytes":
"0100471ddb00a005668683e673274c671802492d6f54cdc33d247aa4..."
}
```

Проверено 13/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции для работы с блоками

Get Block

Получить данные блока.

Запрос

```
http://localhost:1400/api?  
requestType=getBlock&  
block=BLOCKADDRESS
```

Где:

- BLOCKADDRESS адрес блока LibreMoney

Ответ

```
{  
  "generatorRS": "ACCOUNTRS",  
  "previousBlockHash": "32BytesRepresentedInHexadecimalForm",  
  "payloadLength": PAYLOADLEN,  
  "totalAmount": TOTAL,  
  "generationSignature": "64BytesRepresentedInHexadecimalForm",  
  "generator": "ACCOUNT",  
  "baseTarget": "BASETARGET",  
  "payloadHash": "32BytesRepresentedInHexadecimalForm",  
  "nextBlock": "NEXTBLOCK",  
  "numberOfTransactions": NUMTX,  
  "blockSignature": "64BytesRepresentedInHexadecimalForm",  
  "transactions":  
    [  
      "TRANSACTION1",  
      "TRANSACTION2"  
    ],  
  "version": VERSION,  
  "totalFee": "FEE",  
  "previousBlock": "PREVBLOCK",  
  "height": HEIGHT,  
  "timestamp": TIME  
}
```

Где:

- HEIGHT высота блока
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon
- "generator" аккаунт который сгенерировал блок.
- ACCOUNT номер аккаунта.
- TIME время в секундах когда блок был сгенерирован; 0 время создания

генезисного блока.

- NUMTX количество транзакций в этом блоке
- TOTAL общее количество MilliLm в данном блоке
- FEE комиссия MilliLm для данного блока
- VERSION версия блока
- BASETARGET целевой hash для генерации блока
- Payload length = "numberOfTransactions" * 128 Bytes.
- "previousBlock" адрес предыдущего блока. Параметр будет пустым для генезисного блока.
- PREVBLOCK адрес предыдущего блока
- "nextBlock" адрес следующего блока за текущим. Параметр будет пустым для последнего блока в blockchain.
- NEXTBLOCK адрес следующего блока, за указанным
- "transactions" список/массив идентификаторов (ID) транзакций представленных в этом блоке.
- TRANSACTIONX идентификаторы (ID) транзакций в блоке

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getBlock&
  block=3081264592726137683
```

Ответ:

```
{
  "generatorRS": "LMA-25TQ-G377-8TJ9-EVPAD",

  "previousBlockHash": "d2df42de82c3fbc808715ce9738630b31bece282864fafd164b0d4cf5b0
e9e3e",
  "payloadLength": 256,
  "totalAmount": "164003000000000",

  "generationSignature": "5beac191d5a20b443b9fdd83cf3abddfb31294c1999a85c75955a4c
99b3f3c",
  "generator": "14241151062656421686",
  "baseTarget": "191236740",

  "payloadHash": "fc153c1f3d18af9e5a60be42bee896b19089a4b1e391179c98c2a9d2d383275f"
,
  "nextBlock": "5728643978405745233",
  "numberOfTransactions": 2,

  "blockSignature": "5bf8d6f39c6303815dbf591667d613175e916520052f21ffaddec818e2e4ab
0cca07f17b0dea8352e837aa89e774cd95dbeb5a2bdab1f1f50f5955d1a3718375",
  "transactions":
    [
      "10431005438877942947",
```

```
        "16244324356084723003"  
    ],  
    "version":2,  
    "totalFee":"2000000000",  
    "previousBlock":"14482383993582051282",  
    "height":115147,  
    "timestamp":12439666  
}
```

Проверено 15/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции на Бирже Активов

Обратите внимание - единицей биржевой операции является АКТИВ. Активом может быть что угодно - другая криптовалюта, акции и т.п.

Issue asset

Создание актива для биржи

Запрос

```
http://localhost:1400/api?
  requestType=issueAsset&
  secretPhrase=SECRET&
  name=ASSETNAME&
  description=DESCRIPTION&
  quantityQNT=QTY&
  decimals=DECIMALS&
  fee=FEE&
  deadline=DEADLINE&
  referencedTransaction=REFTX&
  broadcast=BROAD
```

Where:

- SECRET секретная фраза аккаунта создающего актив.
- ASSETNAME имя актива, который вы создаете для продажи на бирже
- DESCRIPTION описание актива. Этот текст должен быть URL-encoded, но может включать любые Unicode символы, и иметь максимальную длину 1000 символов. (опционально)
- QTY количество единиц актива
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- FEE комиссия (в MilliLm) за транзакцию по созданию актива. На данный момент минимальная комиссия за создание активасоставляет 1000 MilliLm
- DEADLINE дедлайн до подтверждения транзакции, задается в минутах
- REFTXID ID предыдущей транзакции на которую вы ссылаетесь. Это создает зависимые транзакции (цепочка транзакций), означая, что текущая транзакция не может быть подтверждена, пока транзакция на

которую вы ссылаетесь, не будет подтверждена. (опционально)

•BROADCAST если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

Проверено 15/05/14

Ответ

В ответе возвращаются идентификаторы транзакции (ID), который является также ID актива

```
{
  "fullHash":          "FULLHASH",
  "signatureHash":     "SIGHASH",
  "transactionBytes":  "TXBYTES",
  "hash":              "HASH",
  "transaction":       "TX",
  "broadcasted":       BROADCAST,
  "unsignedTransactionBytes": "UNTXBYTES"
}
```

Where:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Полезно для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH шэх всех полей транзакции за исключением подписи
- TX идентификатор (ID) транзакции
- BROADCAST указывает была транзакция передана или нет
- UNTXBYTES байты транзакции с подписью, частично замененной нулями

Исправлено 22/05/14

Пример

Запрос:

```
http://localhost:1400/api?
requestType=issueAsset&
secretPhrase=123&
name=turtleCoin&
description=This+is+Turtlecoin+issued+by+Tommy-Turtle&
quantityQNT=500&
deadline=1440&
fee=1000
```

Ответ:

```
{
  "fullHash":
  "ebccb29dc4286a4a55ad9f0a4334cd3b0df75c26fc4a6ee3b62832f8f1a403",
```

```
    "signatureHash":
"adb63682ac9598b7ea951df09535ee662b4b2f977e4f8a3f1a1aa0fd1f3e3da1",
    "transactionBytes":
"0200701fdb00a005668683e673274c671802492d6f54cdc33d247aa426e7cef83776...",
    "hash":
"f1fa3f7b8ca5dd38d529e9f9a052aac0f96a7e5a4ef57cdabb0e4633cb4fcd89",
    "transaction":
"5362143131262438635",
    "broadcasted":
true,
    "unsignedTransactionBytes":
"0200701fdb00a005668683e673274c671802492d6f54cdc33d247aa426e7ce..."
}
```

Проверено 13/06/14

Get asset IDs

Получить список всех идентификаторов актива на бирже

Запрос

```
http://localhost:1400/api?
requestType=getAssetIds
```

'Результат'

```
{
    "assetIds":      [ARRAY_OF_ASSET_IDS]
}
```

Где ARRAY_OF_ASSET_IDS - массив/список идентификаторов актива

'Пример'

Запрос:

```
http://localhost:1400/api?
requestType=getAssetIds
```

Результат:

```
{
    "assetIds":
    [
        "923860986719836",
        "46556984592384756298",
        "87918658237693184156",
        ...
    ]
}
```

Проверено 15/05/14

Get asset

Позволяет получить информацию ассоциированную с ID актива

Request

```
http://localhost:1400/api?
requestType=getAsset&
```

asset=ASSETID

Где:

- ASSETID ID актива, который вас интересует

Response

```
{
  "account":          "ACCOUNT",
  "accountRS":        "ACCOUNTRS",
  "name":             "NAME",
  "description":       "DESCRIPTION",
  "quantityQNT":      QTY,
  "asset":            ASSETID,
  "decimals":          DECIMALS,
  "numberOfTrades":   NUMTRADES
}
```

Где:

- ACCOUNT аккаунт которому принадлежит актив
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- NAME имя/название актива
- DESCRIPTION описание актива. Данная запись может не возвращаться, если описание отсутствует.
- QTY количество единиц актива
- ASSETID ID актива который вас интересует
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- NUMTRADES число сделок с этим активом

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getAsset&
asset=46556984592384756298
```

Ответ:

```
{
  "account":          "334765832685676",
  "accountRS":        "LMA-DE2F-W76R-GL25-HMFPR",
  "name":             "cfbBTCx1",
  "description":       "This is BTC issued by Come-from-Beyond",
  "quantityQNT":      500,
  "asset":            46556984592384756298,
  "decimals":          6,
  "numberOfTrades":   0
}
```

Проверено 15/05/14

Get assets

Получить информацию, связанную с несколькими идентификаторами активов

Запрос

```
http://localhost:1400/api?
requestType=getAssets&
assets=ASSETID&
assets=ASSETID&
...
```

Где:

- ASSETID ID актива, который вас интересует

Запрос

```
{
  "assets": [ {
    "account":      "ACCOUNT",
    "accountRS":    "ACCOUNTRS",
    "name":         "NAME",
    "description":  "DESCRIPTION",
    "quantityQNT":  QTY,
    "asset":        ASSETID,
    "decimals":     DECIMALS,
    "numberOfTrades": NUMTRADES
  },
  {
    ...
  }
]
```

Где:

- ACCOUNT аккаунт которому принадлежит актив
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- NAME имя/название актива
- DESCRIPTION описание актива. Данная запись может не возвращаться, если описание отсутствует.
- QTY количество единиц актива
- ASSETID ID актива который вас интересует
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- NUMTRADES число сделок с этим активом

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getAssets&
assets=46556984592384756298
```

Ответ:

```
{
  "assets": [ {
    "account": "334765832685676",
    "accountRS": "LMA-DE2F-W76R-GL25-HMFPR",
    "name": "cfbBTCx1",
    "description": "This is BTC issued by Come-from-Beyond",
    "quantityQNT": 500,
    "asset": 46556984592384756298,
    "decimals": 6,
    "numberOfTrades": 0
  } ]
}
```

Проверено 15/05/14

Get all assets

Предоставляет информацию по всем активам представленным на бирже

Запрос

```
http://localhost:1400/api?
requestType=getAllAssets
```

Ответ

```
{
  "assets":
  [ {
    "account": "ACCOUNT",
    "accountRS": "ACCOUNTRS"
    "name": "NAME",
    "description": "DESCRIPTION",
    "quantityQNT": QTY,
    "asset": ASSETID,
    "decimals": DECIMALS,
    "numberOfTrades": NUMTRADES
  } ]
}
```

Где:

- ACCOUNT аккаунт которому принадлежит актив
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- NAME имя/название актива
- DESCRIPTION описание актива. Данная запись может не возвращаться, если описание отсутствует.

- QTY количество единиц актива
- ASSETID ID актива который вас интересует
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- NUMTRADES число сделок с этим активом

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getAllAssets
```

Ответ:

```
{
  "assets":
  [ {
    "decimals":      7,
    "numberOfTrades": 0,
    "asset":         "3283533989023503499",
    "quantityQNT":   "1322495000000000",
    "description":   "CCGGUQMJKOMUCLBIEGGTKCQVSBSJAZSE",
    "name":          "GZSHFQY",
    "account":       "14136559604731496960",
    "accountRS":     "LMA-KRDB-ZFA3-RNDV-AALNT"
  },
  {
    "decimals":      6,
    "numberOfTrades": 0,
    "asset":         "5008310702202713285",
    "quantityQNT":   "9945120000000000",
    "description":   "QESZUYEJCGBGHVXIRC",
    "name":          "GWNW",
    "account":       "14136559604731496960",
    "accountRS":     "LMA-UWKJ-GFEV-AGY4-5C4YS"
  },
  ...
  ]
}
```

Проверено 15/05/14

Get assets by name (Не используется)

Предоставляет информацию ассоциированную с именем актива. Функция не используется начиная с версии клиента 1.1.4.

Запрос

```
http://localhost:1400/api?
requestType=getAssetsByName&
assetName=ASSET
```

Где:

- ASSET название актива, который вас интересует

Ответ

```
{
  "assets":
  [ {
    "account":          "ACCOUNT",
    "accountRS":        "ACCOUNTRS"
    "name":             "NAME",
    "description":      "DESCRIPTION",
    "quantityQNT":      QTY,
    "asset":            ASSETID,
    "decimals":         DECIMALS,
    "numberOfTrades":   NUMTRADES
  } ]
}
```

Where:

- ACCOUNT аккаунт которому принадлежит актив
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- NAME имя/название актива
- DESCRIPTION описание актива. Данная запись может не возвращаться, если описание отсутствует.
- QTY количество единиц актива
- ASSETID ID актива который вас интересует
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- NUMTRADES число сделок с этим активом

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getAssetsByName&
assetName=GZSHFQY
```

Ответ:

```
{
  "assets":
  [ {
    "decimals":          7,
    "numberOfTrades":    0,
    "asset":             "3283533989023503499",
    "quantityQNT":       "1322495000000000",
    "description":       "CCGGUQMJKOMUCLBIEGGTKCQVSBSJAZSE",
    "name":              "GZSHFQY",
    "account":           "14136559604731496960",
    "accountRS":         "LMA-UWKJ-GFEV-AGY4-5C4YS"
```

```
} ]  
}
```

Проверено 15/05/14

Get assets by issuer

Возвращает все активы, которые создал указанный аккаунт

Запрос

```
http://localhost:1400/api?  
requestType=getAssetsByIssuer&  
account=ACCOUNT
```

Где:

- ACCOUNT аккаунт который выпустил актив

Ответ

```
{  
  "assets":  
  [ {  
    "account": "ACCOUNT",  
    "accountRS": "ACCOUNTRS",  
    "name": "NAME",  
    "description": "DESCRIPTION",  
    "quantityQNT": QTY,  
    "asset": ASSETID,  
    "decimals": DECIMALS,  
    "numberOfTrades": NUMTRADES  
  } ]  
}
```

Где:

- ACCOUNT аккаунт которому принадлежит актив
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- NAME имя/название актива
- DESCRIPTION описание актива. Данная запись может не возвращаться, если описание отсутствует.
- QTY количество единиц актива
- ASSETID ID актива который вас интересует
- DECIMALS количество знаков после запятой для указанного актива. Для примера, для USD это будет 2 (т.е. 1.87 USD), для Bitcoin это будет 8 (т.е. 0.00000025 BTC)
- NUMTRADES число сделок с этим активом

Пример

Запрос:


```
http://localhost:1400/api?
requestType=getAssetsByIssuer&
account=1562462127635514638
```

Ответ:

```
{
  "assets":
  [ {
    "decimals": 7,
    "numberOfTrades": 0,
    "asset": "3283533989023503499",
    "quantityQNT": "1322495000000000",
    "description": "CCGGUQMJKOMUCLBIEGGTKCQVSBSJAZSE",
    "name": "GZSHFQY",
    "accountRS": "LMA-GKAG-2VGD-U473-3VRDD",
    "account": "1562462127635514638"
  } ]
}
```

Проверено 15/05/14

Get all trades

Возвращает все сделки, начиная с определенного времени

Запрос

```
http://localhost:1400/api?
requestType=getAllTrades&
timestamp=TIME
```

Где:

- TIME задается в секундах, от времени создания генезисного блока.
- Установка параметра в 0 вернет все сделки (опционально)

Ответ

```
{
  "trades": [
    {
      "timestamp": TIME,
      "price": "PRICE",
      "asset": "ASSETID",
      "quantityQNT": "QTY",
      "block": "BLOCK",
      "askOrder": "ASK",
      "bidOrder": "BID"
    },
    {
      ...
    }
  ]
}
```

Где:

- TIME время заданное в секундах от момента создания генезисного

блока

- ASSETID ID актива
- QTY количество ордеров
- PRICE цена, выраженная в MilliLm
- BLOCK блок сделки
- ASK ID ордера на покупку
- BID ID ордера на продажу

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=getAllTrades
```

Ответ:

```
{  
  "trades": [  
    {  
      "timestamp": 13530463,  
      "price": "25000000000",  
      "asset": "13920592454636198263",  
      "quantityQNT": "2",  
      "block": "1942829179466651543",  
      "askOrder": "16791208091984728819",  
      "bidOrder": "12730658526877674075"  
    },  
    {  
      ...  
    }  
  ]  
}
```

Проверено 15/05/14

Get trades

Получить все торговые операции по активу с указанным ID

Запрос

```
http://localhost:1400/api?  
requestType=getTrades&  
asset=ASSETID&  
firstIndex=RANGE_START  
lastIndex=RANGE_END
```

Где:

- ASSETID ID актива который вас интересует
- RANGE_START порядковый номер самой ранней торговой операции, которую вы хотите увидеть. Если не указано, то по умолчанию 0

•RANGE_END порядковый номер последней торговой операции которую вы хотите видеть. Если не указано, то по умолчанию устанавливается $2^{32}-1$

Ответ

```
{
  "trades": [
    {
      "timestamp":    TIME,
      "price":        "PRICE",
      "asset":        "ASSETID",
      "quantityQNT":  "QTY",
      "block":        "BLOCK",
      "askOrder":     "ASK",
      "bidOrder":     "BID"
    },
    {
      ...
    }
  ]
}
```

Где:

- TIME is the time expressed in seconds since the genesis block of the trade
- ASSETID ID актива
- QTY количество ордеров
- PRICE цена, указывается в MilliLm
- BLOCK блок сделки
- ASK ID ордера на покупку
- BID ID ордера на продажу

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getTrades&
asset=14269709746849295412&
firstIndex=0
```

Ответ:

```
{
  "trades": [
    {
      "timestamp":    13386268,
      "price":        "1000000000",
      "asset":        "13388701969217905199",
      "quantityQNT":  "12",
      "block":        "15363991879367428772",
      "askOrder":     "9028610672087970547",
      "bidOrder":     "7785235474459518056"
    }
  ]
}
```

```

    },
    {
        ...
    }
]
}

```

Проверено 15/05/14

Transfer asset

Функция для передачи определенного количества актива с одного аккаунта на другой

Запрос

```

http://localhost:1400/api?
requestType=transferAsset&
secretPhrase=SECRET&
recipient=ACCOUNT&
asset=ASSETID&
comment=COMMENT&
quantityQNT=QTY&
fee=FEE&
deadline=DEADLINE&
referencedTransaction=REFTXID&
broadcast=BROAD

```

Где:

- SECRET секретная фраза аккаунта который осуществляет передачу актива
- RECIPIENT аккаунт получателя
- ASSETID идентификатор (ID) актива который будет передаваться
- QTY количество единиц актива которое будет передаваться
- COMMENT комментарий к перемещению (опционально)
- FEE объем комиссии за эту транзакция, в MilliLm
- DEADLINE время в минутах, в течении которого транзакция должна быть подтверждена в сети.
- REFTX Идентификатор (ID) предыдущей транзакции на которую ссылается данная транзакция. Это значит, что создается цепочка зависимых транзакций, и текущая транзакция не может быть подтверждена, пока транзакция, на которую ссылаются, не будет подтверждена (опционально)
- BROAD если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

Ответ

```
{
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "transactionBytes": "TXBYTES",
  "hash": "HASH",
  "transaction": "TX",
  "broadcasted": BROAD,
  "unsignedTransactionBytes": "UNTXBYTES"
}
```

Где

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хэш всех полей транзакции, за исключение подписи
- TX ID транзакции
- BROAD указывает была транзакция передана или нет
- UNTXBYTES байты транзакции с подписью, частично замененной нулями

Пример

Запрос:

```
http://localhost:1400/api?
requestType=transferAsset&
secretPhrase=123&
recipient=81623458239475&
asset=46556984592384756298&
quantityQNT=45&
fee=1000000000&
deadline=1440
```

Ответ:

```
{
  "fullHash":
"9bd14e7a93d228a85e82a98071f1bd96c158dda856311722e6feafb1a36e71e2",
  "signatureHash":
"540a2963aac95e63c5e4d13b452be02160e5ecd9cb1c3e8306eed4b47975b9ba",
  "transactionBytes":
"0201f120db00a005668683e673274c671802492d6f54cdc33d247a...",
  "hash":
"8633ca65458bb9895ef53c3875ff59dd4b85ddf922dcaae3f276796932d33731",
  "transaction": "12117166328294330779",
  "broadcasted": true,
  "unsignedTransactionBytes":
"0201f120db00a005668683e673274c671802492d6f54cdc33d2..."
}
```

Place order

Размещает на бирже заявку на "покупку" или "продажу" актива

Запрос

```
http://localhost:1400/api?
  requestType=BID_OR_ASK&
  secretPhrase=SECRET&
  asset=ASSETID&
  quantityQNT=QTY&
  price=PRICE&
  fee=FEE&
  deadline=DEADLINE&
  referencedTransaction=REFTXID&
  broadcast=BROAD
```

Где:

- BID_OR_ASK тип размещаемой заявки "placeAskOrder" (купить) or "placeBidOrder"(продать)
- SECRET секретная фраза от аккаунта, который создает транзакцию
- ASSETID идентификатор (ID) актива на который формируется запрос
- QTY количество единиц актива на который формируется запрос
- PRICE цена покупки/продажи, измеряется в MilliLm (т.е. "1456" это "1.456 Lm")
- FEE комиссия за эту транзакцию, в MilliLm
- DEADLINE время в минутах, в течении которого данная транзакция должна быть подтверждена в сети.
- REFTX Идентификатор (ID)предыдущей транзакции на которую ссылается данная транзакция. Это значит, что создается цепочка зависимых транзакций, и текущая транзакция не может быть подтверждена, пока транзакция, на которую ссылаются, не будет подтверждена (опционально)
- BROAD если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

'Ответ'

```
{
  "fullHash":          "FULLHASH",
  "signatureHash":     "SIGHASH",
  "transactionBytes":  "TXBYTES",
  "hash":              "HASH",
  "transaction":       "TX",
  "broadcasted":       BROAD,
  "unsignedTransactionBytes": "UNTXBYTES"
}
```

Где:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хеш всех полей транзакции, за исключением подписи
- TX ID транзакции
- BROADCAST статус - транзакция была передана или нет
- UNTXBYTES байты транзакции с частью подписи, заменяемой нулями

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=placeBidOrder&
  secretPhrase=123&
  asset=46556984592384756298&
  quantityQNT=70&
  price=145&
  fee=1000000000&
  deadline=1440&
  referencedTransaction=65374835678
```

Ответ:

```
{
  "fullHash":
"a797ff58c5ed5c6f53c708dbe137812495c6af0eb280a3d001bb1917eb6a5561",
  "signatureHash":
"751c9df738e1894ef550bf9e0725188168d4febf48a4da51b1efafaaa840dbcb",
  "transactionBytes":
"02035d0fdb00a005668683e673274c671802492d6f54cdc33d247aa426e7cef83776...",
  "hash":
"4785247b3238fda4fe53d47bc8a97a2b3acf974a02854792b59b092f3473eecd",
  "transaction":
"8024550067924866983",
  "broadcasted":
true,
  "unsignedTransactionBytes":
"02035d0fdb00a005668683e673274c671802492d6f54cdc33d247aa426e7cef..."
}
```

Проверено 15/05/14

Get order IDs

Получить идентификаторы (ID) заявок на покупку/продажу определенного актива

Запрос

```
http://localhost:1400/api?
  requestType=GET_BID_OR_ASK&
  asset=ASSETID
```

Где:

- GET_BID_OR_ASK тип операции "getAskOrderIds" или "getBidOrderIds"
- ASSETID ID актива, ордера по которому вы желаете получить

'Результат'

Возвращает массив (список) ордеров покупки/продажи:

```
{
  "GET_BID_OR_ASK": [ARRAY_OF_ORDER_IDS]
}
```

Где:

- GET_BID_OR_ASK тип операции "getAskOrderIds" или "getBidOrderIds"

'Пример'

Запрос:

```
http://localhost:1400/api?
  requestType=getAskOrderIds
  asset=14269709746849295412
```

Ответ:

```
{
  "askOrderIds":
    [
      "3467356724512",
      "824756987236452"
    ]
}
```

Проверено 16/05/14

Get orders

Возвращает все ордера на покупку/продажу, по указанному активу

Запрос

```
http://localhost:1400/api?
  requestType=GET_BID_OR_ASK&
  asset=ASSETID&
  limit=LIMIT
```

Где:

- GET_BID_OR_ASK может быть "getAskOrders" или "getAskOrders"
- ASSETID идентификатор (ID) актива, информация по которому вас интересует
- LIMIT определяет сколько ордеров необходимо отобразить, если не указано, должен отобразить их все.

Проверено 18/05/14

Ответ

Возвращает массив ордеров покупки/продажи:

```
{
  "GET_BID_OR_ASK": [ARRAY_OF_ORDERS]
}
```

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAskOrders
  asset=15680163033397272955
```

Ответ:

```
{
  "askOrders": [
    {
      "height": 90703,
      "price": "78932342",
      "asset": "15680163033397272955",
      "order": "4529944714996630823",
      "quantityQNT": "20",
      "accountRS": "LMA-JNYK-TP3W-VDBB-9FRVB",
      "account": "8425933418718057425",
      "type": "ask"
    },
    {
      "height": 77415,
      "price": "990000000",
      "asset": "15680163033397272955",
      "order": "3158651792442050495",
      "quantityQNT": "39",
      "accountRS": "LMA-AMPQ-B6ZZ-S8TL-EFDBH",
      "account": "14386024746077933238",
      "type": "ask"
    },
    {
      "height": 78822,
      "price": "1500000000",
      "asset": "15680163033397272955",
      "order": "954178876942447633",
      "quantityQNT": "1",
      "accountRS": "LMA-FVXE-SLB6-KL5V-A2Q75",
      "account": "9337862827899547564",
      "type": "ask"
    }
  ]
}
```

Исправлено 22/05/14

Get current account order IDs

Возвращает идентификаторы (ID) ордеров на покупку или продажу для указанного аккаунта, опционально данные фильтруются согласно заданному

идентификатору актива (assetID). Реализовано начиная с версии 0.5.9.

Запрос

```
http://localhost:1400/api?  
requestType=GET_BID_OR_ASK&  
account=ACCOUNT&  
asset=ASSETID
```

Где:

- GET_BID_OR_ASK может быть "getAccountCurrentAskOrderIds" или "getAccountCurrentBidOrderIds"
- ACCOUNT номер аккаунта ордера которого вы хотите получить.
- ASSETID идентификатор (ID) актива ордера с которым вы хотите получить.

Ответ

Возвращает массив из всех идентификаторов (ID) запросов/предложений:

```
{  
  "GET_BID_OR_ASK_IDS":  
    [ARRAY_OF_ORDER_IDS]  
}
```

Где:

- GET_BID_OR_ASK_IDS может быть "askOrderIds" или "bidOrderIds"

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=getAccountCurrentAskOrderIds&  
account=1189524830352486657&  
asset=2438729719254516952
```

ОТВЕТ:

```
{  
  "askOrderIds":  
    [  
      "3467356725412",  
      "824756987456452"  
    ]  
}
```

Get order

Возвращает полную информацию о указанном ордере на покупку/продажу

Запрос

```
http://localhost:1400/api?
```

```
requestType=GET_BID_OR_ASK&
order=ORDERID
```

Где:

- GET_BID_OR_ASK может быть "getAskOrder" или "getBidOrder"
- ORDERID Идентификатор (ID) ордера, который вас интересует

Ответ

```
{
  "account":      "ACCOUNT",
  "accountRS":    "ACCOUNTRS",
  "height":       HEIGHT,
  "asset":        "ASSETID",
  "quantityQNT":  QTY,
  "price":        PRICE,
  "order":        "ORDER",
  "type":         "TYPE"
}
```

Где:

- ACCOUNT номер аккаунта ассоциированный с ордером
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon.
- ASSETID идентификатор (ID) актива в ордере
- QTY количество актива в ордере
- PRICE цена, указана в MilliLm
- HEIGHT высота блока
- ORDER номер ордера
- TYPE тип ордера (покупка/продажа)

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getBidOrder&
order=17936094171153028796
```

Ответ:

```
{
  "height":      90703,
  "price":       "78932342",
  "asset":       "15680163033397272955",
  "order":       "17936094171153028796",
  "quantityQNT": "20",
  "account":     "8425933418718057425",
  "accountRS":   "LMA-7HL4-WMNR-XGFL-ASVHV",
  "type":        "ask"
}
```

Проверено 15/05/14

Get all open orders

Возвращает информацию о всех открытых ордерах

Запрос

```
http://localhost:1400/api?  
requestType=getAllOpenOrders
```

Ответ

```
{  
  "height":      HEIGHT,  
  "account":     "ACCOUNT",  
  "accountRS":   "ACCOUNTRS",  
  "asset":       "ASSETID",  
  "order":       "ORDER",  
  "quantityQNT": QTY,  
  "price":       PRICE,  
  "type":        TYPE  
}
```

Где:

- ACCOUNT номер аккаунта связанный с ордером
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon
- ASSETID идентификатор (ID) актива
- QTY количеств актива в ордере
- PRICE цена, задается в MilliLm
- TYPE тип ордера (т.е. покупка, продажа)
- HEIGHT высота блока

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=getAllOpenOrders
```

Ответ:

```
{  
  "openOrders":  
  [ {  
    "height":      90703,  
    "price":       "78932342",  
    "asset":       "15680163033397272955",  
    "order":       "17936094171153028796",  
    "quantityQNT": "20",  
    "account":     "8425933418718057425",  
    "accountRS":   "LMA-2UKS-7VYN-Q73Y-EKE8Y",  
    "type":        "ask"  
  },  
  {  

```

```

    ...
  }
}

```

Проверено 15/05/14

Cancel order

Отмена ранее созданного ордера

Запрос

```

http://localhost:1400/api?
requestType=BID_OR_ASK&
secretPhrase=SECRET&
order=ORDERID&
fee=FEE&
deadline=DEADLINE&
referencedTransaction=REFTXID&
broadcast=BROAD

```

Где:

- **BID_OR_ASK** указание типа операции "cancelAskOrder" или "cancelBidOrder"
- **SECRET** секретная фраза от аккаунта создавшего транзакцию
- **ORDERID** идентификатор(ID) ордера, который необходимо отменить
- **FEE** размер комиссии за данную транзакцию
- **DEADLINE** время (в минутах) в течении которого транзакция должна быть подтверждена
- **REFTXID** идентификатор транзакции (ID) на которую ссылается текущая транзакция. Используется если создана зависимая транзакция и означает, что текущая транзакция не может быть подтверждена, если транзакция, на которую ссылаются еще не подтверждена. Опциональный параметр.
- **BROAD** если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

'Ответ'

```

{
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "transactionBytes": "TXBYTES",
  "hash": "HASH",
  "transaction": "TX",
  "broadcasted": BROAD,
  "unsignedTransactionBytes": "UNTXBYTES"
}

```

- **FULLHASH** полный хэш транзакции

- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хеш всех полей транзакции, за исключением подписи
- TX ID транзакции
- BROADCAST статус - транзакция была передана или нет
- UNTXBYTES байты транзакции с частью подписи, заменяемой нулями

Пример

Запрос:

```
http://localhost:1400/api?
requestType=cancelAskOrder&
secretPhrase=123&
order=3467356724512&
fee=1000000000&
deadline=1440
```

Ответ:

```
{
  "fullHash":
    "e9c821547e851998f05ef39a39dc196b27f5e3069c1bf516fffaf53359304cc4",
  "signatureHash":
    "a5bf7481f5ad02be69f5fff7cd73c3dd7e443a4335721511e9e79198bc6f032d",
  "transactionBytes":
    "02051316db00a005668683e673274c671802492d6f54cdc33d247aa...",
  "hash":
    "10ffcff771db618ca243b561943e09fd4098ade100efb2dc68f7861705f6262d",
  "transaction":
    "10959937945806686441",
  "broadcasted":
    true,
  "unsignedTransactionBytes":
    "02051316db00a005668683e673274c671802492d6f54cdc33d247aa426e7c..."
}
```

Проверено 15/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции с опросами

Create Poll

Создать новый опрос

Запрос

```
http://localhost:1400/api?  
requestType=createPoll&  
name=NAME&  
description=DESCRIP&  
minNumberOfOptions=MINOPTIONS&  
maxNumberOfOptions=MAXOPTIONS&  
optionsAreBinary=BINARY&  
option1=OPTION1&  
option2=OPTION2&  
option3=OPTION3&  
secretPhrase=SECRET&  
fee=FEE&  
deadline=DEADLINE&  
referencedTransactionFullHash=REFTXHASH&  
broadcast=BROAD
```

Где:

- DESCRIP описание опроса
- NAME название опроса
- MINOPTIONS минимальное число вариантов для опроса
- MAXOPTIONS максимальное число вариантов для опроса
- OPTION1 название первой опции (OPTIONS 2 & 3 опционально)
- SECRET секретная фраза (private key) аккаунта
- DEADLINE дедлайн транзакции, задается в минутах
- FEE комиссия за транзакцию (в MilliLm)
- BINARY указатель бинарности опций (опционально)
- REFTXHASH Идентификатор (ID) предыдущей транзакции на которую ссылается данная транзакция (опционально)
- BROAD true или false если должна быть широковещательная передача (опционально)

Ответ

```
{  
  "fullHash": "FULLHASH",
```

```

    "signatureHash": "SIGHASH",
    "transactionBytes": "TXBYTES",
    "hash": "HASH",
    "transaction": "TX",
    "broadcasted": BROADCAST,
    "unsignedTransactionBytes": "UNTXYTES"
}

```

Где:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хеш всех полей транзакции, за исключением подписи
- TX ID транзакции
- BROADCAST статус - транзакция была передана или нет
- UNTXYTES байты транзакции с частью подписи, заменяемой нулями

Проверено 18/05/14

Пример

Запрос:

```

http://localhost:1400/api?
  requestType=createPoll&
  name=Testing123&
  description=Testing&
  minNumberOfOptions=1&
  maxNumberOfOptions=1&
  option1=yes&
  option2=no&

```

```

secretPhrase=MySuperSecretPasswordThatIsNotActuallyPostedHereOrIsEvenRealSoDoNot
EvenTryIt&
  fee=1000000000&
  deadline=1440

```

Ответ:

```

{
  "fullHash":
    "7ef9e19d0c7943002fd98a7ae08f1b4a1d473c67796f89225b167fd4913ce73f",
  "signatureHash":
    "a76722be0652d5c88cb7292d1cb04ea299eaea74c35ad726e868d08799946c38",
  "transactionBytes":
    "010234fbda00a005668683e673274c671802492d6f54cdc33d247aa426e...",
  "hash":
    "940383ff1bf177d4e4fee05c78865b12ace2570152c64570219dfc0aa10c246b",
  "transaction": "18991918535014782",
  "broadcasted": true,
  "unsignedTransactionBytes":
    "010234fbda00a005668683e673274c671802492d6f54cdc33d247..."
}

```


Проверено 13/06/14

Get Poll

Получить информацию о заданном опросе

Запрос

```
http://localhost:1400/api?  
requestType=getPoll&  
poll=POLL
```

Где:

- POLL идентификатор (id) опроса который вас интересует

Ответ

```
{  
  "description":      "DESCRIP",  
  "name":             "NAME",  
  "maxNumberOfOptions": MAXOPTIONS,  
  "optionsAreBinary": BINARY,  
  "voters":           [VOTERS],  
  "minNumberOfOptions": MINOPTIONS,  
  "options":           [OPTIONS]  
}
```

Where:

- DESCRIP описание опроса
- NAME название опроса
- MAXOPTIONS максимальное количество опций опроса
- BINARY указатель что опции бинарные
- VOTERS количество голосующих в опросе
- MINOPTIONS минимальное количество опций опроса
- OPTIONS список опций

Пример

Request:

```
http://localhost:1400/api?  
requestType=getPoll&  
poll=15826354927269615782
```

Response:

```
{  
  "description":      "Does this work?",  
  "name":             "Test",  
  "maxNumberOfOptions": 1,  
  "optionsAreBinary": false,  
  "voters":           [],  
  "minNumberOfOptions": 1,
```

```
    "options":          [ "Yes", "No", "I don't know" ]
}
```

Проверено 18/05/14

Get Poll IDs

Возвращает ID (идентификаторы) всех голосований

Запрос

http://localhost:1400/api?
requestType=getPollIds

Ответ

```
{
  "pollIds":
  [
    "POLLID",
    ...
  ]
}
```

Где:

- POLLID ID голосования

Пример

Запрос:

http://localhost:1400/api?
requestType=getPollIds

Ответ:

```
{
  "pollIds":
  [
    "15826354927269615782",
    "8883605113950920958"
  ]
}
```

Проверено 15/05/14

Cast Vote

Проголосовать в опросе

Запрос

http://localhost:1400/api?
requestType=castVote&
poll=NAME&
vote1=VOTE1&
vote2=VOTE2&
vote3=VOTE3&
secretPhrase=SECRET&

fee=FEE&
deadline=DEADLINE&
referencedTransactionFullHash=REFTXHASH&
broadcast=BROAD

Где:

- NAME название голосования в котором вы желаете принять участие, проголосовать
- VOTE1 ваш голос (VOTE 2 & 3 опционально, если голосование допускает множественные опции)
- SECRET секретная фраза (private key) вашего аккаунта
- DEADLINE дедлайн транзакции, задается в минутах
- FEE комиссия за транзакцию (в MilliLm)
- REFTXHASH Идентификатор (ID)предыдущей транзакции на которую ссылается данная транзакция (опционально)
- BROAD если вы хотите чтобы транзакция широковещательно передавалась или нет (опционально)

Ответ

```
{  
  "fullHash":          "FULLHASH",  
  "signatureHash":     "SIGHASH",  
  "transactionBytes":  "TXBYTES",  
  "hash":              "HASH",  
  "transaction":       "TX",  
  "broadcasted":       BROAD,  
  "unsignedTransactionBytes": "UNTXBYTES"  
}
```

Where:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- HASH хеш всех полей транзакции, за исключением подписи
- TX ID транзакции
- BROAD статус - транзакция была передана или нет
- UNTXBYTES байты транзакции с частью подписи, заменяемой нулями

Проверено 18/05/14

Пример

Запрос:

```
http://localhost:1400/api?  
  requestType=castVote&  
  poll=18991918535014782&  
  vote1=yes&
```

```
secretPhrase=MySuperSecretPasswordThatIsNotActuallyPostedHereOrIsEvenRealSoDoNot  
EvenTryIt&  
  fee=1000000000&  
  deadline=1440
```

Ответ:

```
{  
  "fullHash":  
    "637ea3215258adef61f7fc5a90eff6435d1a7ad8c3790d5855c6943b9aff1daf",  
  "signatureHash":  
    "a6745e26165e9c7870c85bd174348e4e72c825378cfb71760e04f08e30879fa6",  
  "transactionBytes":  
    "010314feda00a005668683e673274c671802492d6f54cdc33...",  
  "hash":  
    "10ee68836162f6a8e2559a6dbcbc41c4db4b7441e6ff321f01b268d761c6916d",  
  "transaction":  
    "17270557255810645603",  
  "broadcasted":  
    true,  
  "unsignedTransactionBytes":  
    "010314feda00a005668683e673274c671802492d6f54cdc33d247aa426e7..."  
}
```

Проверено 13/06/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции для работы с серверной частью

Get Constants

Запрос на получение списка всех констант, используемых сервером, и их текущих значений.

Request

```
http://localhost:1400/api?  
requestType=getConstants
```

Запрос

```
{  
  "maxBlockPayloadLength":40800,  
  "genesisAccountId":"GENESISACC",  
  "genesisBlockId":"GENESISBLK",  
  "transactionTypes":  
  [  
    {  
      "description":"Payment","value":0,"subtypes":  
      [  
        {"description":"Ordinary payment","value":0}  
      ]  
    },  
    {  
      "description":"Messaging","value":1,"subtypes":  
      [  
        {"description":"Arbitrary message","value":0},  
        {"description":"Alias assignment","value":1},  
        {"description":"Poll creation","value":2},  
        {"description":"Vote casting","value":3},  
        {"description":"Hub terminal announcement","value":4},  
        {"description":"Account info","value":5}  
      ]  
    },  
    {  
      "description":"Colored coins","value":2,"subtypes":  
      [  
        {"description":"Asset issuance","value":0},  
        {"description":"Asset transfer","value":1},  
        {"description":"Ask order placement","value":2},  
        {"description":"Bid order placement","value":3},  
        {"description":"Ask order cancellation","value":4},  
        {"description":"Bid order cancellation","value":5}  
      ]  
    },  
    {  
      "description":"Digital goods","value":3,"subtypes":  
      [  
        {"description":"Listing","value":0},  
        {"description":"Delisting","value":1},  
        {"description":"Price change","value":2},
```

```

        {"description": "Quantity change", "value": 3},
        {"description": "Purchase", "value": 4},
        {"description": "Delivery", "value": 5},
        {"description": "Feedback", "value": 6},
        {"description": "Refund", "value": 7}
    ]
},
{
    "description": "Account Control", "value": 4, "subtypes":
    [
        {"description": "Effective balance leasing", "value": 0}
    ]
}
],
"peerStates":
[
    {"description": "Non-connected", "value": 0},
    {"description": "Connected", "value": 1},
    {"description": "Disconnected", "value": 2}
]
}

```

Где:

- GENESISACC генезис аккаунт
- GENESISBLK генезис блок

Пример

Запрос:

http://localhost:1400/api?
requestType=getConstants

Ответ:

```

{
    "maxBlockPayloadLength": 40800,
    "genesisAccountId": "1739068987193023818",
    "genesisBlockId": "2680262203532249785",
    "transactionTypes":
    [
        {
            "description": "Payment", "value": 0, "subtypes":
            [
                {"description": "Ordinary payment", "value": 0}
            ]
        },
        {
            "description": "Messaging", "value": 1, "subtypes":
            [
                {"description": "Arbitrary message", "value": 0},
                {"description": "Alias assignment", "value": 1},
                {"description": "Poll creation", "value": 2},
                {"description": "Vote casting", "value": 3},
                {"description": "Hub terminal announcement", "value": 4},
                {"description": "Account info", "value": 5}
            ]
        },
        {
            "description": "Colored coins", "value": 2, "subtypes":

```

```

    [
      {"description": "Asset issuance", "value": 0},
      {"description": "Asset transfer", "value": 1},
      {"description": "Ask order placement", "value": 2},
      {"description": "Bid order placement", "value": 3},
      {"description": "Ask order cancellation", "value": 4},
      {"description": "Bid order cancellation", "value": 5}
    ],
    {
      "description": "Digital goods", "value": 3, "subtypes":
      [
        {"description": "Listing", "value": 0},
        {"description": "Delisting", "value": 1},
        {"description": "Price change", "value": 2},
        {"description": "Quantity change", "value": 3},
        {"description": "Purchase", "value": 4},
        {"description": "Delivery", "value": 5},
        {"description": "Feedback", "value": 6},
        {"description": "Refund", "value": 7}
      ]
    },
    {
      "description": "Account Control", "value": 4, "subtypes":
      [
        {"description": "Effective balance leasing", "value": 0}
      ]
    }
  ],
  "peerStates":
  [
    {"description": "Non-connected", "value": 0},
    {"description": "Connected", "value": 1},
    {"description": "Disconnected", "value": 2}
  ],
  "maxArbitraryMessageLength": 1000
}

```

Проверено 15/05/14

Get My Info

возвращает данные сервера - хост и адрес.

Запрос

http://localhost:1400/api?
requestType=getMyInfo

Ответ

```

{
  "host":      "HOSTNAME",
  "address":   "ADDRESS"
}

```

Где:

- HOSTNAME - имя хоста сервера
- ADDRESS - адрес сервера

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=getMyInfo
```

Ответ:

```
{  
  "host":      "127.0.0.1",  
  "address":   "127.0.0.1"  
}
```

Проверено 18/05/14

Get Peer

получает данные о заданном узле (peer)

Запрос

```
http://localhost:1400/api?  
requestType=getPeer&  
peer=PEERNAME
```

Где:

- PEERNAME - имя узла (peer) о котором вы желаете запросить информацию.

Ответ:

Информация возвращается в том виде и объеме, который хранится на сервере узла в файле web.xml file !

```
{  
  "platform":      "PLATFORM",  
  "blacklisted":   BLACKLIST,  
  "shareAddress":  SHAREADD  
  "application":   "LIBREMONEY",  
  "weight":        WEIGHT,  
  "hallmark":      "HALLMARK",  
  "state":         1,  
  "announcedAddress": "ANNOUNCED",  
  "downloadedVolume": DLBYTES,  
  "version":       "VERSION",  
  "uploadedVolume": ULBYTES  
}
```

Где:

- PLATFORM строка описывающая платформу указанного сервера
- WEIGHT вес узла
- BLACKLIST указывает включен узел в blacklisted или нет
- SHAREADD указывает является ли узел общедоступным (shared)

- HALLMARK строка в формате hex, описывающая hallmark узла (если он установлен)
- STATE определяет состояние узла. 1 - активный узел; 0 - известный узел.
- ANNOUNCED имя, которое узел сообщает сети (может быть имя DNS, IP-адрес или любая другая строка)
- VERSION версия программного обеспечения, работающего на узле
- DLBYTES and ULBYTES количество байтов, загруженных и выгруженных узлом

Проверено 18/05/14

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getPeer&
  peer=23.88.104.217
```

Ответ:

```
{
  hallmark":          "ea53f569819ad8c40d62dd7e4c1f4895b2a3...",
  "downloadedVolume": 115001,
  "blacklisted":      false,
  "announcedAddress": "mode.libremoney.org",
  "application":      "LIBREMONEY",
  "weight":           911527,
  "uploadedVolume":   4707,
  "state":             1,
  "version":           "0.9.9",
  "platform":          "lm.now.im",
  "shareAddress":      true
}
```

Проверено 15/05/14

Get Peers

Возвращает все известные вашему клиенту узлы (peers), не зависимо от их состояния на данный момент.

Запрос

```
http://localhost:1400/api?
  requestType=getPeers
```

Результат:

Массив имен/адресов узлов:

```
{
  "peers":
  [
    "PEER",
```

```

        "PEER",
        "PEER"...
    ]
}

```

Пример

Запрос:

http://localhost:1400/api?
requestType=getPeers

Ответ:

```

{
    "peers":
    [
        "188.226.176.151",
        "88.202.37.38",
        "87.211.164.253"...
    ]
}

```

Проверено 18/05/14

Get State

Возвращает информацию о статусе/состоянии запрашиваемого сервера

Запрос

http://localhost:1400/api?
requestType=getState

Ответ

```

{
    "numberOfPolls":          NUMPOLLS,
    "numberOfVotes":          NUMVOTES,
    "numberOfTrades":         NUMTRADES,
    "lastBlock":              "LASTBLOCKID",
    "numberOfAliases":        NUMALIASES,
    "lastBlockchainFeeder":    "FEEDERPEER",
    "numberOfBlocks":         HEIGHT,
    "numberOfPeers":          NUMPEERS,
    "totalMemory":             CURMEMORY,
    "freeMemory":              FREEMEMORY,
    "maxMemory":               MAXMEMORY,
    "numberOfTransactions":    NUMTRANS,
    "numberOfUnlockedAccounts": NUMUSERS,
    "version":                 "VERSION",
    "numberOfOrders":         NUMORDERS,
    "totalEffectiveBalance":    EFFECTIVEBALANCE,
    "time":                    TIME,
    "availableProcessors":     NUMPROCESSORS,
    "numberOfAssets":          NUMASSETS,
    "cumulativeDifficulty":     "CUMEDIFF",
    "numberOfAccounts":        NUMACCOUNTS
}

```

Где:

- LASTBLOCKID идентификатор (ID) последнего полученного блока
- NUMALIASES количество известных алиасов в сети
- FEEDERPEER опубликованное имя узла фидера в последнем блоке
- HEIGHT высота блока
- NUMPEERS количество узлов в сети
- CURMEMORY общий объем памяти используемой узлом (в Байтах)
- FREEMEMORY общий объем свободной памяти на узле (в Байтах)
- MAXMEMORY максимальный объем памяти, доступной для использования на узле (в Байтах)
- NUMTRANS число транзакций в blockchain
- NUMUSERS число разблокированных аккаунтов на этом узле
- NUMPOLLS количество опросов на этом узле
- NUMTRADES количество сделок на этом узле
- NUMVOTES количество голосов (для опросов) на этом узле
- NUMORDERS количество ордеров на узле
- EFFECTIVEBALANCE эффективный баланс, актуальный для сети, указанный в LibreMoney-узлах
- NUMPROCESSORS количество процессоров на данном узле
- NUMASSETS число счетов на узле
- CUMEDIFF текущий, накопительный уровень сложности форжинга
- NUMACCOUNTS количество аккаунтов в blockchain

Пример

Запрос:

```
http://localhost:1400/api?
requestType=getState
```

Ответ:

```
{
  "numberOfPeers":868,
  "numberOfPolls":0,
  "numberOfVotes":0,
  "availableProcessors":4,
  "totalEffectiveBalance":991634266,
  "numberOfAccounts":33172,
  "numberOfUnlockedAccounts":0,
  "numberOfBlocks":126493,
  "numberOfOrders":0,
  "numberOfTransactions":182379,
  "version":"1.0.0",
  "maxMemory":477626368,
  "lastBlock":"4329081735143326181",
  "totalMemory":466092032,
```

```
{
  "numberOfAliases":85657,
  "cumulativeDifficulty":"3936422822105943",
  "numberOfTrades":0,
  "time":13671770,
  "numberOfAssets":0,
  "lastBlockchainFeeder":"cubie-solar.mjke.de",
  "freeMemory":135007384
}
```

Проверено 15/05/14

Get Time

Получает текущее время в системе. Внимание ! В LibreMoney время исчисляется в секундах с момента генерации генезисного блока, т.е. с момента зарождения LibreMoney

Запрос

```
http://localhost:1400/api?
  requestType=getTime
```

Ответ

```
{
  "time":      TIME
}
```

Где:

- TIME количество секунд прошедших с момента генерации генезисного блока.

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getTime
```

Ответ:

```
{
  "time":      417
}
```

Проверено 21/06/14

Get Blockchain Status

Получить текущий статус blockchain

Запрос

```
http://localhost:1400/api?
  requestType=getBlockchainStatus
```

Ответ

```
{
  "lastBlock": "LASTBLK",
  "lastBlockchainFeederHeight": LASTHEIGHT,
  "time": TIME,
  "lastBlockchainFeeder": "FEEDERPEER",
  "numberOfBlocks": NUMBLKS,
  "isScanning": SCAN,
  "cumulativeDifficulty": "CUMDIF",
  "version": "VER"
}
```

Where:

- LASTBLK последний блок в blockchain
- LASTHEIGHT высота последнего блока
- TIME число секунд начиная с момента генерации генезисного блока.
- FEEDERPEER опубликованное имя хоста фидера последнего блока
- NUMBLKS общее количество блоков
- SCAN возвращает статус blockchain - происходит сейчас сканирование или нет.
- CUMDIF текущая сложность форжинга
- VER версия LibreMoneyNode который сейчас выполняется

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getBlockchainStatus
```

Ответ:

```
{
  "lastBlock": "3153685344875873318",
  "lastBlockchainFeederHeight": 95337,
  "time": 14604616,
  "lastBlockchainFeeder": "192.241.223.132",
  "numberOfBlocks": 95339,
  "isScanning": false,
  "cumulativeDifficulty": "2406110597084752",
  "version": "1.1.0"
}
```

Проверено 18/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции с клеймами (hallmarks)

Decode Hallmark

Декодирует клеймо узла.

Запрос

```
http://localhost:1400/api?  
requestType=decodeHallmark&  
hallmark=HEXSTRING
```

Где:

- HEXSTRING — значение клейма

Ответ

```
{  
  "valid":          VALID,  
  "weight":         WEIGHT,  
  "host":           "HOSTIP",  
  "account":        "ACCOUNT",  
  "accountRS":      "ACCOUNTRS",  
  "date":           "DATE"  
}
```

Где:

- VALID — либо true (истина), либо false (ложь)
- WEIGHT — вес, присвоенный клейму
- HOSTIP — IP-адрес или имя домена, ассоциированные с клеймом
- ACCOUNT — номер аккаунта, ассоциированный с клеймом
- ACCOUNT номер аккаунта ассоциированный с заданным клеймом
- ACCOUNTRS адрес аккаунта в кодировке Reed-Solomon
- DATE — дата создания клейма, в формате уууу-mm-dd

Пример

Запрос:

```
http://127.0.0.1:1400/api?  
requestType=decodeHallmark&  
hallmark=d67f03a47cd91fc...8b9e30f371a9fcc8f
```

Ответ:

```
{
  "valid":      true,
  "weight":     1000,
  "host":       "162.23.47.189",
  "account":    "LMA-5EGB-HKFP-NQUT-DKYFH",
  "accountRS":  "ACCONTRS",
  "date":       "2013-12-09"
}
```

Проверено 15/05/14

Generate Hallmark

Генерирует клеймо узла.

Запрос

```
http://localhost:1400/api?
  requestType=markHost&
  secretPhrase=MY_SECRET&
  host=MY_HOST&
  weight=WEIGHT&
  date=CURRENT_DATE
```

Где:

- MY_SECRET — секретная фраза аккаунта, для которого на ноде создается клеймо. Должна быть URL-encoded.
- MY_HOST — имя хоста для узла, которую вы хотите "заклеймить"
- WEIGHT — вес, который вы хотели бы назначить ноде. Если на одном и том же аккаунте "заклеймены" более, чем одна нода, этот вес будет определен как (этот_вес/сумма_всех_весов)
- CURRENT_DATE — сегодняшняя дата, в формате уууу-mm-dd

Ответ

```
{
  "hallmark":      "HEXSTRING"
}
```

Где:

- HEXSTRING — строка с закодированным клеймом (1856 байт)

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=markHost&
  secretPhrase=MySuperSecretPasswordThatIsNotEasy&
  host=node23.nodesrus.com&
  weight=100&
  date=2014-01-01
```

Ответ:

```
{  
  "hallmark": "d67f03a47cd91fc1...fd49f66b462daa1f14ae8"  
}
```

Проверено 18/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции с токенами

Decode Token

Декодирует токен авторизации. Используется для авторизации аккаунта на указанном веб-сайте, без необходимости передачи секретной фразы.

Запрос

```
http://localhost:1400/api?  
requestType=decodeToken&  
website=WEBSITE&  
token=AUTHSTRING
```

Где:

- WEBSITE — URL веб-сайта, где необходима авторизация по токену. По принятому соглашению не содержит часть "http://" URL'a.
- AUTHSTRING — закодированная строка авторизации

Ответ

```
{  
  "account":      "ACCOUNT",  
  "timestamp":    TIME,  
  "valid":        BOOLEAN  
}
```

Где:

- ACCOUNT — номер аккаунта LibreMoney, ассоциированный с токеном
- TIME — время в секундах начиная с 0 времени создания блока генезиса, определяющее время создания токена
- BOOLEAN — либо "true" (истина), либо "false" (ложь), показывает валиден ли токен

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=decodeToken&  
website=www.domain.com&  
token=StringOf160Chars
```

Ответ:

```
{
  "account":      "398532577100249608",
  "timestamp":    622,
  "valid":        true
}
```

Проверено 18/05/14

Generate Token

Создаёт токен (ключ) авторизации. Используется для авторизации аккаунта на определенном веб-сайте без необходимости передачи секретной фразы.

Запрос

```
http://localhost:1400/api?
  requestType=generateToken&
  secretPhrase=SECRET&
  website=WEBSITE
```

Где:

- **SECRET** - ключевая фраза (private key) аккаунта, который создаёт токен.
- **WEBSITE** - URL веб-сайта, на котором проходит авторизация. По соглашению не содержит часть "http://" URL'a.

Ответ

```
{
  "token": StringOf160Chars
}
```

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=generateToken&
  secretPhrase=ItWasABrightColdDayInAprilAndTheClocksWereStrikingThirteen&
  website=www.genesisblock.com
```

ОТВЕТ:

```
{
  "token":
  "7dstcbs4fnbq614faea405ga3dj...etb0n9f0k6uvlku4d751kftqlb128t4"
}
```

Проверено 18/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

Операции с транзакциями

Broadcast Transaction

Широковещательная передачи транзакции в сеть

Запрос

```
http://localhost:1400/api?  
requestType=broadcastTransaction&  
transactionBytes=TX_BYTECODE
```

Где:

- TX_BYTECODE - байткод транзакции

Ответ

```
{  
  "transaction":      "TRANSACTION_ID"  
}
```

Пример

Запрос:

```
http://localhost:1400/api?  
requestType=broadcastTransaction&  
transactionBytes= 000022e22800e8036653833f4272604d...
```

Ответ:

```
{  
  "transaction":      "17484500871708980634"  
}
```

- Внимание: Вы можете получить ошибку "'error': 'Double spending transaction'" если транзакция уже была транслирована в сеть

Проверено 18/05/14

Calculate Full Hash

Вычислить полный хэш транзакции

Запрос

```
http://localhost:1400/api?  
requestType=calculateFullHash&  
unsignedTransactionBytes=UNTXBYTES&
```

signatureHash=SIGHASH

Где:

- UNTXBYTES неподписанные байты, содержащиеся в транзакции
- SIGHASH sha256 хэш подписи транзакции. Применяется для операций условного депонирования

Ответ

```
{
  "fullHash": "FULLHASH"
}
```

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=calculateFullHash&
  unsignedTransactionBytes=01016e23d700a005668683e673274c67...&

signatureHash=07afbcbfb72b4fa91d8e22f8f47950404270edc1d250d2c6edad7a58679fbe1cf
```

ОТВЕТ:

```
{
  "fullHash":
  "cc98e57a065610d17c6ba6a6406e4930b9a740bf1737847ab2e96e3733233984"
}
```

Проверено 15/05/14

Get Account Transaction IDs

Позволяет получать список идентификаторов транзакций, связанных с указанным аккаунтом

Запрос

```
http://localhost:1400/api?
  requestType=getAccountTransactionIds&
  account=ACCOUNT&
  timestamp=TIME
```

Где:

- ACCOUNT - номер аккаунта по которому вы запрашиваете информацию
- TIME - время в секундах, с момента генерации генезисного блока

Ответ

```
{
  "transactionIds":
    [ARRAY_OF_IDS]
}
```

Где:

- **ARRAY_OF_IDS** - список (массив) идентификаторов транзакций, связанных с указанным аккаунтом

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getAccountTransactionIds&
  account=6331110009135084386&
  timestamp=0
```

Ответ:

```
{
  "transactionIds":
    [
      "17484500871708980634"
    ]
}
```

Get Unconfirmed Account Transaction IDs

Получить список идентификаторов неподтвержденных транзакций связанных с заданным LibreMoney аккаунтом

Запрос

```
http://localhost:1400/api?
  requestType=getUnconfirmedTransactionIds&
  account=ACCOUNT
```

Где:

- **ACCOUNT** номер аккаунта, список идентификаторов неподтвержденных транзакций которого, вы хотите получить

Ответ

```
{
  "unconfirmedTransactionIds":
    [ARRAY_OF_IDS]
}
```

Где:

- **ARRAY_OF_IDS** массив идентификаторов неподтвержденных транзакций, относящихся к заданному аккаунту

Пример

Запрос:

```
http://localhost:1400/api?
```

```
requestType=getUnconfirmedTransactionIds&
account=6331110009135084386
```

Ответ:

```
{
  "unconfirmedTransactionIds":
    [
      "17484500871708980634"
    ]
}
```

Проверено 18/05/14

Get Transaction

Возвращает данные заданной транзакции. Может задаваться **любым** параметром: ID транзакции, hash, или fullhash транзакции.

Запрос

```
http://localhost:1400/api?
requestType=getTransaction&
transaction=TRANSID&
hash=HASH&
fullHash=FULLHASH
```

Где:

- TRANSID ID транзакции LibreMoney
- HASH hash транзакции
- FULLHASH fullhash транзакции

Ответ

```
{
  "sender": "SENDERACCOUNT",
  "senderRS": "SENDERACCONTRS",
  "fee": "FEE",
  "amount": "AMOUNT",
  "timestamp": TIME,
  "referencedTransaction": REFTX,
  "confirmations": CONFIRMS,
  "subtype": SUBTYPE,
  "block": "BLOCKID",
  "senderPublicKey": "PUBKEY",
  "type": TYPE,
  "deadline": DEADLINE,
  "signature": "SIGNATURE",
  "recipient": "RECIPACCOUNT",
  "recipientRS": "RECIPACCONTRS",
  "fullHash": "FULLHASH",
  "signatureHash": "SIGHASH",
  "hash": "HASH",
  "transaction": "TRANSID",
  "attachment":
    {
      ATTACHMENT
    }
}
```

}

Где:

- SENDERACCOUNT номер аккаунта отправителя
- SENDERACCONTRS адрес отправителя в кодировке Reed-Solomon
- FEE комиссия (в MilliLm) за данную транзакцию
- AMOUNT количество MilliLm в данной транзакции
- TIME время транзакции, измеряется в количестве секунд от времени создания генезисного блока
- REFTX Идентификатор (ID)предыдущей транзакции на которую ссылается данная транзакция. Должно быть "0" если отсутствует транзакция на которую ссылается данная.
- CONFIRMS количество подтверждений транзакции
- TYPE тип транзакции:
 - 0 обыкновенная транзакция
 - 1 альяс или произвольное сообщение в составе транзакции
 - 2 если это транзакция биржи активов
- SUBTYPE субтип транзакции:
 - Для альясов/произвольных сообщений:
 - 0 транзакция произвольного сообщения
 - 1 транзакция альяса
 - Для биржи активов:
 - 0 выпуск актива
 - 1 трансфер актива
 - 2 запрос на размещение заказа на покупку
 - 3 запрос на размещение заказа на продажу
 - 4 отмена ордера на покупку
 - 5 отмена ордера на продажу
- BLOCKID ID идентификатор блока содержащий данную транзакцию
- ATTACHMENT суб-бъект, представляющий любые дополнительные данные, необходимые для транзакции. Будет присутствовать только если применимо к транзакции.
- PUBKEY публичный ключ отправляющего аккаунта для транзакции
- DEADLINE дедлайн транзакции, указывается в минутах
- SIGNATURE цифровая подпись транзакции transaction

- SIGHASH хэш sha256 подписи транзакции. Используется для условного операций депонирования.
- RECIPACCOUNT номер аккаунта получателя
- RECIPACCONTRS адрес получателя в кодировке Reed-Solomon

Внимание: Данные о неподтвержденной транзакции не содержат параметров "block" и/или "confirmations". Данные double-spending транзакций не возвращаются.

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getTransaction&
  transaction=16244659048134841060
```

Ответ:

```
{
  "sender": "14336428216306750617",
  "senderRS": "LMA-6DWM-GFYD-A7PS-72P3P",
  "fee": "100000000",
  "amount": "0",
  "timestamp": 7376798,
  "referencedTransaction": "0",
  "confirmations": 6,
  "subtype": 1,
  "block": "16385359704305994802",
  "blockTimestamp": 7376860,
  "attachment": {
    "alias": "LmApi",
    "uri": "http://libremoney.org/"
  },
  "senderPublicKey":
"64f103fbf0cc41045b1544cda6624eecfb8212b021fba22029a461c4b40c5d79",
  "type": 1,
  "deadline": 1,
  "signature": "0cdf64cb06ea21396180a6f201b6a...",
  "recipient": "1739068987193023818",
  "recipientRS": "LMA-BK2J-ZMY4-93UY-8EM9V",
  "fullHash":
"a3c47c4a2e60c2907efc9c4b9310066cabdf3e2137e67814aa57cb960ad2dfe1",
  "signatureHash":
"95cc276d1608dc0c14942a5efc82ccc701a56e873746ef1bfd37995c37ef5991",
  "hash": "15a86f71e1ac267dce35791cefafb71460f458920784ea0d10894844f05894b0",
  "transaction": "10431005438877942947"
}
```

Проверено 15/05/14

Parse Transaction

Получает данные транзакции

Запрос

```
http://localhost:1400/api?
requestType=parseTransaction&
transactionBytes=TRANSBYTES
```

Где:

- TRANSBYTES байты LibreMoney транзакции

Ответ

```
{
  "sender":          "SENDERACCOUNT",
  "senderRS":        "SENDERACCONTRS",
  "fee":             "FEE",
  "amount":          "AMOUNT",
  "timestamp":       TIME,
  "referencedTransaction": REFTX,
  "confirmations":   CONFIRMS,
  "subtype":         SUBTYPE,
  "block":           "BLOCKID",
  "senderPublicKey": "PUBKEY",
  "type":            TYPE,
  "deadline":        DEADLINE,
  "signature":        "SIGNATURE",
  "recipient":        "RECIPACCOUNT",
  "recipientRS":      "RECIPACCONTRS",
  "fullHash":         "FULLHASH",
  "signatureHash":    "SIGHASH",
  "hash":             "HASH",
  "transaction":      "TRANSID",
  "attachment":      {
    ATTACHMENT
  }
}
```

Где:

- SENDERACCOUNT номер аккаунта отправителя
- SENDERACCONTRS адрес отправителя в кодировке Reed-Solomon
- FEE комиссия (в MilliLm) за данную транзакцию
- AMOUNT количество MilliLm в данной транзакции
- TIME время транзакции, измеряется в количестве секунд от времени создания генезисного блока
- REFTX Идентификатор (ID)предыдущей транзакции на которую ссылается данная транзакция. Должно быть "0" если отсутствует транзакция на которую ссылается данная.
- CONFIRMS количество подтверждений транзакции
- TYPE тип транзакции:
 - 0 обыкновенная транзакция

- 1 альяс или произвольное сообщение в составе транзакции
- 2 если это транзакция биржи активов
- SUBTYPE субтип транзакции:
 - Для альясов/произвольных сообщений:
 - 0 транзакция произвольного сообщения
 - 1 транзакция альяса
 - Для биржи активов:
 - 0 выпуск актива
 - 1 трансфер актива
 - 2 запрос на размещение заказа на покупку
 - 3 запрос на размещение заказа на продажу
 - 4 отмена ордера на покупку
 - 5 отмена ордера на продажу
- BLOCKID ID идентификатор блока содержащий данную транзакцию
- ATTACHMENT суб-бъект, представляющий любые дополнительные данные, необходимые для транзакции. Будет присутствовать только если применимо к транзакции.
- PUBKEY публичный ключ отправляющего аккаунта для транзакции
- DEADLINE дедлайн транзакции, указывается в минутах
- SIGNATURE цифровая подпись транзакции transaction
- SIGHASH хэш sha256 подписи транзакции. Используется для условного операций депонирования.
- RECIPACCOUNT номер аккаунта получателя
- RECIPACCONTRS адрес получателя в кодировке Reed-Solomon

Внимание: Данные о неподтвержденной транзакции не содержат параметров "block" и/или "confirmations". Данные double-spending транзакций не возвращаются.

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=parseTransaction&
  transactionBytes=000022e22800e8036653833f4272604d5d3ef39...
```

Ответ:

```
{
  "sender":      "14336428216306750617",
```

```

"senderRS": "LMA-6DWM-GFYD-A7PS-72P3P",
"fee": "1000000000",
"amount": "0",
"timestamp": 7376798,
"referencedTransaction": "0",
"confirmations": 6,
"subtype": 1,
"block": "16385359704305994802",
"blockTimestamp": 7376860,
"senderPublicKey":
"64f103fbf0cc41045b1544cda6624eecfb8212b021fba22029a461c4b40c5d79",
"type": 1,
"deadline": 1,
"signature": "0cdf64cb06ea21396180a6f201b6a...",
"recipient": "1739068987193023818",
"recipientRS": "LMA-BK2J-ZMY4-93UY-8EM9V",
"fullHash":
"a3c47c4a2e60c2907efc9c4b9310066cabdf3e2137e67814aa57cb960ad2dfe1",
"signatureHash":
"95cc276d1608dc0c14942a5efc82ccc701a56e873746ef1bfd37995c37ef5991",
"hash": "15a86f71e1ac267dce35791cefafb71460f458920784ea0d10894844f05894b0",
"transaction": "10431005438877942947"
}

```

Проверено 15/05/14

Get Transaction Bytes

Возвращает байт-код транзакции.

Запрос

```

http://localhost:1400/api?
requestType=getTransactionBytes&
transaction=TXID

```

Где:

- TXID - идентификатор транзакции (ID)

Ответ

```

{
  "confirmations": CONF,
  "transactionBytes": "BYTECODE",
  "unsignedTransactionBytes": "UNBYTECODE"
}

```

Где:

- CONF количество подтверждений транзакции
- BYTECODE запрашиваемый байт-код транзакции
- BYTECODE неподписанные байты, включенные в транзакцию

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=getTransactionBytes&
  transaction=17484500871708980634
```

Ответ:

```
{
  "confirmations":      4462,
  "transactionBytes":    "000022e22800e8036653833f4272604d5...",
  "unsignedTransactionBytes": "000022e22800e8036653833f42..."
}
```

Проверено 18/05/14

Send Lm

Переслать указанное количество MilliLm на указанный аккаунт.

Запрос

```
http://localhost:1400/api?
  requestType=sendMoney&
  secretPhrase=SECRET&
  recipient=RECIPACCOUNT&
  amount=AMOUNT&
  fee=FEE&
  deadline=DEADLINE&
  referencedTransaction=REFTRANSACTION&
  publicKey=PUBKEY
```

Где:

- SECRET - секретная фраза/пароль для аккаунта с которого будут отправлены средства
- RECIPACCOUNT - аккаунт получателя
- AMOUNT - Количество MilliLm передаваемых в данной транзакции
- FEE - комиссия (в MilliLm) для этой транзакции
- DEADLINE - deadline для транзакции, в минутах
- REFTRANSACTION - опционально: ссылка на идентификатор (ID) предыдущей транзакции (основная транзакция не подтвердится, пока не подтвердилась предыдущая)
- PUBKEY 64-битный публичный ключ ассоциированный с аккаунтом и дополняет поле SECRET.

Ответ

```
{
  "transaction":      "TRANSACTIONID"
}
```

Где:

- TRANSACTIONID - идентификатор только что созданной транзакции.

'Внимание! "deadline" указывается в минутах. "referencedTransaction" не обязательный параметр, может не использоваться.

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=sendMoney&
  secretPhrase=IWontTellYou&
  recipient=398532577100249608&
  amount=1500&
  fee=5&
  deadline=900&
  referencedTransaction=13689168149259791567
```

Результат:

```
{
  "transaction":      "16244659048134841060"
}
```

Проверено 18/05/14

Sign Transation

Рассчитать полный хэш (full hash), хэш подписи (signature hash), и транзакцию из неподписанной транзакции

Request

```
http://localhost:1400/api?
  requestType=signTransaction&
  unsignedTransactionBytes=UNTXBYTES&
  secretPhrase=SECRET
```

Где:

- UNTXBYTES неподписанные байты в транзакции
- SECRET секретная фраза аккаунта

Ответ

```
{
  "fullHash":          "FULLHASH",
  "signatureHash":     "SIGHASH",
  "transactionBytes":  "TXBYTES",
  "transaction":       "TX",
  "verify":            "VERIFY"
}
```

Где:

- FULLHASH полный хэш транзакции
- SIGHASH sha256 хэш подписи транзакции. Используется для транзакций условного депонирования
- TXBYTES байткод транзакции
- TX ID транзакции
- VERIFY если верифицировано (true/false)

Пример

Запрос:

```
http://localhost:1400/api?
  requestType=signTransaction&
  unsignedTransactionBytes=01016e23d700a005668683e673274c67...&
  secretPhrase=0123456789
```

Ответ:

```
{
  "fullHash":
    "d56ade2c322083d76ebcdb9f9ef18d7d52c6d105f67be040cc423c7a41a47aea",
  "signatureHash":
    "458514e72f4272c789f99201c27cff611460de10fe484b474166c1199c23e754",
  "transactionBytes": "0100471ddb00a005668683e673274c67180240...",
  "transaction":      "15529291339976829653",
  "verify":           true
}
```

Проверено 15/05/14

LibreMoney

2. Основная часть

2.8. API (RestAPI)

""Коды ошибок""

- Код ошибки: 1 - Incorrect request. Неверный запрос. Скорее всего вы допустили ошибку при написании/копировании запроса.

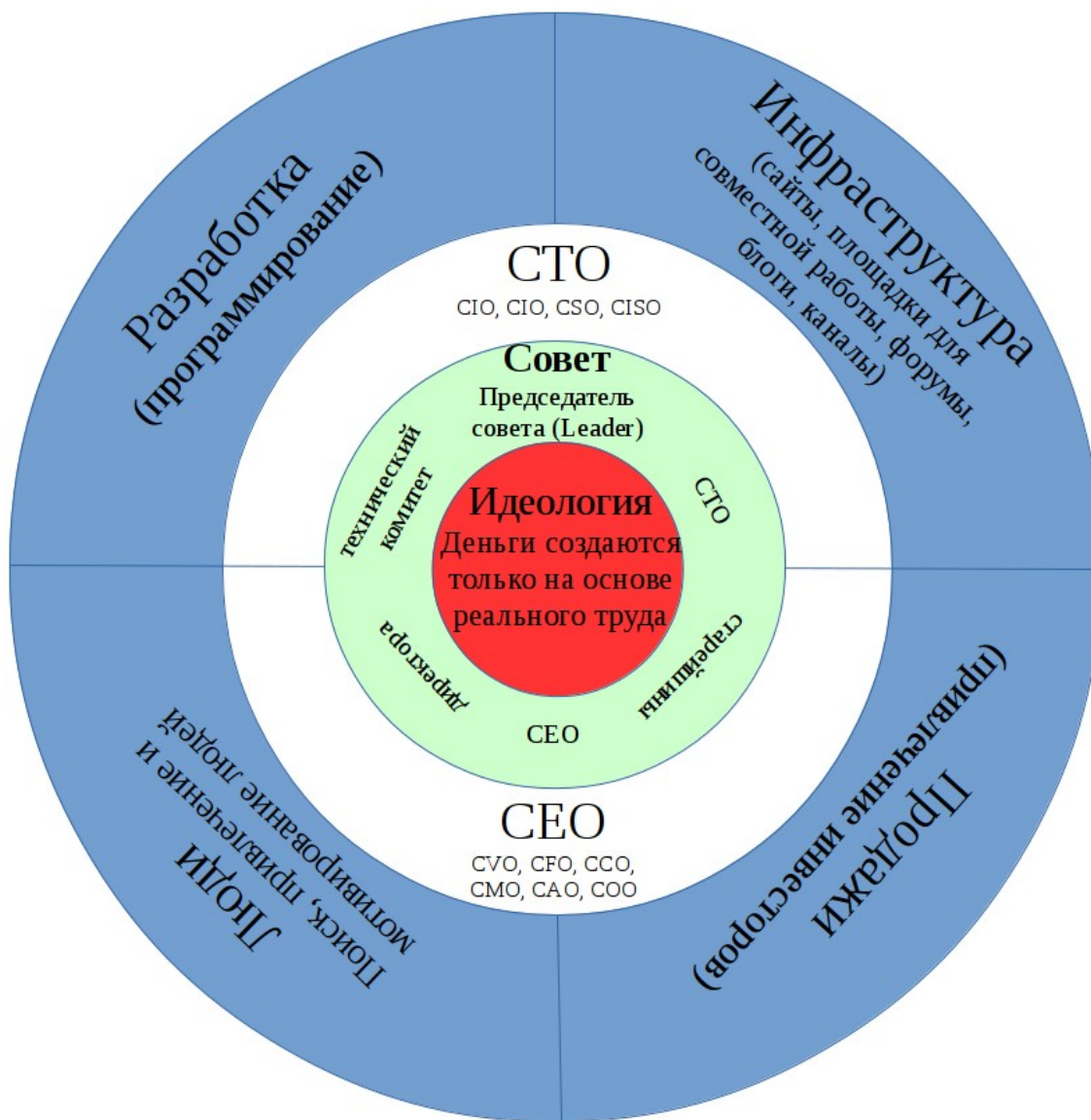
Внимательно смотрите запрос к серверу http://localhost:1400/api?requestType=здесь_должна_быть_введена_корректная_функция_системы&...

- Код ошибки: 2 - Blockchain not up to date. Blockchain не обновлен, т.е. находится в неактуальном состоянии.
- Код ошибки: 3 - Parameter not specified. Параметр не определен. Скорее всего вы указали в запросе параметр, которого не должно быть (лишний, некорректный)
- Код ошибки: 4 - Incorrect parameter. Некорректный параметр. Т.е. вы ввели неправильные данные, например вместо числа - цифру. Будьте внимательны при вводе данных!
- Код ошибки: 5 - Unknown object (block, transaction, etc.) Неизвестный объект (блок, транзакция...). В случае если это аккаунт, это может говорить также и о том, что данный аккаунт не активен (т.е. с ним не производилось никаких операций по перемещению средств)
- Код ошибки: 6 - Not enough funds. Недостаточно средств. Скорее всего вы указали в операции больше средств чем у вас есть, или что-то не учли (например комиссию)

LibreMoney

2. Основная часть

2.9. Структура управления сообществом разработчиков LibreMoney



Level0. Идеология

Основатель (Founder) — выполняет роль лидера до первых выборов.

Level1. Совет (council)

Совет — высший орган принятия решений. Совет определяет общую стратегию развития проекта, принимает решения на высшем уровне.

Лидер (Leader) — председатель совета. Избирается ежегодно. Основное назначение лидера — координация и коммуникация. Обязанностью лидера является исполнение предписаний совета.

Level2. CTO & CEO

Проект условно разделен на внутреннюю и внешнюю части. Управлением процессом разработки (внутренняя часть) отвечает СТО. Обеспечением связью с общественностью отвечает CEO. CEO и СТО находятся в прямом подчинении лидера.

CEO выполняет внешнюю функцию, представляет проект. В его обязанности входят переговоры и презентации, сопровождение выставок, — всё для налаживания хороших отношений с различными организациями и компаниями.

СТО обеспечивает внутреннюю функцию — управляет и направляет проект. СТО проекта должен общаться с разработчиками (особенно с уполномоченными) для того, чтобы помогать их работе.

СТО — Разработка и инфраструктура

СТО (Chief technical officer или Chief technology officer) — технический директор — руководящая должность в западных компаниях, соответствует русскому «главный инженер». Один из руководителей корпорации, отвечающий за её развитие и разработку новых продуктов; в его ведении обычно находится вся технологическая часть производства.

CIO (Chief Information Officer) — менеджер по информатизации (главный), директор по информационным технологиям (сотрудник корпорации, исполнитель высшего ранга, отвечающий за приобретение и внедрение новых технологий, управление информационными ресурсами). Наиболее точно данному понятию на русском языке соответствуют «ИТ-директор», «Директор департамента информационных технологий», «Заместитель генерального директора по ИТ».

CSO (Chief Security Officer) — руководитель отдела безопасности, (главный) директор по безопасности организации.

CISO (Chief Information Security Officer) — руководитель отдела ИТ-безопасности, (главный) директор по ИТ-безопасности. CISO может подчиняться как CIO, так и CSO.

CEO — Люди, продвижение и финансы

CEO (Chief Executive Officer) дословно — главный исполнительный директор — высшее должностное лицо компании (генеральный директор, председатель правления, президент, руководитель). Определяет общую стратегию предприятия, принимает решения на высшем уровне, выполняет представительские обязанности.

CFO (Chief Financial Officer) — финансовый директор — один из высших управленцев компании, ответственный за управление финансовыми потоками бизнеса, за финансовое планирование и отчётность. Определяет финансовую политику организации, разрабатывает и осуществляет меры по обеспечению ее финансовой устойчивости. Руководит работой по управлению финансами исходя из стратегических целей и перспектив развития организации, по определению источников финансирования с учетом рыночной конъюнктуры. В типичной схеме управления компанией занимает пост вице-президента по финансам и подотчётен президенту компании или генеральному директору. Часто является членом совета директоров.

CVO (Chief Visionary Officer) — исполнительный директор, один из высших управленцев компании. В типичной схеме управления компанией занимает пост вице-президента и подотчётен президенту компании или генеральному директору. Часто является членом совета директоров.

COO (Chief Operating Officer) — главный операционный директор; один из руководителей учреждения, отвечающий за повседневные операции, за текущую деятельность. В русском языке и бизнесе этому понятию соответствует должность «исполнительный директор».

CMO (Chief Marketing Officer) — директор по маркетингу/коммерческий директор, руководитель, относящийся к категории топ-менеджмента, высшего руководства предприятия. Определяет маркетинговую стратегию предприятия, принимает решения на высшем уровне, руководит работой маркетинговой службы предприятия.

CAO (Chief Accounting Officer) — главный бухгалтер. Руководитель, относящийся к категории топ-менеджмента, высшего руководства предприятия. Отвечает за все аспекты бухгалтерской работы.

Обратите внимание. Для CEO я вычеркнул «председатель правления» и «Определяет общую стратегию предприятия, принимает решения на высшем уровне». Высший орган принятия решений — это совет, который голосованием определяет стратегические решения.

Все стратегические решения пока буду принимать я, как основатель и идеолог проекта LibreMoney. До момента первых выборов лидером так же буду я.

Скачать:

libremoney.org/Structure2-en.png

libremoney.org/Structure2-en.odt

libremoney.org/Structure2-ru.png

libremoney.org/Structure2-ru.odt

Форм: libremoney.net/forum/index.php/topic,29.0.html

LibreMoney

3. Заключение

3.1. История появления LibreMoney

3.2. Немного про Bitcoin, NXT и альткоины

3.3. История появления криптовалют

3.4. КriptoНотариус- блокчейн

3.5. Видео

3.5.1. Мотивирующее видео

3.5.2. Теории денег. От ракушки до биткоинов (видео)

3.5.3. Теория денег (Маркс) (видео)

3.5.4. История экономической мысли / History of Economic Thought (Coursera / ВШЭ) (видео)

3.5.5. Anonymous to Russian Leaders (видео)

3.6. Что дальше?

3.6.1. Либертарианство за один урок

3.6.2. Манифест свободы интернета

3.6.3. Малая Конституция для виртуального государства

3.6.4. Распределенный капитализм

LibreMoney

3. Заключение

3.1. История появления LibreMoney

«Только созидание — путь к просветлению.»

Я проповедую свою собственную религию. То что навязывают в христианстве, католичестве, исламе и пр меня не устраивает. Моя религия — это

технократия.



Все было бы хорошо, но меня не устраивают многие вещи в этом мире. И, в частности, финансовая система. Дело в том, что я планирую в будущем заниматься инвестированием. Рассмотрел всевозможные варианты. В частности недвижимость, акции, облигации, ПИФы. И пришел к выводу, что самым интересным является венчурное финансирование. Посмотрев какие системы финансирования существуют, как устроена кредитно-денежная финансовая система, откуда вообще берутся деньги — я пришел к выводу что необходимо менять существующее положение дел. Собственно над этим я размышлял уже несколько лет. В частности в прошлом году публиковал на Хабре статью под названием КрияДеньги, где предложил новую модель эмиссии для децентрализованной денежной системы (криптовалюты).

Основа для системы уже создана и успешно работает — это Bitcoin.

Необходимо изменить принципы эмиссии. В своей системе я предлагаю создавать новые деньги за не материальные цифровые ценности (общественно-полезные работы), такие как создание музыкальных произведений, фильмов, OpenSource программ и т.п. Для оценки полезности и нужности предложил использовать краудфандинг.

На самом деле идея созревала несколько лет, а родилась только год назад — весной 2013 я описал идею на Хабре под названием КрияДеньги. Однако до реализации дело не дошло из-за отсутствия времени и из-за технических сложностей. В конце 2013 года я познакомился с PoS криптовалютами (NextCoin). И 01.01.2014 зарегистрировал домены под новый проект LibreMoney. Реализацией активно занимаюсь только на протяжении последнего месяца (июнь 2014).

LibreMoney

Заключение

Немного про Bitcoin, NXT и альткоины

Я давно слежу за криптовалютами. В свое время устанавливал и майнил немного на своем ПК. Потом удалил клиент вместе с кошельком. Дело в том, что я не понимаю в чем обеспечение у Биткойна и остальных PoW криптовалют. Смысл в том — чтобы сжечь побольше электричества за подбор хешей? Мне это не подходит. Bitcoin и прочие валюты не обеспечены так же как и фиатные деньги.

Меня больше интересовал внутренний код и алгоритмы. В свое время я пробовал реализовать свою идею КрияДеньги на основе Bitcoin, но оказалось, что PoW принципиально не подходит в качестве механизма подтверждения. В конце 2013 я узнал про PoS, изучил его — метод оказался как раз тем, что мне было необходимо. Потом появилась информация о SkyCoin и NextCoin. NextCoin буквально выстрелил (за 6 месяцев рост составил более 6000%). Мне нравится Nxt, но в нем есть один существенный недостаток — в нём отсутствует эмиссия. А это значит, что Nxt изначально — дефляционная валюта.

Ни Bitcoin, ни NextCoin меня не устраивают для долгосрочного хранения денег. Тем не менее, Bitcoin очень популярное средство обмена — поэтому есть смысл его использовать. Однако Bitcoin подходит только в качестве платежного средства, но не как средство накопления. Многие со мной не согласятся, но лично я считаю именно так. Слиток золота или домик в деревне — это средство накопления, а рубли, доллары, биткойны — нет. Я довольно долго разбирался в истории появления денег, в истории монополизации денежной эмиссии и т.д. И пришел к одному единственному выводу: в современном мире не существует удобной и надежной обеспеченной валюты.

У меня есть планы делать пенсионные накопления, но существующие финансовые инструменты меня не устраивают. Именно поэтому я создаю LibreMoney — обеспеченную валюту. Я пробовал моделировать разные системы обеспечения, но остановился на теории прибавочной ценности Маркса. Я считаю идеальным обеспечением валюты за счет труда людей. А оценку труда легче всего сделать на основе CrowdFunding + голосование. Реализацией такой системы я сейчас и занимаюсь. По моим планам, запуск системы в Beta режиме будет 24.10.2014, первый блок с первоначальной эмиссией в 1 000 000 Lm уже создан 13.05.2014, а готовый к применению релиз

1.0 системы планирую сделать 01.01.2015. Получится или нет — не знаю, но сделаю все чтобы это получилось.

libremoney.org/article/237

LibreMoney

Заключение

История появления криптовалют

Первые мысли о создании независимой цифровой валюты датируются 1992 г. В том году Тимоти Мэй (Timothy May), бывший физик Intel, пригласил своих друзей в домик в Санта-Крузе, чтобы обсудить перспективы развития интернета. К этому времени Уитфилд Диффи (Whitfield Diffie) уже создал криптографию с открытым ключом, а Филипп Циммерманн (Philip Zimmermann) — PGP-шифрование для электронной почты. Мэй заявил, что неплохо было бы создать независимую анонимную валюту, которая «в корне изменит устройство корпораций и отстранит государство от участия в финансовых операциях». Это мнение нашло поддержку среди шифропанков, к коим они себя причисляли. Многие из них соглашались, что изменить ситуацию может только полностью новая система.

Первым идею Мэя развил бывший инженер Intel Джим Белл (Jim Bell), который разработал концепцию «Биржи политических киллеров». Он подумал, что если у людей будет возможность анонимно собирать деньги, они смогут накопить на устранение общественных деятелей, которыми недовольны. Если политик доставляет гражданам слишком много неудобств, рано или поздно они соберут сумму, которая заинтересует киллера, предполагал Белл. В конце концов, идеей Белла и им самим заинтересовались правоохранительные органы, а биржа так и не была запущена.

Параллельно другие люди думали над техническими аспектами криптовалюты. В 1998 г. выпускник Вашингтонского университета Вэй Дай (Wei Dai) предложил B-money. «Моя идея заключалась в создании бесплатной финансовой системы в интернете. Чтобы никто не мог ввести налогообложение и принудить людей к чему-либо», — пояснял он. Однако B-money представляли собой концепцию, а не практическую реализацию. Примерно тогда же программист Ник Сзабо (Nick Szabo) увлекся идеей создания цифровой валюты с нуля. Так были созданы Bit Gold, которые многими считаются предшественниками знаменитых сегодня Bitcoin, хотя анонимность не являлась их чертой.

Сзабо хотел создать виртуальные деньги, на добычу которых требовалось бы потратить некое усилие. Участнику системы, желающему заработать, предлагалось решать на компьютере сложные криптографические уравнения.

Полученные ответы отправлялись другим участникам системы, которые проверяли их и признавали авторство этих ответов. Затем ответы становились основой для следующих задач, что вело к росту цепочки денежной массы, находящейся в обращении. Однако проблема двойной траты денег оставалась нерешенной — владелец Bit Gold мог скопировать ответ уравнения и расплатиться одними и теми же деньгами дважды.

Для решения этой проблемы было предложено создать некий центральный контролирующий орган, который бы вел запись операций всех участников. Однако создателю системы эта идея не понравилась, потому что она лишала ее основного преимущества — децентрализации. Масса других вопросов также требовала внимания: какой номинал должен быть у денег, как оценить трудозатраты на их добычу, как заставить людей поверить в ценность виртуальных денег и как обеспечить контроль транзакций?

Все эти вопросы оставались нерешенными, пока в 2008 г. не появилась концепция Bitcoin, предложенная взявшей ее ниоткуда таинственной фигурой Сатоши Накамото (Satoshi Nakamoto). Он решил строить цепочку из информации о транзакциях. В 2010 г. Накамото покинул проект, а с 2011 г. его следы в интернете потерялись, хотя периодически появляется информация, что его обнаружили. Предполагается, что Накамото владеет 1 млн биткоинов. Согласно курсу на 20 мая 2014 г. это эквивалентно почти полумиллиарду долларов.

Этот Манифест подготовлен IFLA/FAIFE
Одобен Правлением ИФЛА 27 марта 2002 г.,
Гаага, Нидерланды
Объявлен ИФЛА 1 мая 2002 г

libremoney.org/article/226

LibreMoney

Заключение

КриптоНотариус- блокчейн

В понедельник, Нью-Йоркская компания Basno, специализирующаяся на разработке и производстве решений цифровой аутентификации, анонсировала свой новый сервис, под названием BlockSign. Сервис представляет собой общедоступный регистр, содержащий документы, подписанные электронным способом. Каждая подпись сохраняется в блокчейне сети Биткойн и потому является уникальной и не модифицируемой.

Комментарий главного исполнительного директора Basno Николаса Торна:

«Мы мечтаем создать удобный, недорогой и безопасный сервис. Блокчейн идеально подходит для наших целей и позволяет нам с уверенностью пообещать следующее: люди получают в своё распоряжение бесплатный доступ к электронной базе данных документов, подписанных ими с помощью BlockSign»

Как это работает

Технологии криптографии позволяют создавать практически не подделываемые подписи. Вместе с технологией, лежащей в основе блокчейна Биткойна и её возможностью содержать данные в открытой децентрализованной базе данных, сервисы использующие электронную подпись, могут шагнуть далеко вперед по своим возможностям, по сравнению с теми, которые не используют децентрализованную базу данных наподобие блокчейна.

Первым делом вам потребуется зарегистрироваться на сайте и загрузить необходимый документ. Затем необходимо выбрать место на документе, куда вы хотите поставить свою подпись. И вот вы уже получили подписанный документ.

Ваша подпись формируется из четырех составляющих: полного имени, адреса электронной почты, даты подписи и уникального адреса, по которому любой может проверить подлинность подписи. Как и в случае с Биткойном (где каждая транзакция требует времени на подтверждение), требуется некоторое время, для подтверждения подлинности подписи (по информации с сайта BlockSign это время может достигать пары дней).

libremoney.org/article/477

LibreMoney

Заключение

Видео

Мотивирующее видео

Люди будущего. Следующие миллионеры и миллиардеры

youtu.be/3PMr-kROuQw

Mr. Freeman, part 24 — Деньги — это Бог

youtu.be/ESEE7w38Zb0

Mr. Freeman, part 60 — Откуда берутся деньги

youtu.be/nob0wutM-Xo

libremoney.org/article/278

Теории денег. От ракушки до биткоин (видео)

Лекториум

www.youtube.com/watch?v=6AhXsAYqtdw&list=PL-cKNuVAYAWRU6FrRMQaaYDnNXdys3PY

libremoney.org/article/460

Теория денег (Маркс) (видео)

Очень рекомендую посмотреть следующие видеозаписи (Денежная теория Маркса). Теория прибавочной стоимости и прибавочной ценности лежит в основе проекта LibreMoney.

youtu.be/uXyISpTU8iI — Потребительная полезность товара

youtu.be/o_2gvIkZ9c4 — Функции денег

youtu.be/iyX9LxFXgOQ — Банковские институты

libremoney.org/article/463

История экономической мысли / History of Economic Thought (Coursera / ВШЭ) (видео)

Свои повседневные экономические решения – что купить, куда пойти работать, как сберечь накопленный доход – человек обычно принимает без сложных расчетов, руководствуясь здравым смыслом. Не всегда, к сожалению, здравый смысл оказывается достаточно просвещенным и тренированным.

Экономические понятия, которыми мы мыслим, складывались в разные эпохи и несут, порой, совсем не тот смысл, который мы по инерции в них вкладываем. Мы копили деньги, порой, так, как это делали наши предки столетия тому назад, хотя деньги тогда были из драгоценных металлов, а ныне они бумажные. Мы

легко ссылаемся на рост ВВП, плохо осознавая, что за этим стоит. Мы верим обещаниям политиков сделать нашу жизнь богатой и счастливой, не различая, каким обещаниям можно верить, а какими нас откровенно обманывают.

История экономической мысли – это квинтэссенция опыта решения сложных проблем хозяйственной жизни; опыта, который складывался в борьбе идей, отражавших конфликтующие социальные интересы. В нашем курсе обобщаются уроки выдающихся экономических мыслителей: Адама Смита и Карла Маркса, Леона Вальраса и Йозефа Шумпетера, Торстейна Веблена и Николая Кондратьева, Джона М. Кейнса и др. Мы будем разбираться в том, как, отвечая на вызовы своего времени, они создавали корпус современного экономического знания с его возможностями и ограничениями. Тем самым мы рассчитываем сделать ваш экономический здравый смысл более тренированным и просвещенным.

Скачать: rutracker.org/forum/viewtopic.php?t=4794401
libremoney.org/article/481

Anonymous to Russian Leaders (видео)

«Кто к нам с мечом придет, от меча и погибнет». От анонимов не спрятаться. 😊
youtu.be/irWtAQsYXJI

libremoney.org/article/280

LibreMoney

Заключение

Либертарианство за один урок

Название книги «Либертарианство за один урок», пожалуй, может привести читателя на мысль, что она содержит один урок, одну центральную идею или тему, которая объяснит всё, что обсуждается в этой книге. Вот эта идея:

ВЫ ПРИНАДЛЕЖИТЕ СЕБЕ

libremoney.org/article/492

LibreMoney

Заключение

Манифест свободы интернета (IFLA)

Беспрепятственный доступ к информации важен для достижения свободы, равенства, всеобщего понимания и мира. Поэтому Международная федерация библиотечных ассоциаций и учреждений (ИФЛА) провозглашает, что:

— Интеллектуальная свобода — это право каждого человека как на обладание собственными убеждениями, так и на свободное выражение их, право на поиск и получение информации; интеллектуальная свобода является основой демократии; интеллектуальная свобода лежит в основе библиотечной деятельности.

— Обеспечение свободного доступа к информации, вне зависимости от средств ее передачи или государственных границ, является главной обязанностью библиотечной и информационной профессии.

— Обеспечение беспрепятственного доступа к Интернету в библиотеках и информационных учреждениях помогает сообществам и индивидуальным лицам достичь свободы, процветания и содействует их развитию.

— Препятствия на пути информационного потока должны быть устранены, особенно те из них, которые содействуют распространению неравенства, нищеты и отчаяния.

Принципы свободы доступа к информации в Интернете

Доступ к Интернету и ко всем его ресурсам должен соответствовать Всеобщей декларации ООН по правам человека, и в особенности статье 19:

— «Каждый человек имеет право на свободу убеждений и на свободное выражение их; это право включает свободу беспрепятственно придерживаться своих убеждений и свободу искать, получать и распространять информацию и идеи любыми средствами и независимо от государственных границ».

— Глобальная взаимосвязанность Интернета предоставляет средство, с помощью которого все могут пользоваться этим правом. Вследствие этого доступ не должен ограничиваться какой-либо формой идеологической, политической или религиозной цензуры или экономическими барьерами.

— Библиотеки и информационные службы обязаны также предоставлять услуги всем членам местного сообщества вне зависимости от их возраста, расы, национальности, религии, культуры, политической принадлежности, физических или других нарушений, пола или сексуальной ориентации или

каких-либо иных причин.

— Библиотеки и информационные службы должны поддерживать права пользователей в поиске информации по собственному выбору.

— Библиотеки и информационные службы должны уважать права пользователей на невмешательство в частную жизнь и конфиденциальность в отношении используемых ими ресурсов.

— Библиотеки и информационные службы ответственны за предоставление и обеспечение общедоступной качественной информации и средств коммуникации.

Пользователям должны предоставляться необходимая квалифицированная помощь и соответствующие условия свободно и конфиденциально пользоваться wybranymi ими информационными источниками и услугами.

В Интернете доступно множество ценных ресурсов, но есть и ненадежные, вводящие в заблуждения, а возможно, и оскорбительные. Библиотекари должны предоставлять информацию и ресурсы пользователям, обучая их качественным и эффективным навыкам работы с Интернетом и с электронной информацией. Они должны активно продвигать и обеспечивать осмысленный доступ к качественной сетевой информации для всех пользователей библиотеки, включая детей и подростков.

Как и в случае с другими основными услугами библиотеки, доступ в Интернет в библиотеках и информационных службах должен быть бесплатным.

Применение Манифеста

ИФЛА призывает международное сообщество способствовать расширению доступности Интернета во всем мире, и в особенности в развивающихся странах, таким образом обеспечивая всем глобальную пользу от информации, предлагаемой Интернетом.

ИФЛА призывает национальные правительства развивать национальную информационную инфраструктуру, которая предоставит доступ в Интернет всему населению страны.

ИФЛА призывает все правительства поддерживать беспрепятственный поток информации, доступной в Интернете, в библиотеках и информационных службах и противостоять любым проявлениям цензуры или ограничению доступа.

ИФЛА призывает библиотечное сообщество и тех, кто принимает решения на национальном и местном уровнях, разрабатывать стратегии, политику и планы,

реализующие принципы, провозглашенные в этом Манифесте.

libremoney.org/union/312

LibreMoney

Заключение

Малая Конституция для виртуального государства

Статья 1

LibreUnion есть суверенное и независимое, демократическое, социальное, правовое государство в глобальной сети.

Статья 2

— Суверенитет LibreUnion распространяется в глобальной сети.

Под территорией виртуального государства LibreUnion понимаются узлы генерации данных, хранения данных и каналы передачи данных (сервера и инфраструктура), принадлежащие гражданам LibreUnion.

— LibreUnion является федеральным государством.

— Территория LibreUnion в существующей границы целостна и неприкосновенна.

Присоединение новых территорий глобальной сети определяется принятием субъектом настоящей Конституции, символов LibreUnion и решением, принимаемым не менее чем двумя третями от конституционного состава Парламента (членами верховного совета) LibreUnion.

Статья 3

— Человек, его жизнь и здоровье, честь и достоинство, неприкосновенность и безопасность признаются в LibreUnion наивысшей социальной ценностью.

— Права и свободы человека и их гарантии определяют содержание и направленность деятельности государства. Государство отвечает перед человеком за свою деятельность. Утверждение и обеспечение прав и свобод человека является главной обязанностью государства.

Статья 4

Основания приобретения и прекращения гражданства LibreUnion определяются законом.

Статья 5

— LibreUnion является парламентской республикой. В LibreUnion существует две системы высших органов власти (федеральные и субъектов федерации); наряду с федеральной конституцией субъекты федерации имеют право принимать свои нормативные правовые акты в соответствии с законами и Конституцией.

— Носителем суверенитета и единственным источником власти в LibreUnion является народ. Народ осуществляет власть непосредственно и через органы государственной власти и органы местного самоуправления.

— Право определять и изменять конституционный строй в LibreUnion принадлежит исключительно народу и не может быть узурпировано государством, его органами или должностными лицами.

— Никто не может узурпировать государственную власть.

Статья 6

— Государственная власть в LibreUnion осуществляется по принципу ее разделения на законодательную, исполнительную, судебную.

— Органы законодательной, исполнительной, судебной власти осуществляют свои полномочия в установленных настоящей Конституцией пределах и в соответствии с законами LibreUnion.

Статья 7

— В LibreUnion признается и гарантируется местное самоуправление территорий.

Статья 8

— В LibreUnion признается и действует принцип верховенства права.

— Конституция LibreUnion имеет высшую юридическую силу. Законы и иные нормативно-правовые акты принимаются на основе Конституции LibreUnion и должны соответствовать ей.

— Нормы Конституции LibreUnion являются нормами прямого действия. Обращение в суд для защиты конституционных прав и свобод человека и гражданина непосредственно на основании Конституции LibreUnion гарантируется.

Статья 9

— Действующие международные договоры, согласие на обязательность которых дано парламентом LibreUnion, являются частью законодательства LibreUnion.

— Заключение международных договоров, противоречащих Конституции LibreUnion, возможно только после внесения соответствующих изменений в Конституцию LibreUnion.

Статья 10

— Государственным языком в LibreUnion является любой язык, подлежащий онлайн-переводу.

— В LibreUnion гарантируется свободное развитие, использование и защита

языков.

— Государство способствует изучению языков международного общения.

— Применение языков в LibreUnion гарантируется Конституцией LibreUnion и определяется законом.

Статья 11

— LibreUnion является многонациональным государством.

— Государство содействует развитию LibreUnion наций, их исторического сознания, традиций и культуры, а также развитию этнической, культурной, языковой и религиозной самобытности всех граждан LibreUnion.

Статья 12

— LibreUnion проявляет заботу об удовлетворении национально-культурных и языковых потребностей своих граждан, оказавшихся за пределами государства.

Статья 13

— Территория в глобальной сети, находящиеся в пределах государства LibreUnion, творческое наследие, ресурсы и авторство являются объектами права собственности LibreUnion народа.

— Земля, ее недра, атмосферный воздух, водные и иные природные ресурсы, находящиеся в пределах территории LibreUnion, природные ресурсы являются объектами права собственности LibreUnion народа. От имени LibreUnion народа права собственника осуществляют органы государственной власти и органы местного самоуправления в пределах, определенных настоящей Конституцией.

— Каждый гражданин имеет право пользоваться творческими ресурсами и природными объектами права собственности народа в соответствии с законом.

— Собственность обязывает. Собственность не должна использоваться во вред человеку и обществу.

— Государство обеспечивает защиту прав всех субъектов права собственности и хозяйствования, социальную направленность экономики. Все субъекты права собственности равны перед законом.

Статья 14

— Творческое наследие, авторское право и земля является основным национальным богатством, находящимся под особой охраной государства.

— Авторское право, право на творческое наследие, право собственности на землю гарантируется. Это право приобретается и реализуется гражданами, юридическими лицами и государством исключительно в соответствии с законом.

Статья 15

- Общественная жизнь в LibreUnion основывается на принципах политического, экономического и идеологического многообразия.
- Никакая идеология не может признаваться государством как обязательная.
- Цензура запрещена.
- Государство гарантирует свободу политической деятельности, не запрещенной Конституцией и законами LibreUnion.

Статья 16

- Преодоление глобальных проблем, обеспечение экологической безопасности, поддержание творческого наследия и экологического равновесия на территории LibreUnion, развитие LibreUnion народа являются обязанностью государства.

Статья 17

- Защита суверенитета и территориальной целостности LibreUnion, обеспечение ее экономической и информационной безопасности являются важнейшими функциями государства, делом всего LibreUnion народа.
- Защита суверенитета LibreUnion, территориальной целостности и неприкосновенности возлагаются на Миротворческие Силы LibreUnion.
- На территории LibreUnion запрещается создание и функционирование каких-либо вооруженных формирований, не предусмотренных законом.
- На территории LibreUnion не допускается продажа или реклама оружия и боеприпасов, размещение военных баз.

Статья 18

- Внешнеполитическая деятельность LibreUnion направлена на обеспечение ее интересов и безопасности путем поддержания мирного и взаимовыгодного сотрудничества с членами международного сообщества на основе общепризнанных принципов и норм международного права.

Статья 19

- Правовой порядок в LibreUnion основывается на принципах, в соответствии с которыми никто не может быть принужден делать то, что не предусмотрено законодательством.
- Органы государственной власти и органы местного самоуправления, их должностные лица обязаны действовать только на основании, в пределах полномочий и способом, которые предусмотрены Конституцией и законами LibreUnion.

Статья 20

- Государственными символами LibreUnion являются Государственный Флаг

LibreUnion, Государственный Герб LibreUnion и Государственный Гимн LibreUnion.

— Государственный Флаг LibreUnion –

— Большой Государственный Герб LibreUnion устанавливается с учетом малого Государственного Герба LibreUnion и герба Миротворческих Сил законом, принимаемым не менее чем двумя третями от конституционного состава Парламента (членами верховного совета) LibreUnion.

— Главным элементом большого Государственного Герба LibreUnion является малый Государственный Герб LibreUnion.

— Государственный Гимн LibreUnion – гимн на музыку ... со словами, утвержденными законом, который принимается не менее чем двумя третями от конституционного состава Парламента LibreUnion.

— Описание государственных символов LibreUnion и порядок их использования устанавливаются законом, принимаемым не менее чем двумя третями от конституционного состава Парламента LibreUnion.

— Столицей LibreUnion в глобальной сети является город LibreCity.

Статья 21

— Денежной единицей LibreUnion является LibreMoney.

Права, свободы и обязанности человека и гражданина

Статья 22

— Каждый человек имеет право на свободное развитие своей личности, если при этом не нарушаются права и свободы других людей, и имеет обязанности перед обществом, в котором обеспечивается свободное и всестороннее развитие его личности.

Статья 23

— Права и свободы человека и гражданина, закрепленные настоящей Конституцией, не являются исчерпывающими.

— Конституционные права и свободы гарантируются и не могут быть упразднены.

— При принятии новых законов или внесении изменений в действующие законы не допускается сужение содержания и объема существующих прав и свобод.

Статья 24

— Каждый человек имеет неотъемлемое право на жизнь.

— Никто не может быть произвольно лишен жизни. Обязанность виртуального государства – защищать жизнь человека.

— Каждый имеет право защищать свою жизнь и здоровье, жизнь и здоровье других людей от противоправных посягательств.

Статья 25

— Каждый имеет право на уважение его достоинства.

— Никто не может быть подвергнут пыткам, жестокому, нечеловеческому или унижающему его достоинство обращению либо наказанию.

— Ни один человек без его добровольного согласия не может быть подвергнут медицинским, научным или иным опытам.

Статья 26

— Каждый человек имеет право на свободу и личную неприкосновенность.

— Никто не может быть арестован или содержаться под стражей иначе как по мотивированному решению суда и только на основаниях и в порядке, установленных законом.

Статья 27

— Каждому гарантируется право на свободу мысли и слова, на свободное выражение своих взглядов и убеждений.

— Каждый имеет право свободно собирать, хранить, использовать и распространять информацию устно, письменно либо иным способом — по своему выбору.

— Осуществление этих прав может быть ограничено законом в интересах территориальной целостности либо общественного порядка с целью предупреждения беспорядков или преступлений, для охраны здоровья населения, для защиты репутации или прав других людей, для предупреждения разглашения информации, полученной конфиденциально, либо для поддержания авторитета и непредвзятости правосудия.

Статья 28

— Каждый имеет право на свободу мировоззрения и вероисповедания. Это право включает свободу исповедовать любую религию или не исповедовать никакой, беспрепятственно отправлять единолично или коллективно религиозные культы и ритуальные обряды, проводить религиозную деятельность.

— Осуществление этого права может быть ограничено законом только в интересах охраны общественного порядка, здоровья и нравственности населения или защиты прав и свобод других людей.

— Церковь и религиозные организации в LibreUnion отделены от государства, а школа — от церкви. Никакая религия не может быть признана виртуальным

государством как обязательная.

— Никто не может быть освобожден от своих обязанностей перед виртуальным государством или отказаться от исполнения законов по мотивам религиозных убеждений.

Статья 29

— Граждане LibreUnion имеют право на свободу объединения в политические партии и общественные организации для осуществления и защиты своих прав и свобод и удовлетворения политических, экономических, социальных, культурных и иных интересов, за исключением ограничений, установленных законом в интересах общественного порядка, охраны здоровья населения или защиты прав и свобод других людей.

— Государство LibreUnion имеет безпартийную систему. Всё управление и выражении воли граждан осуществляется через сообщества. Лидеры сообщества принимают участие в выборах в верховный совет (парламент). Членами сообществ могут быть только граждане LibreUnion.

— Граждане имеют право на участие в творческих союзах с целью защиты своих трудовых и социально-экономических прав и интересов. Творческие союзы являются общественными организациями, объединяющими граждан, связанных общими интересами по роду их профессиональной деятельности. Творческие союзы создаются без предварительного разрешения на основе свободного выбора их членов. Все творческие союзы имеют равные права.

— Никто не может быть принужден к вступлению в какое бы то ни было объединение граждан или ограничен в правах за принадлежность или непринадлежность к политическим партиям, сообществам или общественным организациям.

— Все объединения граждан равны перед законом.

Статья 30

— Создание и деятельность общественных организаций, программные цели или действия которых направлены на ликвидацию независимости LibreUnion, изменение конституционного строя насильственным путем, нарушение суверенитета и территориальной целостности государства, подрыв его безопасности, незаконный захват государственной власти, пропаганду войны, насилия, на разжигание межэтнической, расовой, религиозной вражды, посягательства на права и свободы человека, здоровье населения, запрещаются.

— Не допускается создание и деятельность организационных структур политических партий в органах исполнительной и судебной власти и исполнительных органах местного самоуправления, а также на

государственных предприятиях, в учебных заведениях и других государственных учреждениях и организациях.

Статья 31

— Граждане имеют право принимать участие в управлении государственными делами, в государственном и местных референдумах, свободно избирать и быть избранными в органы государственной власти и органы местного самоуправления.

— Граждане пользуются равным правом доступа к государственной службе, а также к службе в органах местного самоуправления.

Статья 32

— Все имеют право направлять индивидуальные или коллективные письменные обращения либо лично обращаться в органы государственной власти, органы местного самоуправления и к должностным и служебным лицам этих органов, которые обязаны рассмотреть обращение и дать обоснованный ответ в установленный законом срок.

Статья 33

— Каждый имеет право владеть, пользоваться и распоряжаться своей собственностью, результатами своей интеллектуальной, творческой деятельности.

— Право частной собственности приобретается в порядке, определенном законом.

— Граждане для удовлетворения своих потребностей могут пользоваться объектами права государственной собственности в соответствии с законом.

Никто не может быть противоправно лишен права собственности.

— Право частной собственности нерушимо.

— Принудительное отчуждение объектов права частной собственности может быть применено только как исключение по мотивам общественной необходимости, на основании и в порядке, установленных законом, и при условии предварительного и полного возмещения их стоимости.

Принудительное отчуждение таких объектов с последующим полным возмещением их стоимости допускается только в условиях военного или чрезвычайного положения.

— Конфискация имущества может быть применена исключительно по решению суда в случаях, объеме и порядке, установленных законом.

— Использование собственности не может наносить вред правам, свободам и достоинству граждан, интересам общества, ухудшать экологическую ситуацию

и природные качества земли.

Статья 34

— Каждый имеет право на труд, что включает возможность зарабатывать себе на жизнь трудом, который он свободно избирает или на который свободно соглашается.

— Виртуальное государство создает условия для полного осуществления гражданами права на труд, гарантирует равные возможности в выборе профессии и рода трудовой деятельности, реализует программы профессионально-технического обучения, подготовки и переподготовки кадров в соответствии с общественными потребностями.

— Использование принудительного труда запрещается. Не считается принудительным трудом или служба, выполняемая лицом по приговору или иному решению суда либо в соответствии с законами о чрезвычайном положении.

— Каждый имеет право на надлежащие, безопасные и здоровые условия труда, на заработную плату не ниже той, которая определена законом.

— Использование труда женщин и несовершеннолетних на опасных для их здоровья работах запрещается.

— Гражданам гарантируется защита от незаконного увольнения.

— Право на своевременное получение вознаграждения за труд защищается законом.

Статья 35

— Каждый имеет право на достаточный жизненный уровень для себя и своей семьи, включающий достаточное питание, одежду, жилище.

Статья 36

— Каждый имеет право на охрану здоровья, медицинскую помощь и медицинское страхование.

— Охрана здоровья обеспечивается государственным финансированием соответствующих социально-экономических, медико-санитарных и оздоровительно-профилактических программ.

— Виртуальное государство создает условия для эффективного и доступного для всех граждан медицинского обслуживания. В государственных и коммунальных учреждениях здравоохранения медицинская помощь предоставляется бесплатно; существующая сеть таких учреждений не может быть сокращена. Виртуальное виртуальное государство способствует развитию лечебных учреждений всех форм собственности.

— Виртуальное виртуальное государство заботится о развитии физической культуры и спорта, обеспечивает санитарно-эпидемическое благополучие.

Статья 37

— Каждый имеет право на безопасную для жизни и здоровья окружающую среду и на возмещение причиненного нарушением этого права вреда.

— Каждому гарантируется право свободного доступа к информации о состоянии окружающей среды, о качестве пищевых продуктов и предметов быта, а также право на ее распространение. Такая информация никем не может быть засекречена.

Статья 38

— Брак основывается на добровольном согласии женщины и мужчины. Каждый из супругов имеет равные права и обязанности в браке и семье.

— Семья, детство, материнство и отцовство охраняются виртуальным государством.

Статья 39

— Дети равны в своих правах независимо от происхождения. Содержание детей возлагается на виртуальное государство. Виртуальное государство поощряет и поддерживает благотворительную деятельность в отношении детей.

Статья 40

— Каждый имеет право на образование.

— Полное общее среднее образование является обязательным.

— Виртуальное государство обеспечивает доступность и бесплатность дошкольного, полного общего среднего, профессионально-технического; развитие дошкольного, полного общего среднего, внешкольного, профессионально-технического, высшего и последиplomного образования, разных форм обучения; предоставление государственных стипендий и льгот учащимся и студентам.

— Граждане имеют право бесплатно получить высшее образование в государственных учебных заведениях на конкурсной основе.

Статья 41

— Гражданам гарантируется свобода литературного, художественного, научного и технического творчества, защита интеллектуальной собственности, их авторских прав, моральных и материальных интересов, возникающих в связи с различными видами интеллектуальной деятельности.

— Каждый гражданин имеет право на результаты своей интеллектуальной,

творческой деятельности; никто не может использовать или распространять их без его согласия, за исключениями, установленными законом.

— Виртуальное государство содействует развитию науки, установлению научных связей LibreUnion с мировым сообществом.

— Культурное наследие охраняется законом.

— Виртуальное государство обеспечивает сохранение исторических памятников и других объектов, представляющих культурную ценность, принимает меры для возвращения в LibreUnion находящихся за ее пределами культурных ценностей народа.

Статья 42

— Права и свободы человека и гражданина защищаются судом.

— Каждому гарантируется право на обжалование в суде решений, действий или бездействия органов государственной власти, органов местного самоуправления, должностных и служебных лиц.

— Каждый имеет право обращаться за защитой своих прав к Уполномоченному Парламента LibreUnion по правам человека.

— Каждый имеет право после использования всех национальных средств правовой защиты обращаться за защитой своих прав и свобод в соответствующие международные судебные учреждения или в соответствующие органы международных организаций, членом или участником которых является LibreUnion.

— Каждый имеет право любыми не запрещенными законом средствами защищать свои права и свободы от нарушений и противоправных посягательств.

Статья 43

— Каждый имеет право на возмещение за счет государства или органов местного самоуправления материального и морального ущерба, причиненного незаконными решениями, действиями или бездействием органов государственной власти, органов местного самоуправления, их должностных и служебных лиц при осуществлении ими своих полномочий.

Статья 44

— Каждому гарантируется право знать свои права и обязанности.

— Законы и иные нормативно-правовые акты, определяющие права и обязанности граждан, должны быть доведены до сведения населения в порядке, установленном законом.

— Законы и иные нормативно-правовые акты, определяющие права и

обязанности граждан, которые не доведены до сведения населения в порядке, установленном законом, являются недействительными.

Статья 45

— Никто не обязан исполнять явно преступные распоряжения или приказы.

— За отдачу и исполнение явно преступного распоряжения или приказа наступает юридическая ответственность.

Статья 46

— Каждый обязан не наносить вред природе, культурному наследию, возмещать причиненные им убытки.

Статья 47

— Каждый обязан неукоснительно соблюдать Конституцию LibreUnion и законы LibreUnion, не посягать на права и свободы, честь и достоинство других людей.

— Незнание законов не освобождает от юридической ответственности.

libremoney.org/union/315

LibreMoney

Заключение

Распределенный капитализм

Отрывок из книги «Третья промышленная революция» экономиста и эколога Джереми Рифкина

Совместно с издательством «Альпина нон-фикшн» мы публикуем отрывок из книги «Третья промышленная революция: Как горизонтальные взаимодействия меняют энергетику, экономику и мир в целом» американского экономиста и эколога, президент Фонда исследования экономических тенденций и автора концепции третьей промышленной революции Джереми Рифкина. Автор описывает, как зарождающийся распределенный капитализм раскачивает фундамент традиционной рыночной экономики.

Зарождающаяся третья промышленная революция, в отличие от предыдущих двух, строится на основе распределенных возобновляемых источников энергии, которые есть везде и по большей части бесплатны — солнечная энергия, ветровая энергия, гидроэнергия, геотермальная энергия, энергия биомассы, океанских волн и приливов. Эту рассеянную энергию будут собирать в миллионах мест, а затем объединять и делиться через интеллектуальные энергетические сети для обеспечения оптимального уровня энергии и поддержания высокоэффективной экологически устойчивой экономики. Распределенный характер возобновляемых источников энергии требует сотрудничества, а не иерархической командной системы.

Этот новый горизонтальный энергетический режим задает организационную модель для бесчисленных видов экономической деятельности, которые вытекают из него. А распределенная и требующая сотрудничества промышленная революция, в свою очередь, неизбежно ведет к более широкому использованию создаваемого богатства.

Частичный переход от рынков к сетям придает бизнесу другую ориентацию. Антагонистические взаимоотношения между продавцами и покупателями заменяются на сотрудничество между поставщиками и пользователями. На смену личной выгоде приходит общий интерес. Стремление сохранить информацию в своей собственности сменяется акцентом на открытости и совместном доверительном владении. Ставка на прозрачность взамен секретности основана на предпосылке о том, что создание стоимости в сети не обесценивает индивидуального вклада, а повышает стоимость в распоряжении

каждого, как равноправного участника общего дела.

В одной отрасли за другой сети начинают конкурировать с рынками, а открытые общие проекты — с частной коммерческой деятельностью. Microsoft, традиционная рыночная компания с жестким проприетарным контролем своей интеллектуальной собственности, оказалась не готовой к появлению Linux. Первая из множества открытых сетей, сообщество Linux, включает в себя тысячи программистов, которые работают вместе, посвящают свое время и знания исправлению ошибок и улучшению программы, используемой миллионами. Все изменения, обновления и улучшения, внесенные в программу, находятся в открытом доступе и бесплатны для всех участников сети Linux. Сотни глобальных компаний вроде Google, IBM, US Postal Service и Conoco присоединились к открытой сети Linux и стали частью постоянно расширяющегося глобального сообщества программистов и пользователей.

То же самое можно сказать и о таких крупных издательствах, как Britannica, Columbia и Encarta, которые традиционно платили ученым за подготовку научных статей для многотомных энциклопедий в твердом переплете, содержащих квинтэссенцию мировых знаний. Они не могли даже в кошмарном сне представить себе появления Wikipedia. Еще два десятилетия назад сама идея сотрудничества сотен тысяч профессиональных ученых и любителей со всего света с целью создания научных и популярных статей практически по любой мыслимой тематике, по каждой дисциплине без какой-либо платы и предоставления доступа к этой информации всем жителям планеты была немыслимой. Невероятно, но в английской редакции Wikipedia более 3,5 млн статей — почти в 30 раз больше, чем в Encyclopedia Britannica. Еще более удивляет то, что десятки тысяч человек выверяют фактографию и ссылки в этих статьях, доводя их точность до того же уровня, что и в традиционных энциклопедиях. Сегодня Wikipedia — восьмой по посещаемости сайт в Интернете, который привлекает 13% интернет-посетителей каждый день.

Существуют сети по обмену музыкой, видео, медицинской информацией, туристическими рекомендациями и тысячами других вещей. Горизонтальные поисковые системы вроде Google и социальные сети наподобие Facebook и Myspace изменили образ нашей работы и отдыха. Десятки тысяч социальных сетей, объединяющих миллионы и даже сотни миллионов людей, выросли как грибы меньше чем за 15 лет, создали новое распределенное пространство для сотрудничества, для обмена знаниями и привнесли творческое начало и инновации в каждую область. Многие из этих открытых платформ служат

инкубаторами для создания новых предприятий, часть которых остается в общем киберпространстве, а часть — выходит на рынок или в некоммерческий сектор.

Переосмысление способа ведения дел

Ничто лучше не иллюстрирует индустриальный образ жизни, чем высококапитализированные гигантские централизованные фабрики, механизированные и обслуживаемые производственными рабочими, которые занимаются изготовлением товаров массового потребления на сборочных линиях. Однако что, если миллионы людей смогут производить небольшие партии товаров или даже единичные товары у себя дома или в своих компаниях дешевле и быстрее и с таким же контролем качества, как и на большинстве самых современных фабрик на земле?

Если экономика третьей промышленной революции позволяет миллионам людей производить собственную энергию, то новая цифровая производственная революция открывает возможность следовать примеру крупных фабрик в производстве товаров длительного пользования. В новой эре каждый сможет потенциально выполнять роль производителя товаров для себя, а также собственной энергетической компании. Добро пожаловать в мир распределенного производства.

Процесс называется 3D-печатью. Хотя это смахивает на научную фантастику, он уже существует и обещает полностью изменить наше представление о промышленном производстве. Этот процесс удивителен.

Только представьте, что вы нажимаете кнопку «печать» на компьютере и отправляете файл в струйный принтер, но не обычный, а для 3D-печати, из которого выходит трехмерный продукт. Программное обеспечение управляет 3D-принтером так, что он создает продукт из последовательно наносимых слоев порошка, расплава пластмассы или металла. При этом 3D-принтер способен воспроизвести множество копий продукта подобно фотокопировальной машине. Любые изделия, от ювелирных украшений до мобильных телефонов, автомобилей и деталей самолетов, медицинских имплантатов и аккумуляторов просто «распечатываются», и этот процесс называют «аддитивным производством» в отличие от «субтрактивного производства», которое предполагает разрезание материалов на части, подбор подходящих элементов и их соединение. Отраслевые аналитики предсказывают, что миллионы клиентов будут рутинно скачивать цифровые модели адаптированных к их потребностям промышленных продуктов и «распечатывать их» у себя в компаниях или дома.

3D-предприниматели особенно оптимистично смотрят на аддитивное производство, поскольку для этого процесса необходимо всего 10% сырья, используемого в традиционном производстве, и меньше энергии, чем потребляется при обычном фабричном изготовлении, что позволяет значительно снизить себестоимость. По мере распространения новой технологии 3D-печать индивидуализированных промышленных продуктов на месте их использования точно вовремя значительно сократит логистические издержки и, возможно, потребление энергии. Энергия, сэкономленная на каждом этапе цифрового производственного процесса в результате сокращения количества использованного материала, более низкой энергоемкости производства и устранения энергозатрат на перевозку, если взять всю глобальную экономику, даст такой качественный скачок энергоэффективности, который нельзя было даже представить во времена первой и второй промышленных революций.

Если учесть, что энергия, используемая в процессе, будет поступать из возобновляемых источников и генерироваться на месте потребления, то полный эффект горизонтальной третьей промышленной революции станет совершенно очевидным. Если Интернет радикально сократил входные затраты на генерирование и распространение информации и тем самым открыл дорогу новым компаниям вроде Google и Facebook, то аддитивное производство с его огромным потенциалом снижения себестоимости товаров длительного пользования способно сделать входные затраты достаточно низкими для привлечения сотен тысяч мини-производителей, малых и средних компаний, которые бросят вызов и, возможно, переиграют гигантские производственные компании, составлявшие костяк первой и второй промышленных революций.

Рынок 3D-печати уже испытывает наплыв новых стартапов с такими названиями, как Within Technologies, Digital Forming, Shape Ways, Rapid Quality Manufacturing и Stratasys, которые твердо вознамерились изменить саму идею производства в эру третьей промышленной революции. Производство становится горизонтальным, и это будет иметь колоссальные последствия для общества.

Чтобы почувствовать, насколько распределенная, построенная на сотрудничестве модель бизнеса отличается от традиционной модели XIX и XX веков, посмотрим на Etsy, стремительно растущую, молодую интернет-компанию, которая встала на ноги менее чем за четыре года. Etsy была основана молодым выпускником Нью-Йоркского университета Робом Калином,

занимавшимся изготовлением мебели у себя дома. Столкнувшись с отсутствием выходов на потенциальных покупателей мебели ручной работы, он объединился со своими друзьями и создал веб-сайт, на котором индивидуальные мастера всех специальностей и потенциальные покупатели могли находить друг друга. Сайт стал глобальным виртуальным демонстрационным залом, местом встреч для миллионов покупателей и продавцов более чем из полусотни стран, вдохнувшим новую жизнь в ремесленное производство — искусство, которое почти исчезло под натиском современного промышленного капитализма.

Ткачество и многие другие ремесла стали жертвой промышленного производства еще на заре первой промышленной революции. Местный кустарный промысел не мог соперничать с централизованным фабричным производством и экономией на масштабе, которые стали возможными в результате крупных вложений финансового капитала. Товары фабричного производства просто были более дешевыми, и это поставило ручное производство на грань практически полного исчезновения.

Интернет изменил характер игры, уравнив условия. Контакты между миллионами продавцов и покупателей в виртуальном пространстве устанавливаются почти бесплатно. Заменяя всех посредников — от оптовиков до розничных продавцов — на распределенную сеть, состоящую из миллионов людей, и устранив операционные издержки, которые повышают цену на каждой ступени цепочки поставок, Etsy создала новую глобальную ярмарку товаров ремесленного производства, растущую вширь, а не ввысь, и опирающуюся на горизонтальные, а не вертикальные взаимодействия.

Etsy добавила рынку еще одно измерение — персонализацию взаимоотношений между продавцом и покупателем. Веб-сайт поддерживает тематические чаты, координирует онлайн-выставки изделий и проводит семинары, позволяющие продавцам и покупателям взаимодействовать, обмениваться идеями и устанавливать социальные связи, которые могут сохраняться всю жизнь. Гигантским глобальным компаниям, массово производящим стандартную продукцию на сборочных линиях с безликой рабочей силой, нечего противопоставить тесной связи между ремесленником и покупателем. По словам Калина, «именно живая взаимосвязь человека, который делает вещь, с человеком, который покупает ее, лежит в основе Etsy».

Горизонтальное наращивание масштаба, обусловленное прямым взаимодействием, и ничтожные операционные издержки (если не считать доставки) позволяют кустарному производству конкурировать по цене с

массовым производством. Хотя компания еще очень молода, она демонстрирует быстрый рост. В первой половине 2009 г., когда после краха глобальной экономики рост продаж товаров длительного пользования во всем мире прекратился, у ярмарки Etsy объем продаж достиг \$70 млн и появился миллион новых продавцов и покупателей. В 2010 г. объем продаж вырос до \$350 млн.

В нашем недавнем разговоре Калинин заметил, что его миссия — помогать формированию «эмпатического сознания» на глобальной экономической арене и созданию фундамента более инклюзивного общества, то есть общества, в большей мере обеспечивающего равенство возможностей. Его видение перспективы, в которой будут существовать «миллионы локальных живых экономических ячеек, воссоздающих чувство принадлежности к экономике», — это и есть сущность модели третьей промышленной революции.

Если сайты, подобные Etsy, открывают небольшим ремесленным производителям доступ на глобальный рынок почти с нулевыми входными затратами, то местное генерирование зеленой энергии помогает им снижать себестоимость производства. По мере превращения ремесленниками, а также малыми и средними компаниями своих мастерских в мини-электростанции производственные затраты будут снижаться, обеспечивая им конкурентное преимущество в новой, сетевой экономике.

Как уже говорилось, во времена первой и второй промышленных революций стоимость добычи, переработки и распределения ископаемых энергоносителей была настолько высокой, что только горстка крупных, централизованных игроков могла привлечь достаточный капитал для управления энергетическим потоком. Крупным нефтяным компаниям требовались крупные банки.

Сегодня такие организации микрофинансирования, как Grameen Bank, ASA и EKI, выдали в совокупности более \$65 млрд в виде кредитов более чем сотне миллионов заемщиков в беднейших регионах мира. Микрокредитование все шире используется для финансирования местного генерирования зеленой энергии в таких местах, где раньше никогда не было электричества. Grameen Shakti (GS), дочерняя компания Grameen Bank, предоставляет микрокредиты на установку домашних солнечных и других систем получения возобновляемой энергии в тысячах деревень. К концу 2010 г. GS профинансировала установку полумиллиона домашних систем энергоснабжения на солнечных батареях, при этом за каждые 30 дней монтировалось примерно 17 000 установок. Компания обучила тысячи женщин — они приобрели профессию техников по обслуживанию солнечных установок и получили работу.

Предоставляя микрокредиты беднейшим предпринимателям на земле, Grameen успешно сочетает в своей модели деятельности традиционную коммерческую банковскую практику с нетрадиционной миссией — ликвидацией заколдованного круга бедности. Kiva, некоммерческий участник рынка микрокредитования, продвинула процесс финансирования еще дальше, создав чисто распределенную банковскую модель на основе сотрудничества. Вряд ли можно отыскать что-либо более непохожее на философию коммерческого банкинга, чем философская предпосылка Kiva, созданной в 2005 г. Ее учредители считают, что «люди щедры от природы и помогают другим, если это можно сделать прозрачным, понятным образом». Для реализации своей миссии Kiva «вступает в партнерские отношения вместо осуществления благотворительной деятельности». Для каждого потенциального предпринимателя создается страница с его резюме, фотографией и описанием цели, на которую предполагается использовать кредит. Кредиторы выбирают кредитную заявку, которую они хотели бы профинансировать, и устанавливают размер своего вноса (он может не превышать \$25), а затем объединяются с другими кредиторами, чтобы собрать полную сумму кредита. Все кредиторы ежемесячно получают отчет о погашении кредита.

Организационный процесс кредитования является распределенным в полном смысле. Более сотни полевых партнеров в различных регионах мира предоставляют кредиты еще до того, как заявки на кредит появляются на сайте Kiva. Затем полевые партнеры получают средства от Kiva для восполнения выданных кредитов. Полевые партнеры устанавливают процентные ставки. Kiva не взимает никаких процентов с полевых партнеров и не выплачивает проценты своим кредиторам. После погашения кредита кредитор Kiva может предоставить средства другому предпринимателю, передать их в дар Kiva или забрать.

Используя этот инновационный подход к микрофинансированию, Kiva свела более полумиллиона кредиторов из 209 стран с 469 076 мелкими предпринимателями в 57 странах. Сумма предоставленных кредитов составила \$178 338 325, причем 81% средств получили женщины. Средний размер кредита Kiva равен \$380, а коэффициент погашения — 98,9%. Эти кредиты получают только мелкие предприниматели, чей бизнес имеет незначительные экологические последствия.

Новая деловая практика сотрудничества проникает во все уголки экономической жизни. Сельскохозяйственное производство, поддерживаемое

местным сообществом, — хороший пример влияния бизнес-модели третьей промышленной революции на процесс выращивания и распределения сельскохозяйственной продукции. После столетнего господства нефтехимического сельскохозяйственного производства, которое привело практически к полному уничтожению семейных ферм и к появлению гигантских агрофирм вроде Cargill и ADM, пришло новое поколение фермеров с прямыми связями с домохозяйствами, покупающими их продукцию. Сельскохозяйственное производство, поддерживаемое местным сообществом, зародилось в Европе и Японии в 1960-х гг. и распространилось в Америке в середине 1980-х гг.

Дольщики, обычно городские домохозяйства, вносят фиксированную сумму перед посевным сезоном для покрытия годовых фермерских издержек. В обмен они получают долю урожая фермы в период плодоношения. На практике, как правило, это ящик с созревшими фруктами и овощами, доставляемый прямо к дому или к определенному месту. Таким образом, на протяжении всего сезона плодоношения потребители получают постоянный поток свежей местной продукции.

Фермы по большей части придерживаются экологической сельскохозяйственной практики и используют природные и органические методы выращивания продукции. Поскольку сельскохозяйственное производство, поддерживаемое местным сообществом, является совместным предприятием, основанным на разделении рисков между фермерами и потребителями, последние выигрывают в урожайные годы и проигрывают в неурожайные. В случае неблагоприятных погодных условий или других неудач дольщики мирятся с сокращением еженедельных поставок тех или иных продуктов. Такое равное деление рисков и вознаграждений связывает всех дольщиков, делает их участниками общего предприятия.

Интернет позволил объединить фермеров и потребителей в распределенную структуру на основе сотрудничества с целью организации цепочки поставок пищевых продуктов. Буквально за несколько лет в сфере сельскохозяйственного производства, поддерживаемого местным сообществом, из горстки пилотных проектов возникли почти 3000 предприятий, обслуживающих десятки тысяч семей.

Бизнес-модель сельскохозяйственного производства, поддерживаемого местным сообществом, особенно привлекает молодое поколение, которому привычна идея сотрудничества в цифровом социальном пространстве. Рост ее

популярности также связан с повышением осведомленности потребителей и их стремлением к сокращению неблагоприятного экологического воздействия. С устранением нефтехимических удобрений и пестицидов, выбросов углекислого газа в результате дальних перевозок продуктов питания через океаны и континенты, а также рекламных, маркетинговых и прочих издержек, связанных с традиционным производством и распределительными цепочками второй промышленной революции, каждый дольщик начинает вести более экологически устойчивый образ жизни. Все больше фермеров, занятых в сельскохозяйственном производстве, поддерживаемом местным сообществом, начинают превращать свою недвижимость в мини-электростанции, использующие энергию ветра и солнца, геотермальную энергию и энергию биомассы, и таким образом кардинально сокращают затраты на электроэнергию. От этой экономии выигрывают и дольщики, для которых снижаются годовой членский взнос, и стоимость подписки.

И вновь, как и в случае множества других выстроенных на сотрудничестве новых предприятий, захватывающих один коммерческий сектор за другим, горизонтальный рост здесь может взять и нередко берет верх над традиционным централизованным подходом, предполагающим создание гигантских организаций, которые растут вертикально и организуют экономическую деятельность иерархически.

Некоторые из компаний, в наибольшей мере связанных с традиционным централизованным рыночным капитализмом, уже испытывают конкуренцию со стороны новой распределенной модели бизнеса, основанной на сотрудничестве. Возьмем, например, автомобиль — ключевой элемент второй промышленной революции. Переход к экономике третьей промышленной революции с ее акцентом на повышении энергоэффективности и сокращении выбросов углекислого газа привел к появлению некоммерческих сетей коллективного использования автомобилей по всему миру.

В Америке сети коллективного использования автомобилей охватывают всю страну. City Wheels в Кливленде, HourCar в Миннеаполисе / Сент-Поле, Car Share в Филадельфии, I-Go в Чикаго и City Car Share в Сан-Франциско — вот лишь некоторые представители новой разновидности некоммерческих сетевых организаций, предоставляющих средство передвижения сотням тысяч пользователей. Уплатив символический членский взнос, люди присоединяются к сети коллективного использования автомобилей и получают смарт-карту, дающую право доступа к стоянкам и автомобилям. Пользователи платят за

пробег в милях, однако, поскольку большинство этих организаций являются некоммерческими, стоимость у них ниже, чем у крупных компаний по прокату машин. У многих таких организаций парк состоит из самых энергоэффективных автомобилей, доступных на рынке.

I-Go в Чикаго даже предоставляет инновационную интернет-услугу, позволяющую членам сети организовать поездку из пункта А в пункт Б с использованием разных видов транспорта на маршруте. Пользователь может, например, отправиться в поездку по железной дороге или на автобусе, а потом взять автомобиль и проехать оставшуюся часть пути на нем. Цель — минимизация пробега автомобиля и, таким образом, значительное сокращение выбросов углекислого газа.

По оценкам, каждый коллективно используемый автомобиль выводит с дорог до 20 машин. Пользователи сетей говорят о том, что их автомобильные поездки в милях сокращаются примерно на 44%. Снижение выбросов углекислого газа может быть очень значительным. Communauto, канадская служба коллективного использования автомобилей в Квебеке, сообщила, что ее 11 000 членов сократили выбросы углекислого газа на 13 000 т. Проведенное в Европе исследование показало, что в результате коллективного использования автомобилей выбросы углекислого газа снижаются ни много ни мало на 50%.

Zipcar, крупнейшая в мире компания по коллективному использованию автомобилей, является коммерческой организацией, созданной в 2000 г. Всего за десятилетие число ее членов выросло до сотен тысяч человек. В мире насчитывается несколько тысяч представительств Zipcar, в распоряжении которых находятся более 8000 автомобилей. Компания, выручка которой достигла \$130 млн в 2009 г., растет феноменальными темпами — на 30% в год. В 2010 г. Zipcar запустила пилотный проект с гибридными автомобилями в Сан-Франциско. Компания пользуется популярностью у осознающих проблемы окружающей среды представителей поколения миллениума, которые даже называют себя зипстерами.

С распространением возобновляемой энергии и развитием инфраструктуры третьей промышленной революции автостоянки сетей коллективного использования автомобилей вроде Zipcar будут производить зеленую электроэнергию на месте для зарядки подключаемых к розетке электромобилей. Сообщества коллективного использования автомобилей станут, скорее всего, существенной альтернативой традиционной модели приобретения машин на рынках, особенно в плотно населенных городских районах, где стоимость

содержания автомобиля для нечастых поездок слишком высока.

На Международном транспортном форуме ОЭСР в Лейпциге, Германия, в 2011 г. мне довелось познакомиться с Робин Чейс, основателем и бывшим генеральным директором Zipcar. Я приехал туда, чтобы выступить на открытии с призывом создать интегрированную постуглеродную транспортную и логистическую сеть — пятый столп — на всех континентах к 2050 г. и таким образом сделать шаг в направлении органично связанных континентальных рынков. Робин выступала на транспортной секции сразу после моей презентации. Она подчеркнула, что новая бизнес-модель коллективного использования автомобилей представляет собой прорывное изменение в характере обеспечения мобильности — превращение индивидуального средства передвижения в коллективный транспорт, а индивидуальной практики — в предприятие, основанное на сотрудничестве.

После заседания секции мы с Робин более детально обсудили, как зарождающийся распределенный капитализм раскачивает фундамент традиционной рыночной экономики. Робин в то время занималась созданием новой компании коллективного использования автомобилей, Buzzcar, которая должна была вывести идею распределенной мобильности на основе сотрудничества на следующий уровень — полностью горизонтальную модель бизнеса. Она обратила внимание на то, что миллионы владельцев автомобилей эксплуатируют своих железных коней всего один-два часа в день, а все остальное время они стоят без дела. Робин рассчитывала задействовать миллионы таких автомобилей, сделать их частью огромного парка коллективно используемых транспортных средств, открыть к ним свободный доступ, дать собственникам возможность получать доход от своих автомобилей, а другим людям обеспечить мобильность в прилегающих районах в любой точке мира. Единственное, что мешало сделать это, — отсутствие согласия страховых компаний на страхование физических лиц, а не автомобилей, так чтобы страховка покрывала ответственность и собственника, и пользователя. Робин сказала, что она ведет переговоры с целым рядом страховых компаний и надеется договориться с ними в ближайшее время.

Молодежь начинает совместно пользоваться не только автомобилями. Couch Surfing — международная некоммерческая ассоциация, которая занимается преобразованием сектора путешествий и туризма и попутно сокращает выбросы углекислого газа, связанные с переездами и проживанием сотен тысяч туристов. Глобальная сеть соединяет туристов с местными жителями, которые

открыли свои дома и предлагают бесплатное размещение и гостеприимство. Уже более миллиона участников сети побывали друг у друга в 69 000 городах по всему свету.

Члены ассоциации получают доступ к информации об интересах друг друга и возможностях, а также делятся впечатлениями о том, как их принимали в том или ином месте. Участникам предлагают устанавливать контакты друг с другом перед поездкой и поддерживать связи впоследствии. Эта распределенная и основанная на сотрудничестве социальная сеть создана для объединения людей, представляющих различные культуры, и их знакомства с жизнью друг друга. Цель — помочь «объединению людей путем открытого и дружелюбного общения». Миссия Couch Surfing заключается в продвижении идеи о том, что все мы являемся членами одной глобальной семьи.

Созданная в 2003 г. сеть оказалась удивительно успешной. Ее члены оставили 4,7 млн положительных отзывов, то есть положительно оценили 99,7% всех своих поездок. Еще более впечатляет то, что, по утверждению членов, в результате поездок у них завязались дружеские отношения в 2,9 млн случаев, причем в 120 000 случаях такие отношения характеризуются как близкие.

Частью ответственного отношения, которое неотделимо от принадлежности к глобальному сообществу, является сохранение нашей общей биосферы путем более экологически устойчивого образа жизни. Бесплатное размещение более чем миллиона туристов в местных домах с помощью Couch Surfing значительно сокращает выбросы углекислого газа по сравнению с выбросами при размещении в более энергоемких гостиницах.

Зарождающаяся экономика третьей промышленной революции приводит к появлению таких компаний, основанных на сотрудничестве, о которых не слыхали еще несколько лет назад. Даже крупные глобальные компании втягиваются в игру. Некоторые новые бизнес-модели настолько необычны и нетрадиционны, что требуют полного переосмысления характера коммерческих сделок. Показательным примером являются так называемые контракты на повышение энергоэффективности.

Компании вроде Philips Lighting заключают с городом контракт на установку высокоэффективных светодиодных ламп нового поколения в осветительных приборах во всех общественных местах и на улицах. Банк Philips финансирует проект, а город делает выплаты Philips в течение нескольких лет из средств, полученных в результате экономии электроэнергии. Если Philips не удастся добиться запланированной экономии, то компания получает убыток. Такой вид

сотрудничества со временем будет становиться нормой в экономике третьей промышленной революции.

«Соглашение о разделении доходов от экономии энергии» — еще одна характерная для третьей промышленной революции модель бизнеса, которая переключается с контрактами на повышение энергоэффективности, однако нацелена на другие результаты. Эта новая бизнес-практика с определенным успехом распространяется на рынке жилой недвижимости в ряде стран. Если в Америке у 68% семей дома находятся в собственности, то в других странах подавляющее большинство семей арендуют жилье. Например, в Испании и Германии более половины семей живут в арендуемых квартирах. В местах, где аренда преобладает, собственники недвижимости слабо заинтересованы в модернизации своих зданий и превращении их в мини-электростанции, поскольку счета за коммунальные услуги оплачиваются арендаторами. В Швейцарии, где только 30% семей имеют собственные дома, а большинство являются арендаторами, некоторые домовладельцы заключают с арендаторами соглашения о разделении доходов от экономии энергии. По условиям такого соглашения домовладелец берет на себя обязательство превратить дом в зеленую мини-электростанцию, а арендаторы отдают часть дохода от снижения платы за электроэнергию домовладельцу в течение срока окупаемости его инвестиций. В итоге домовладелец получает здание с более высокой стоимостью, поскольку оно теперь генерирует собственное зеленое электричество. Добавленная стоимость может служить основанием для повышения арендной платы для новых жильцов, однако это повышение меньше той экономии на счетах за электроэнергию, которые они будут получать. Таким образом, в выигрыше оказываются и домовладелец, и арендаторы.

Если глобальная экономика успешно перейдет на инфраструктуру третьей промышленной революции, то предпринимателям и менеджерам придется научиться использовать все передовые бизнес-модели, включая открытые сети, распределенные и основанные на сотрудничестве стратегии исследований и разработок, экологически устойчивую логистику и методы управления цепочками поставок.

Социальное предпринимательство

Характер новой экономики, построенной на сотрудничестве, фундаментально не соответствует классической экономической теории, которая превозносит стремление к личной выгоде на рынке и представляет его как единственный эффективный путь обеспечения экономического роста. Модель третьей

промышленной революции также отвергает централизованную командную систему управления и контроля, которая традиционно ассоциируется с социалистической экономикой. Новая модель тяготеет к горизонтальным предприятиям, как в социальных сообществах, так и на рынке, и исходит из того, что наилучший путь к экологически устойчивому экономическому развитию — это общие цели, которых добиваются совместно.

Новая эра является олицетворением демократизации предпринимательства — каждый становится производителем энергии — и требует сотрудничества для совместного использования и распределения энергии между соседями, регионами и целыми континентами.

Экономика третьей промышленной революции пронизана духом социального предпринимательства, охватывающего весь земной шар. Предпринимательство и сотрудничество больше не считаются чем-то противоречащим друг другу, а рассматриваются как установка на преобразование экономической, социальной и политической жизни в XXI веке.

Социальные предприниматели выходят из университетов по всему миру и создают новые объединяющие коммерческий и некоммерческий секторы компании — эдакие гибриды, которые, скорее всего, станут обычным явлением в ближайшие годы. Доводилось ли вам слышать о TOMS? Эта компания, имеющая и коммерческий, и некоммерческий компоненты, производит обувь, причем не какую-нибудь, а обувь из экологичного, органического, вторичного и даже растительного сырья. Однако это лишь начало рассказа об обувном бизнесе, который, возможно, является самым неординарным в мире.

Прообразом ее продукции с парусиновым или хлопчатобумажным верхом стала традиционная обувь, которую испокон веку носили аргентинские фермеры, так называемая *alpargata*. Компанию создал Блейк Майкоски, молодой социальный предприниматель из Арлингтона, штат Техас, в 2006 г. Обувь TOMS продается более чем в 500 магазинах в США и за границей, в том числе в Neiman Marcus, Nordstrom и Whole Foods.

Коммерческая часть бизнеса Майкоски, которая базируется в Санта-Монике, штат Калифорния, уже продала более миллиона пар обуви. Однако интересно не это. В ответ на продажу каждой пары некоммерческое крыло бизнеса, известное как Friends of TOMS, дарит пару обуви нуждающимся детям в разных частях мира. Более миллиона бесплатных пар обуви было распределено в рамках инициативы «один к одному» среди детей из бедных слоев населения в США, Гаити, Гватемале, Аргентине, Эфиопии, Руанде и Южной Африке.

Спросите, зачем нужно отдавать пару обуви бесплатно на каждую проданную пару? Майкоски говорит, что без обуви детей не пускают в школу во многих беднейших регионах мира. Босые дети рискуют заразиться болезнью, которая называется подоконниозис, или хромомикоз, и передается через почву. Вызывает ее грибок, проникающий в поры подошвы и разрушающий лимфатическую систему человека. По некоторым данным, более миллиарда человек рискуют подхватить заболевания, передаваемые через почву. Простейшее средство профилактики — обувь.

А что происходит с этими миллионами пар обуви, когда она изнашивается? Веб-сайт TOMS Community Wall предлагает клиентам присылать идеи по ее утилизации и превращению во что-нибудь полезное — браслеты, футбольные мячи, подвески для растений, подставки и т. д. TOMS наглядно демонстрирует новую модель социального предпринимательства, появившуюся в эру третьей промышленной революции.

Изменение подхода к ведению бизнеса спровоцировало сражение грандиозного масштаба между старой гвардией второй промышленной революции, которая твердо намерена удержать остатки своей власти, и молодыми предпринимателями третьей промышленной революции, которые не менее преданы идее продвижения горизонтальной, экологически устойчивой экономики в мире. На карте стоит принципиальный вопрос, кто будет контролировать энергию в глобальной экономике XXI века. Обе стороны стремятся к получению рыночного преимущества и используют лоббирование для получения льготного статуса, включая правительственные субсидии и налоговые стимулы стоимостью в миллиарды долларов.

На деле вопрос должен ставиться так: «Где промышленность и правительство хотят оказаться через 20 лет: в тупике с уходящими источниками энергии, технологиями и инфраструктурой исчерпавшей себя второй промышленной революции или в процессе перехода к новым источникам энергии, технологиям и инфраструктуре зарождающейся третьей промышленной революции?».

Ответ очевиден, однако переход в новую эру распределенного капитализма вряд ли окажется легкой прогулкой. Проблемой в данный момент является не отсутствие плана перехода — он у нас есть. Третья промышленная революция является разумным путем к постуглеродной эре. Загвоздка в восприятии ее обществом. Именно здесь мы натываемся на ошибочное представление о том, как происходят экономические революции, которое граничит с бредом.

libremoney.org/article/329

