



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Факультет Компьютерных Наук
Департамент Программной Инженерии
Выпускная квалификационная работа

Криптографические алгоритмы и протоколы для
распределенных реестров

Cryptographic Algorithms and Protocols for Distributed Ledgers

Выполнил: студент гр.БПИ151 Куприянов Кирилл
Научный руководитель: Профессор, руководитель ДПИ,
к.т.н. Авдошин Сергей Михайлович

2019

Популярность блокчайна

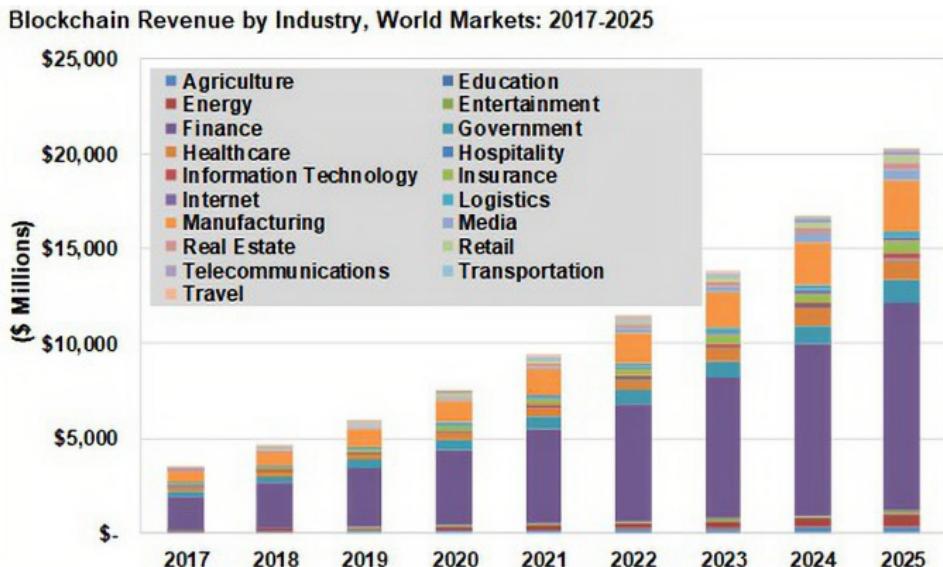


Рис. 1: Рост выручки в индустриях с применением блокчайна [22]

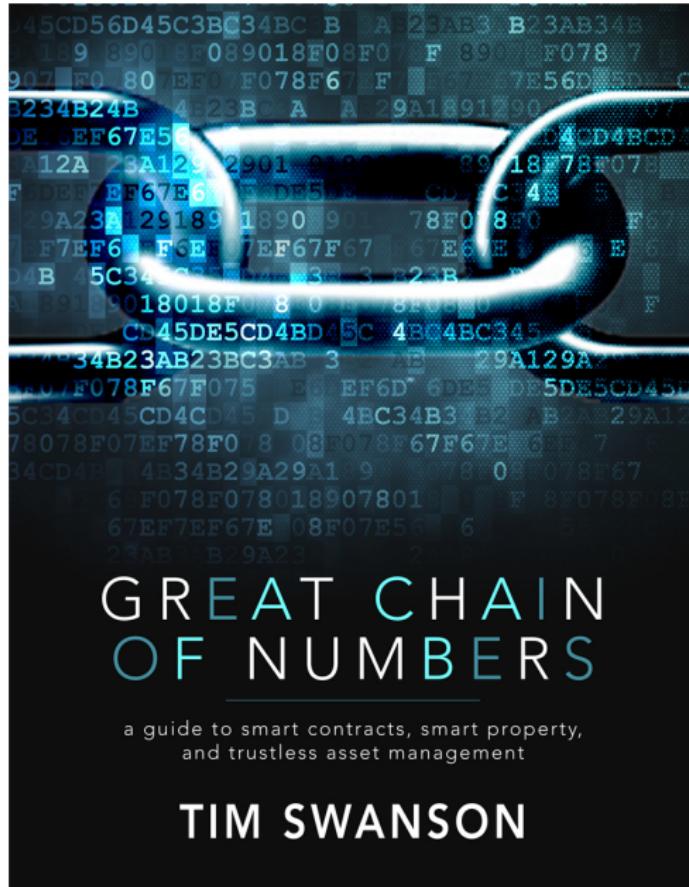


Рис. 2: Swanson, T., Great Chain of Numbers



Определения

- *Распределённый реестр (Distributed Ledger)* — распределённая база данных между сетевыми узлами. Каждый из узлов может получать данные других, при этом храня полную копию реестра. Обновления этих узлов происходят независимо друг от друга
- *Блокчейн* — постоянно растущий список записей, называемых блоками, которые связаны и защищены с помощью криптографии. Он копируется его пользователями и устойчив к модификации
- *Приватный и публичный ключи* — сущности системы асимметричного шифрования для безопасной передачи сообщений между парой субъектов
- *Цифровая подпись* — реквизит электронного документа, полученный в результате криптографического преобразования информации и позволяющий проверить отсутствие искажения информации, принадлежность подписи владельцу сертификата ключа подписи
- *Майнер* — лицо, обеспечивающее достижение консенсуса о том, какие транзакции считать валидными с целью предотвращения траты уже использованной в другой транзакции монеты



Цель и задачи

Расширить существующую классификацию по использованию в реестрах [30] алгоритмов и протоколов, а также создать приложение для автоматизации создания кода распределённого реестра.

Задачи:

- Выявить популярные распределённые реестры; выделить и изучить криптографические алгоритмы и протоколы в них
- Расположить их на диаграмме Эйлера-Венна для создания обновлённой классификации
- Реализовать код блокчейна с интерфейсом встраивания вариаций алгоритмов
- Создать модуль на языке Python3.6.5, позволяющий пользователю генерировать код работающего с использованием выбранных алгоритмов блокчейна
- Автоматизировать работу системы с набором существующих алгоритмов



План

- Вступление
- Источники информации
- Результат обновления классификации
- Описание программного решения
- Схема работы
- Особенности архитектуры
- Технологии реализации
- Заключение





DECENTRALIZE.TODAY

Your Daily Dose of Decentralization

R^G

W



DECENTRALIZE.TODAY

Your Daily Dose of Decentralization



DECENTRALIZE.TODAY

Your Daily Dose of Decentralization



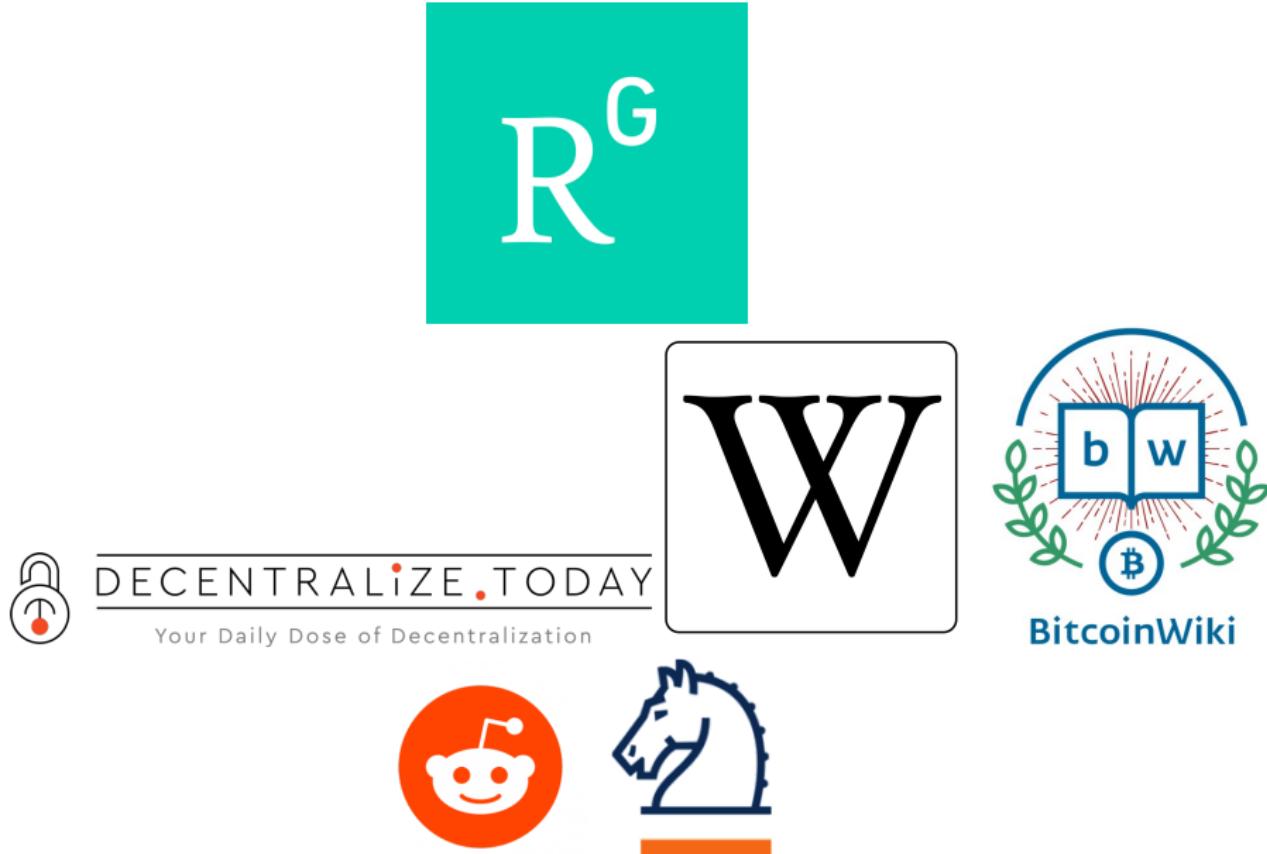


Рис. 3: Источники информации ([1-2], [36], [39], [6], [22])

Классификация от Tim Swanson [30]

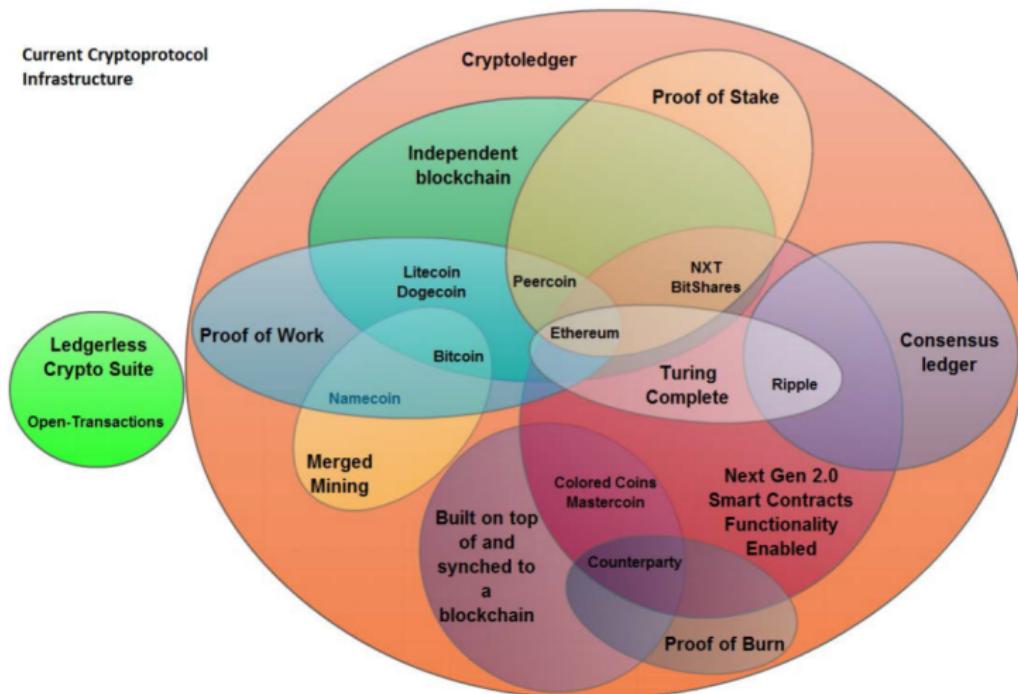


Рис. 4: Криптовидокол по состоянию на 2014 год [30]

Обновлённая классификация

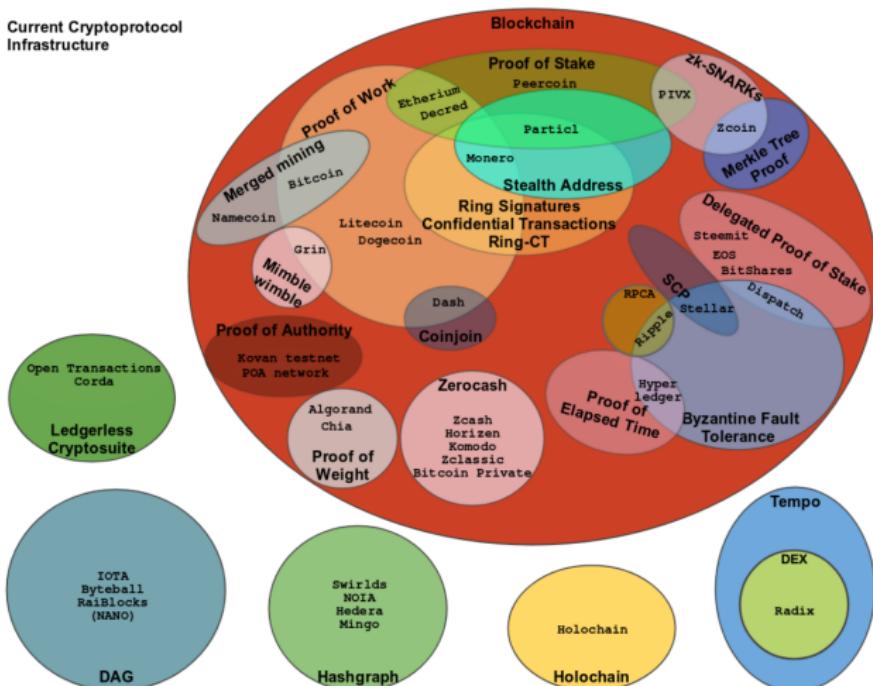


Рис. 5: Криптопротокол по состоянию на 2019 год

Идея программного решения

Аналоги предлагаемого программного решения

Таблица 1. Сравнение с аналогами

Признак	Codecreator	Azure BaaS	MagicCodeGenerator ¹	GSL
Встраивание кода в своё приложение	Частично	Частично	—	Да
Моментальное развертывание	—	—	—	Да
Вариаций алгоритмов использования	< 5	< 10	0	24
Стоимость	≈5600 руб./мес.	≈3300 руб./мес.	Не известно	0 руб. / мес.

¹Релиз MagicCodeGenerator ещё не состоялся.



Рис. 6: Аналоги программного решения. Codecreator [7] и Azure BaaS [20]



1. Компоновщик
2. Реализация блокчейна

Схема работы компоновщика

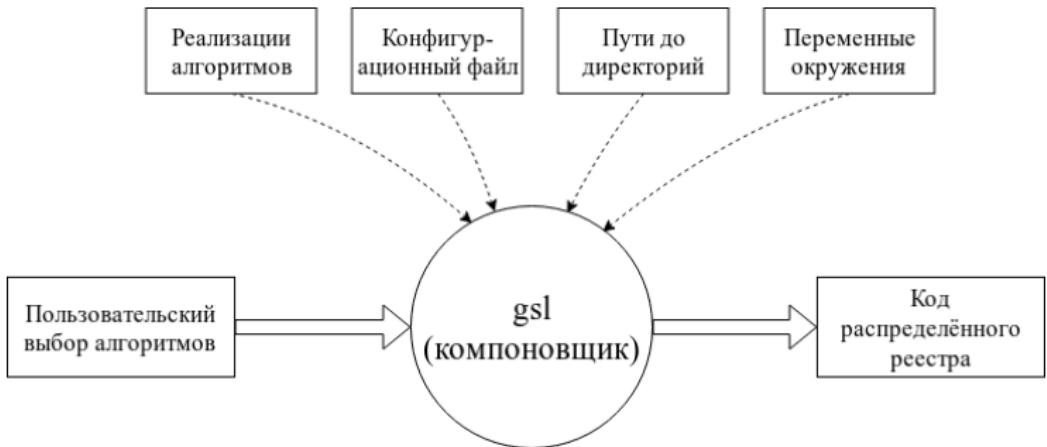


Рис. 7: Схема работы компоновщика

Схема работы компоновщика

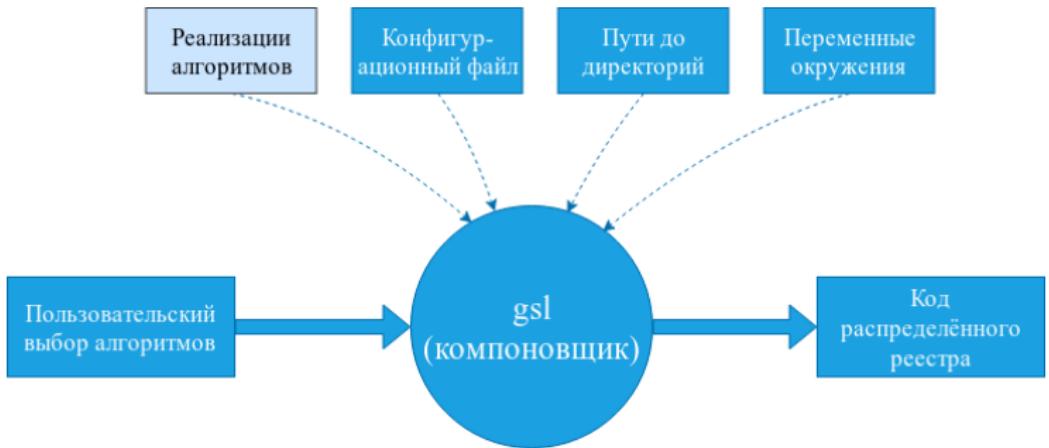


Рис. 8: Схема работы компоновщика (с указанием степени принадлежности реализаций: синий — собственная, голубой — готовая)

Программа

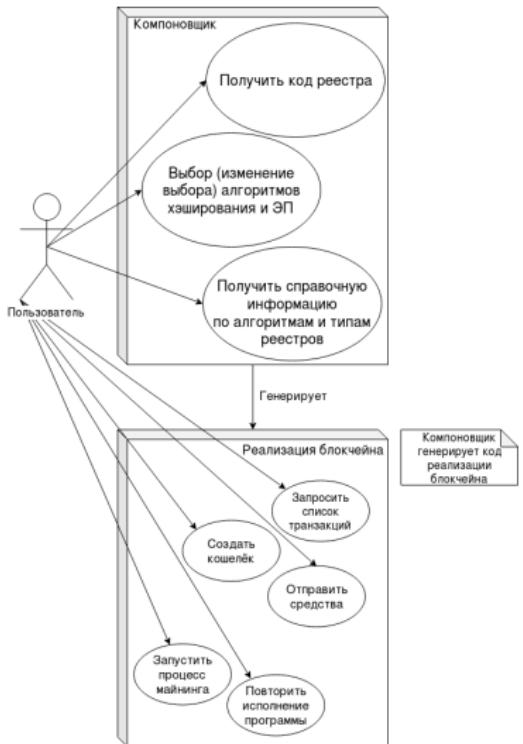


Рис. 9: Диаграмма use case



Особенности архитектуры

- Сервер автообновления поддерживает консистентность реализаций алгоритмов
- **Continuous integration** позволяет отслеживать работоспособность автообновляемой версии реализаций алгоритмов
- Спроектировано для упрощения дальнейшей масштабируемости
- Реализации алгоритмов были адаптированы под **Python3**
- Использование **конфигурационных** файлов вместо разрастания количества аргументов командной строки

Сервер автообновления

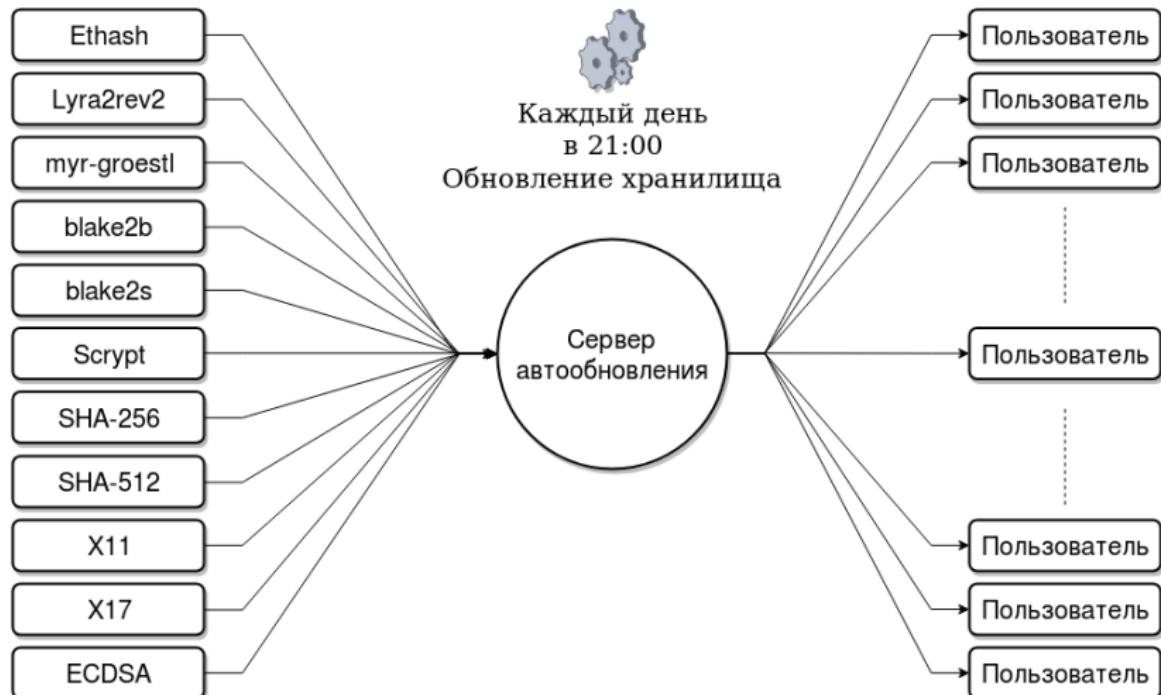


Рис. 10: Процесс работы автообновления



```
1 00 20 * * * echo $(date) >> updater.log && cd src && ./updater.py >> updater.log 2>&1
2 00 21 * * * echo $(date) >> pusher.log && gsl && ./pusher.sh >> pusher.log 2>&1
```

Рис. 11: Регулярное (ежедневное) обновление

Технологии реализации



Рис. 12: Инструменты реализации

Демонстрация



Выводы

- ✓ Устаревшая классификация обновлена
- ✓ В новой отражены не только новые алгоритмы и протоколы, но и современные распределённые реестры
- ✓ Разработано средство автоматизации программирования
- ✓ Создан автоматизированный процесс по работе с кодами реализаций алгоритмов и их обновлению
- ✓ Работа в публичном доступе с инструкциями по установке и использованию

gsl

run shippable License GPLv3

gsl -- Goodsteel ledger. Krinkle Goodsteel will help you to build your own distributed ledger.

Build and use with Python3.6.X+ interpreter.

Installation

Execute following commands linebyline

```
git clone https://github.com/Sinopsys/gsl.git
cd gsl
export PYTHONPATH=$PYTHONPATH":$(pwd)/src"
[[ $PATH != *".local/bin"* ]] && export PATH=$PATH":/home/$USER/.local/bin"
echo "mkdir /tmp/gsl" && mkdir /tmp/gsl
echo "sudo mkdir /etc/gsl" && sudo mkdir /etc/gsl
echo "sudo cp ./example_config.yaml /etc/gsl/config.yaml" && sudo cp ./example_config.yaml /etc/gsl/config.yaml
python3.6 -m pip install . --user

## OPTIONAL, if you do not want to set it manually every time
## Just echoes above exports to your $SHELLrc file
rc_file="/home/$USER/.${echo $SHELL | rev | cut -d / -f 1 | rev}rc"
echo "export PATH=\$PATH":/home/$USER/.local/bin"" >> $rc_file
echo "export PYTHONPATH=\$PYTHONPATH":$(pwd)/src\"" >> $rc_file
```

Then, to successfully launch app, it is needed to have a config with path `/etc/gsl/config.yaml`. During installation, an

Рис. 13: Информационная страница репозитория
[<https://github.com/Sinopsys/gsl>]



Направления дальнейшей работы

- Новые реестры
- Новые алгоритмы
- Интерфейсы для встраивания в различные системы
- Описание взаимодействия с другими приложениями
- Реализации алгоритмов Python → C
- Исследование новых реестров
- Поддержание актуальности классификации



Список используемых источников

1. 279 questions in Blockchain | Science topic. — URL: <https://www.researchgate.net/topic/Blockchain> (дата обр. 20.05.2019).
2. 97 questions in Cryptocurrency | Science topic. — URL: <https://www.researchgate.net/topic/Cryptocurrency> (дата обр. 20.05.2019).
3. API — Flask 0.12.4 documentation. — URL: <http://flask.pocoo.org/docs/0.12/api/#%7Dflask.Flask> (дата обр. 20.05.2019).
4. *Ashish Kotbsbtechdac.* DAG will overcome Blockchain Problems DAG VS. BLOCKCHAIN. — 2018. — URL: <https://medium.com/coinmonks/dag-will-overcome-blockchain-problems-dag-vs-blockchain-9ca302651122> (дата обр. 20.05.2019).
5. Bitcoin Is Unsustainable. — URL: https://motherboard.vice.com/en%7B%5C_%7Dus/article/ae3p7e/bitcoin-is-unsustainable (дата обр. 23.04.2019).
6. Blockchain. — URL: <https://www.reddit.com/r/BlockChain/> (дата обр. 20.05.2019).
7. *Code Creator.* BlockChain. — URL: <https://www.codecreator.com/index.php/blockchain/> (дата обр. 20.05.2019).
8. Continuous Integration and Delivery - CircleCI. — URL: <https://circleci.com/> (дата обр. 20.05.2019).
9. Cryptocurrency news and discussions. — URL: <https://www.reddit.com/r/CryptoCurrency/> (дата обр. 20.05.2019).
10. Distributed Messaging - zeromq. — URL: <http://zeromq.org/> (дата обр. 20.05.2019).
11. *Dwork C., Naor M.* Pricing via Processing or Combating Junk Mail // Advances in Cryptology — CRYPTO' 92. — 2007. — С. 139—147. — DOI: 10.1007/3-540-48071-4_10.
12. *Etherium.* TESTNET Kovan (KETH) Blockchain Explorer. — 2018. — URL: <https://kovan.etherscan.io/> (дата обр. 23.04.2019).
13. *Groth J., Kohlweiss M.* One-out-of-many proofs: Or how to leak a secret and spend a coin // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). — 2015. — Т. 9057. — С. 253—280. — ISSN 16113349. — DOI: 10.1007/978-3-662-46803-6_9.
14. gRPC. — URL: <https://grpc.io/> (дата обр. 20.05.2019).
15. *Jim.* Hash codes are not unique. — 2012. — URL: <http://blog.mischel.com/2012/04/13/hash-codes-are-not-unique/> (дата обр. 20.05.2019).



Список используемых источников

16. Lamport L., Shostak R., Pease M. The Byzantine Generals Problem // ACM Transactions on Programming Languages and Systems. — 2002. — Т. 4, № 3. — С. 382—401. — ISSN 01640925. — DOI: 10.1145/357172.357176.
17. Magic Code Generator – Tools for blockchain-based application development. — URL: <https://magiccodegenerator.com/> (дата обр. 20.05.2019).
18. Maurer F. K., Florian M. Anonymous CoinJoin Transactions with Arbitrary Values. —.
19. Maxwell G. Confidential Transactions. — 2015.
20. Microsoft Azure. Blockchain Technology and Applications. — URL: <https://azure.microsoft.com/en-us/solutions/blockchain/> (дата обр. 20.05.2019).
21. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Consulted, 1–9. // Journal for General Philosophy of Science. — 2008. — № 1. — С. 1–9. — ISSN 09254560. — DOI: 10.1007/s10838-008-9062-0. — arXiv: 43543534v343453.
22. Niranjanamurthy M., Nithya B. N., Jagannatha S. Analysis of Blockchain technology: pros, cons and SWOT // Cluster Computing. — 2018. — Март. — С. 1–15. — ISSN 1386-7857. — DOI: 10.1007/s10586-018-2387-5. — URL: <http://link.springer.com/10.1007/s10586-018-2387-5>.
23. Noether S., Mackenzie A., Research Lab T. M. Ring Confidential Transactions // Ledger. — 2016. — Т. 1. — С. 1–18. — DOI: 10.5195/ledger.2016.34.
24. Poelstra A. Mimblewimble. — 2016.
25. Popov S. IOTA whitepaper v1.4.3. — 2018. — URL: https://assets.ctfassets.net/r1drf6vzxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1%7B%5C_%7D4%7B%5C_%7D3.pdf.
26. Shippable. — URL: <https://app.shippable.com/accounts/5aad7d3e76ee0c1700c1d8d2/dashboard> (дата обр. 20.05.2019).
27. Sir Mark Walport. Distributed Ledger Technology: beyond block chain. — 2018. — DOI: 10.1021/acs.aem.8b00240.
28. SQLite. — URL: <https://sqlite.org/index.html> (дата обр. 20.05.2019).
29. The Official YAML Web Site. — URL: <https://yaml.org/> (дата обр. 20.05.2019).



Список используемых источников

30. *Tim Swanson. Great Chain of Numbers.* — 2014. — URL: <https://www.scribd.com/document/210537698/Great-Chain-of-Numbers-a-Guide-to-Smart-Contracts-Smart-Property-and-Trustless-Asset-Management-Tim-Swanson> (дата обр. 23.04.2019).
31. *Travis CI.* — URL: <https://travis-ci.org/> (дата обр. 20.05.2019).
32. *Using etcd.* — URL: <https://coreos.com/etcd/> (дата обр. 20.05.2019).
33. *Van Saberhagen N. CryptoNote v 1.0.* — 2012.
34. *Van Saberhagen N. Monero: CryptoNote v 2.0 // White Paper.* — 2013. — С. 1–20. — URL: <https://bytecoin.org/old/whitepaper.pdf%7B%5C%7D0Ahttps://cryptonote.org/whitepaper.pdf>.
35. *Vitalik Buterin. On Public and Private Blockchains.* — 2015. — URL: <https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/> (дата обр. 22.04.2019).
36. *Wikipedia.* — URL: <https://www.wikipedia.org/> (дата обр. 20.05.2019).
37. *Zerocoin Electric Coin Company. ZCash.* — 2016.
38. *ZEXE: Enabling decentralized private computation / S. Bowe [и др.].* — 2019. — URL: <https://eprint.iacr.org/2018/962.pdf>.
39. *Направленный ациклический граф — Википедия.* — URL: https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%BD%D1%8B%D0%B9%7B%5C_%7D%D0%B0%D1%86%D0%B8%D0%BA%D0%BB%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9%7B%5C_%7D%D0%B3%D1%80%D0%B0%D1%84 (дата обр. 22.05.2019).
40. *Ольга Скоробогатова. О российском консорциуме, национальной электронной валюте.* — 2016. — URL: <https://bankir.ru/publikacii/20160419/olga-skorobogatova-o-rossiiskom-konsortsiume-natsionalnoi-elektronnoi-valyute-10007442/> (дата обр. 23.04.2019).



Спасибо за внимание!

Факультет Компьютерных Наук
Департамент Программной Инженерии
Выпускная квалификационная работа

Выполнил: студент гр.БПИ151 Куприянов Кирилл
Научный руководитель: Профессор, руководитель ДПИ,
к.т.н. Авдошин Сергей Михайлович

+7-910-008-3926
kikupriyanov@edu.hse.ru
kupriyanovkirill@gmail.com
[@SsinopsysS](https://twitter.com/SsinopsysS)

2019

Процесс работы компоновщика

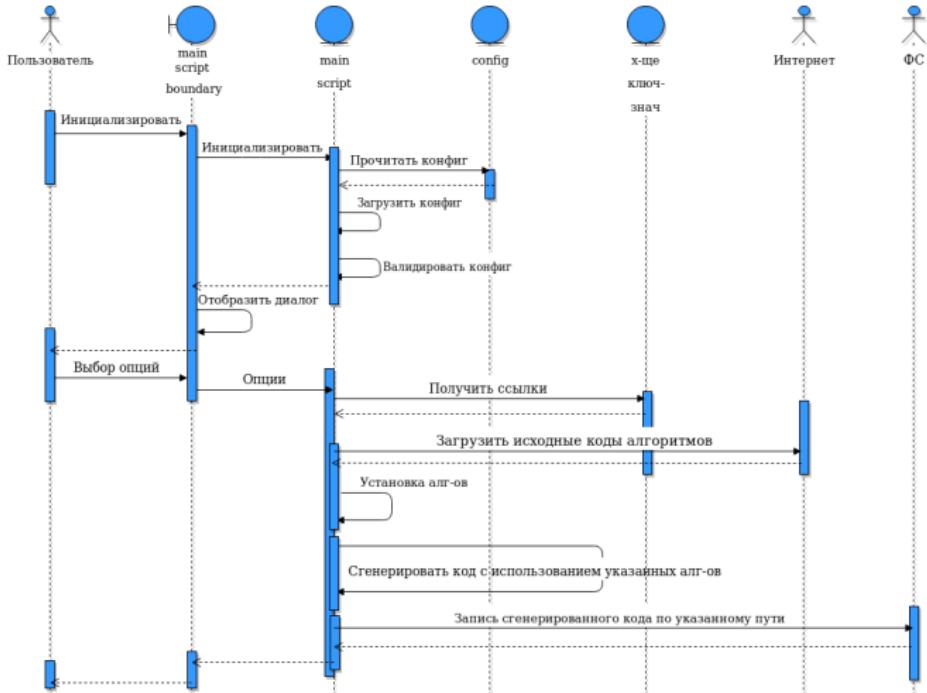


Рис. 14: Диаграмма последовательности работы компоновщика



Репозиторий

The screenshot shows the GitHub README.md page for the 'gsl' repository. The page contains the following sections:

- gsl**: A brief description of the project: "gsl -- Goodsteel ledger. Krinkle Goodsteel will help you to build your own distributed ledger."
- Build and use with Python3.6+ interpreter.**
- Installation**: Instructions to execute the following command line by line:

```
git clone https://github.com/kupriyanov/gsl.git
cd gsl
export PYTHONPATH=$PYTHONPATH:$PWD/src
[[ $PATH != *":$PWD/bin" ]] && export PATH=$PATH:$PWD/bin
echo "$PWD/bin" >/etc/gsl
&& chmod +x /etc/gsl
&& sudo mkdir /etc/gsl
&& sudo cp ./example_config.yaml /etc/gsl/config.yaml
&& sudo cp ./example_config.yml /etc/gsl/config
```
- Then, to successfully launch app, it is needed to have a config with path /etc/gsl/config.yaml**. During installation, an example config has already been put to that path.
- Example config.yaml**:

```
# Key init_dir is path where your blockchain will be stored.
# init_dir: /tmp/gsl
```
- If you encounter error like ModuleNotFoundError: No module named 'goodsteel_ledger' then first, cd into gsl directory and then run**

```
export PYTHONPATH=$PYTHONPATH:$PWD/src
```
- Usage**:

Tip: before usage make a git pull

```
gsl --init --name NAME [-path PATH] # path can also be taken from config.
```
- Questions**:

Write kupriyanov8@gmail.com, mephisto@openmail.cc, <https://t.me/Sinopys5>, or create an issue

Рис. 15: Информационная страница репозитория

Схема хранилища

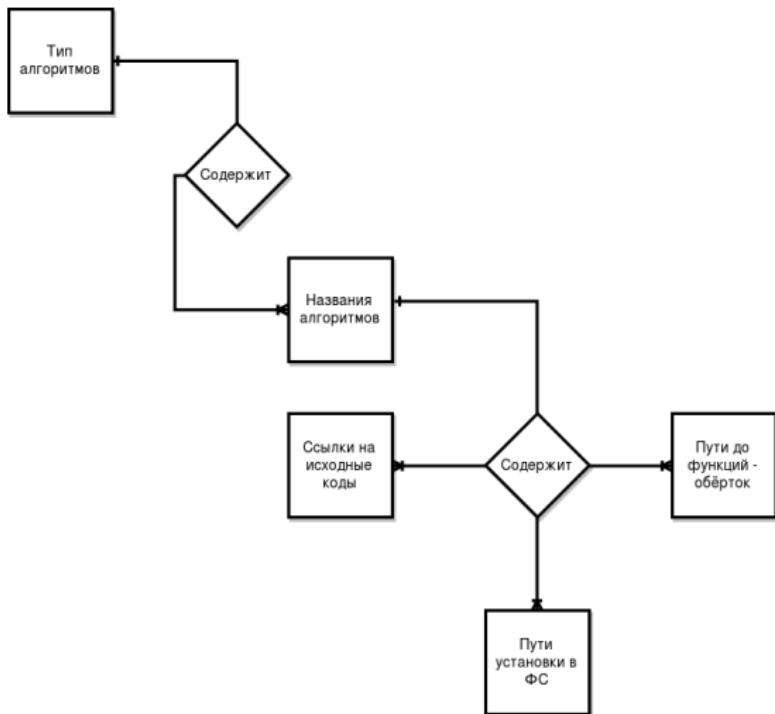


Рис. 16: Схема хранилища