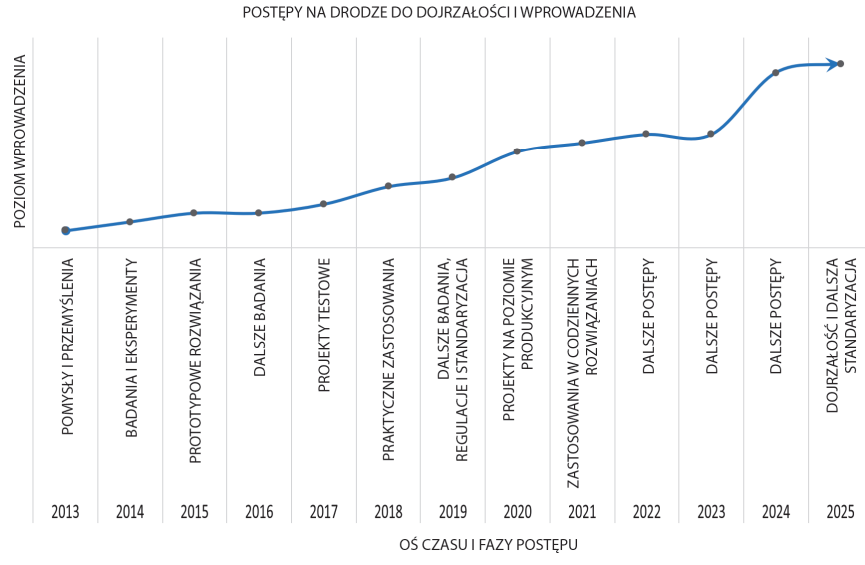
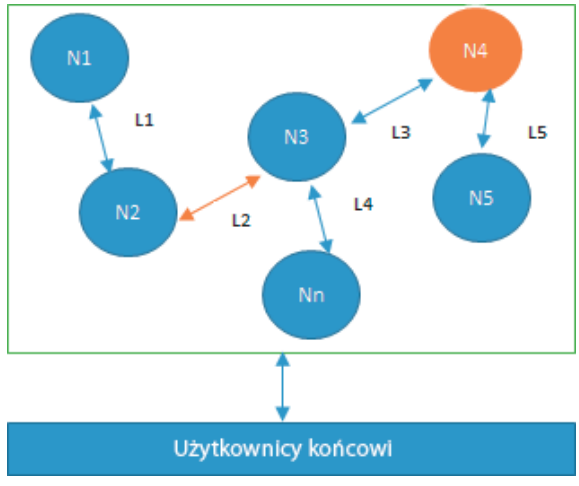
Rozdział 1. ABC łańcucha bloków



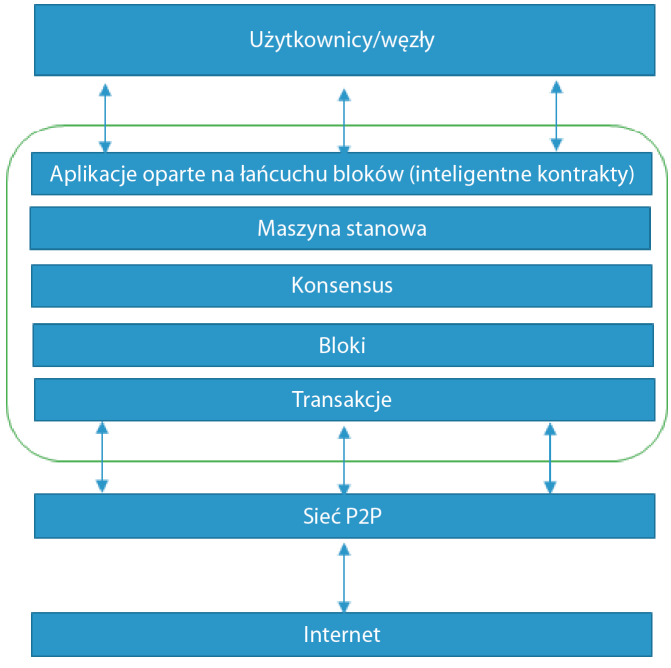
Rysunek 1.1. Wprowadzanie technologii łańcucha bloków i jej dojrzewanie

Rysunek 1.2. Liczba wyszukiwań słowa blockchain (czyli łańcuch bloków) w wyszukiwarce Google

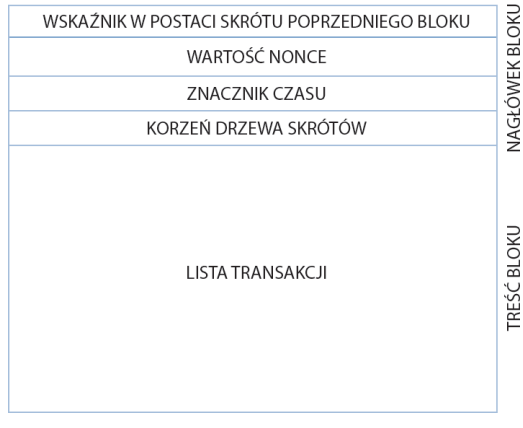


Rysunek 1.3. Projekt systemu rozproszonego: N4 to węzeł bizantyjski, L2 to uszkodzone lub powolne łącze sieciowe

Rysunek 1.4. Różne idee wykorzystane do opracowania Bitcoina i łańcucha bloków



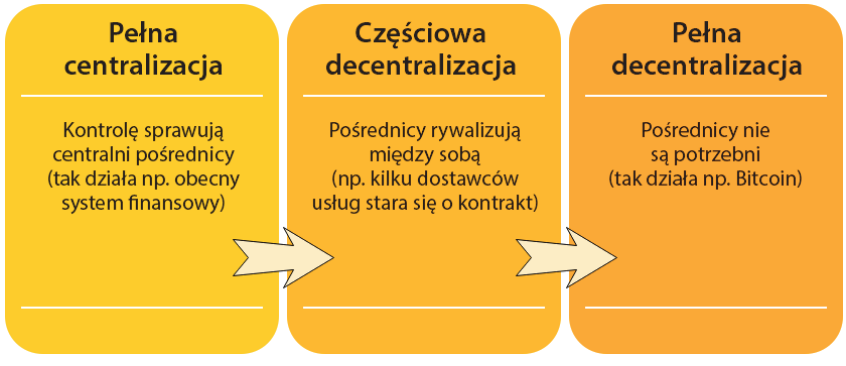
Rysunek 1.5. Obraz łańcucha bloków w kontekście sieci



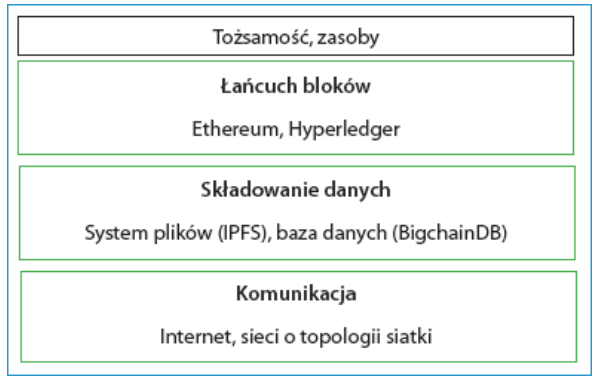
Rysunek 1.6. Ogólna struktura bloku

Rysunek 1.7. Ogólna struktura łańcucha bloków

Rozdział 2. Decentralizacja

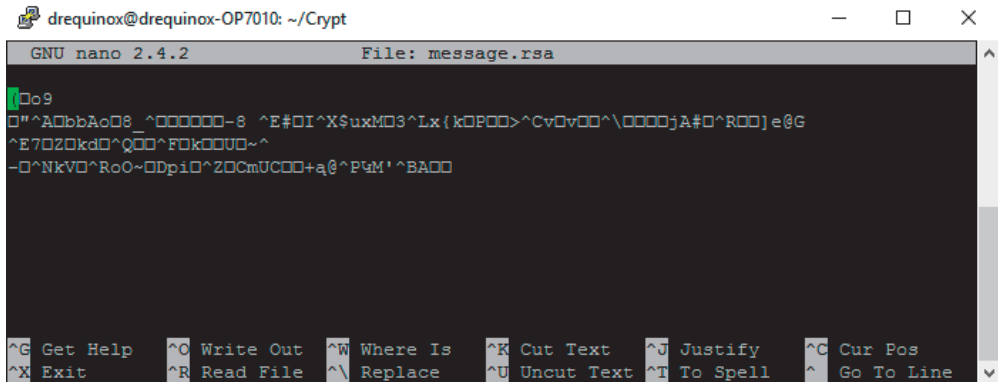


Rysunek 2.2. Poziom decentralizacji

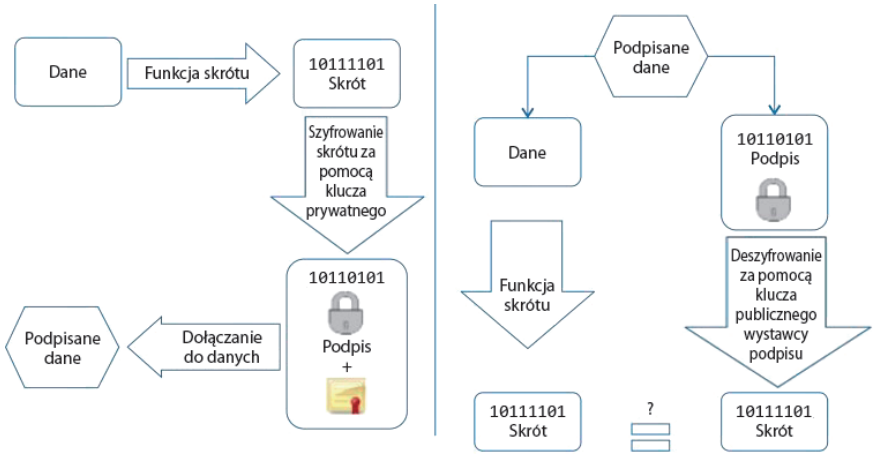


Rysunek 2.3. Zdecentralizowany ekosystem

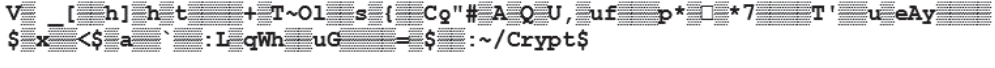
Rozdział 4. Kryptografia klucza publicznego



Rysunek 4.8. Plik message.rsa zawierający niezrozumiałe (zaszyfrowane) dane



Rysunek 4.15. Tworzenie podpisu cyfrowego (po lewej) i proces sprawdzania podpisu (po prawej). Przykład tworzenia podpisów cyfrowych za pomocą RSA

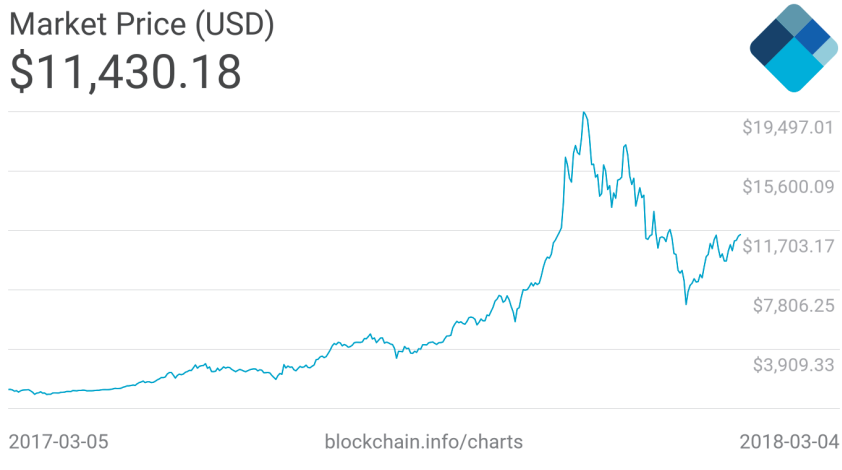


Rysunek 4.16. Zawartość pliku signature.bin

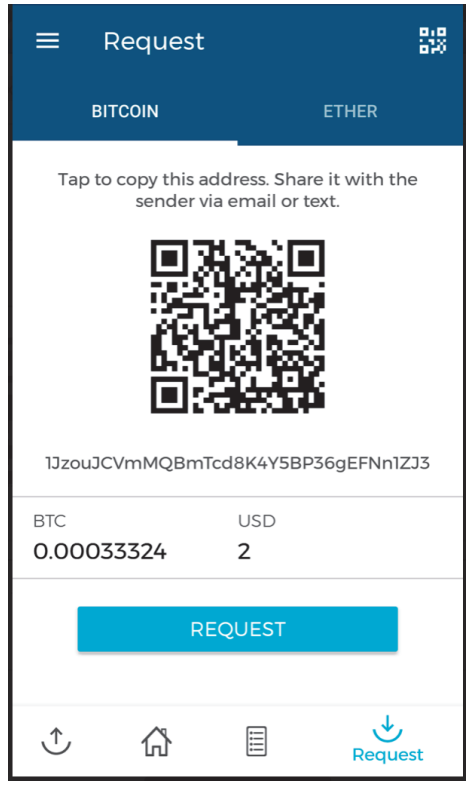
```
drequinox@drequinox-OP7010: ~/Crypt
drequinox@drequinox-OP7010:~/Crypt$ openssl x509 -in ecccertificate.pem -text -noout
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 13205206053355364006 (0xb74250f0fc159ea6)
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=GB, ST=Cambridge, L=Cambridge, O=Dr.Equinox!, OU=NA, CN=drequinox/emailAddress=drequinox@drequinox.com
        Validity
            Not Before: Sep 27 00:09:43 2016 GMT
            Not After : Sep 27 00:09:43 2017 GMT
        Subject: C=GB, ST=Cambridge, L=Cambridge, O=Dr.Equinox!, OU=NA, CN=drequinox/emailAddress=drequinox@drequinox.com
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:10:a2:92:e0:4e:3e:4c:04:c8:78:15:fc:a3:62:
                    7a:3f:12:a4:8d:ca:16:ad:73:f0:35:1a:3f:93:86:
                    3f:09:90:38:a5:7b:e5:c9:38:07:e4:b6:26:41:b5:
                    34:a9:4b:4f:33:b7:40:13:33:ac:6a:85:e6:7a:da:
                    81:fb:a7:0c:f9
                ASN1 OID: secp256k1
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                E1:68:E7:87:EE:E1:44:F0:70:71:76:4D:73:C6:15:14:F1:14:BE:F4
            X509v3 Authority Key Identifier:
                keyid:E1:68:E7:87:EE:E1:44:F0:70:71:76:4D:73:C6:15:14:F1:14:BE:F4

            X509v3 Basic Constraints:
                CA:TRUE
    Signature Algorithm: ecdsa-with-SHA256
         30:44:02:20:5e:ab:c9:85:f1:4f:e5:b1:05:e3:0f:ef:da:84:
         d7:d5:5f:c5:e9:20:be:c3:3c:34:b6:74:f4:a6:5e:11:3c:e0:
         02:20:65:b2:78:78:c7:80:ea:cf:e8:42:c4:ac:de:fb:c8:76:
         a0:15:62:0d:d0:89:f7:41:2a:03:9f:be:92:a7:2d:21
drequinox@drequinox-OP7010:~/Crypt$
```

Rysunek 4.17. Certyfikat X509 wykorzystujący algorytmy ECDSA i SHA-256

Rozdział 5. Wprowadzenie do Bitcoina

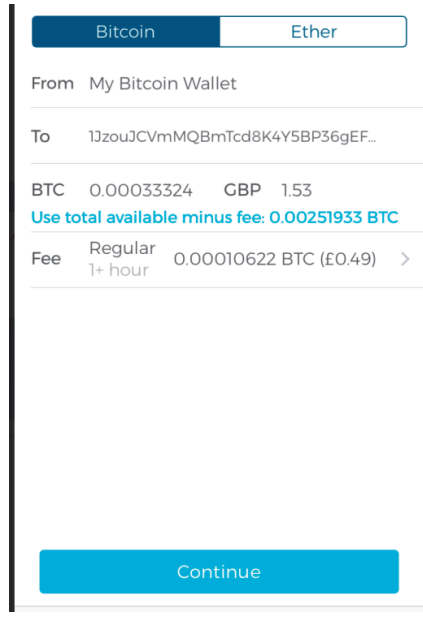


Rysunek 5.1. Ceny bitcoinów od marca 2017 r.

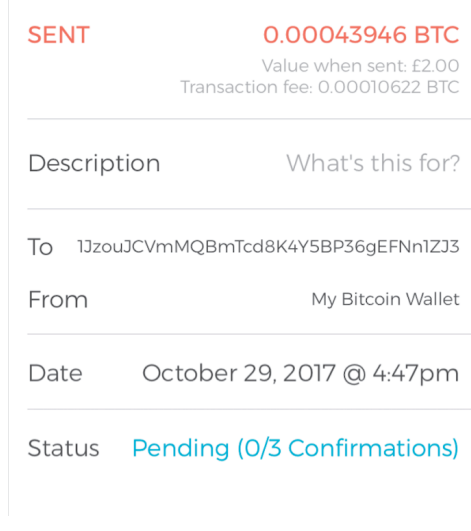


Rysunek 5.2. Żądanie płatności w bitcoinach (w portfelu Blockchain)

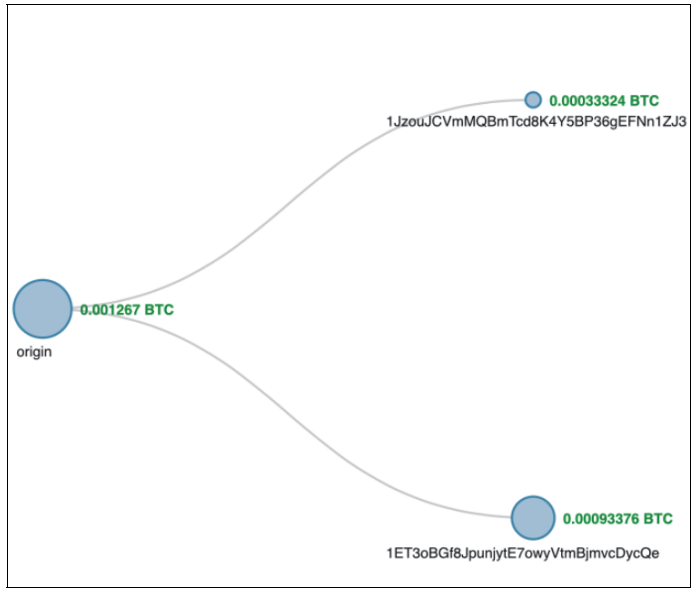
Rysunek 5.3. Kod QR reprezentujący płatność w bitcoinach



Rysunek 5.4. Wysyłanie bitcoinów za pomocą portfela Blockchain [138



Rysunek 5.5. Przesłana transakcja



Rysunek 5.6. Wizualizacja przepływu środków w transakcji (z witryny Blockchain.info)

Transakcja Zobacz informacje o transakcji bitcoin

d28ca5a59b2239864eac1c96d3fd1c23b747f0ded8f5af0161bae8a616b56a1d

1PL6gsm49xCFMvrXqgGcee5cdrG119GoWN	1JzouJCVmMQBmTcd8K4Y5BP36gEFNn1ZJ3 1ET3oBGf8JpunjytE7owyVtmBjmvcDycQe	0.00033324 BTC 0.00093376 BTC

0.001267 BTC

Podsumowanie	
Rozmiar	226 (Bajtów)
Waga	904
Czas otrzymania	2017-10-29 16:51:42
Zawarta w blokach	492229 (2017-10-29 16:51:42 + 0 minut)
Potwierdzeń	47715
Wizualizacja	Zobacz wykres drzewa

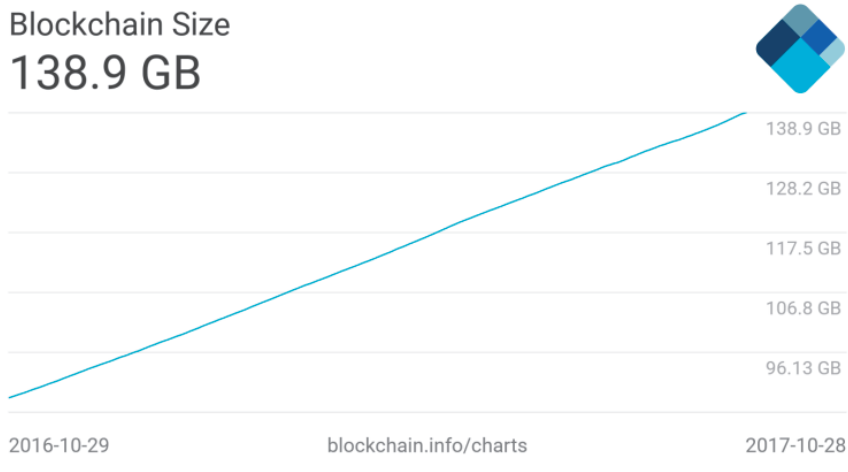
Przychody i wyjścia	
Razem przychodów	0.00137322 BTC
Razem wychodzących	0.001267 BTC
Opłaty	0.00010622 BTC
Opłata za bajt	47 sat/B
Opłata za jednostkę wagi	11.75 sat/WU
Szacunkowa ilość BTC w transakcji	0.00033324 BTC
Skrypty	Pokaż skrypty & coinbase

Rysunek 5.7. Informacje o transakcji pobrane z witryny Blockchain.info

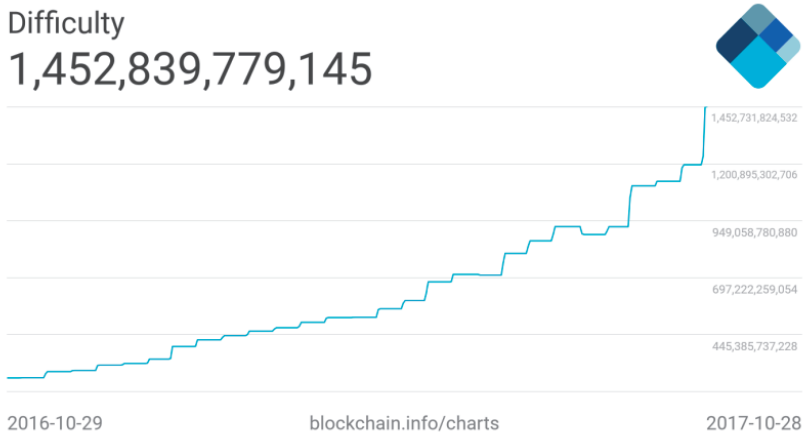
Rysunek 5.8. Hologram zabezpieczający na fizycznym bitcoinie Casascius z widocznym minikluczem i kodem QR

Rysunek 5.10. Klucz prywatny i adres Bitcoin w papierowym portfelu (za stroną: bitaddress.org)

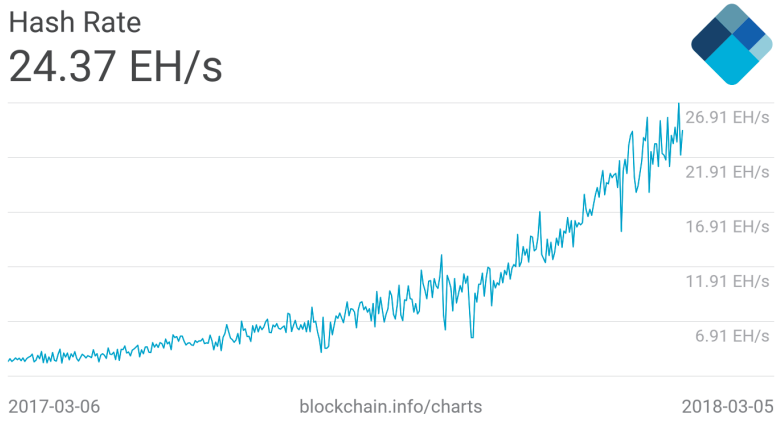
Rysunek 5.12. Adres vanity wygenerowany na stronie https://bitcoinvanitygen.com/



Rysunek 5.15. Aktualna wielkość łańcucha bloków na dzień 29.10.2017

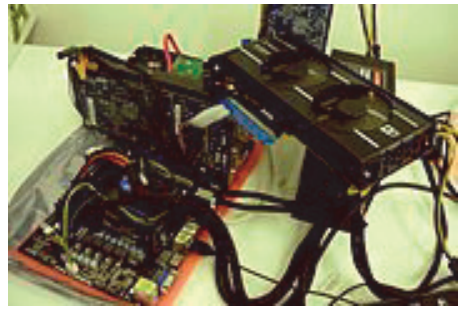


Rysunek 5.17. Trudność wydobywania bloków w ostatnim roku

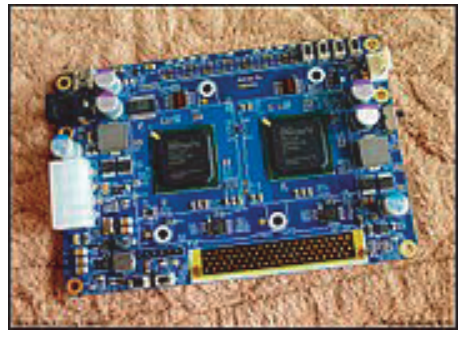


Rysunek 5.18. Szybkość obliczania skrótów (mierzona w eksahashach) w okresie roku do marca 2018 r.

Rysunek 5.19. Zwykły procesor

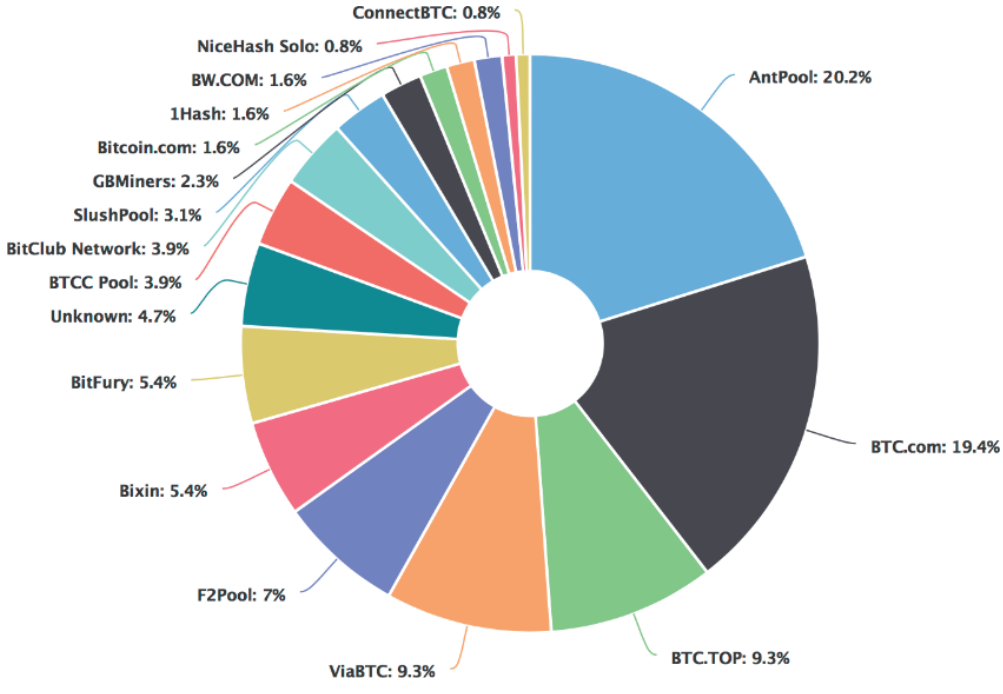


Rysunek 5.20. Procesor graficzny



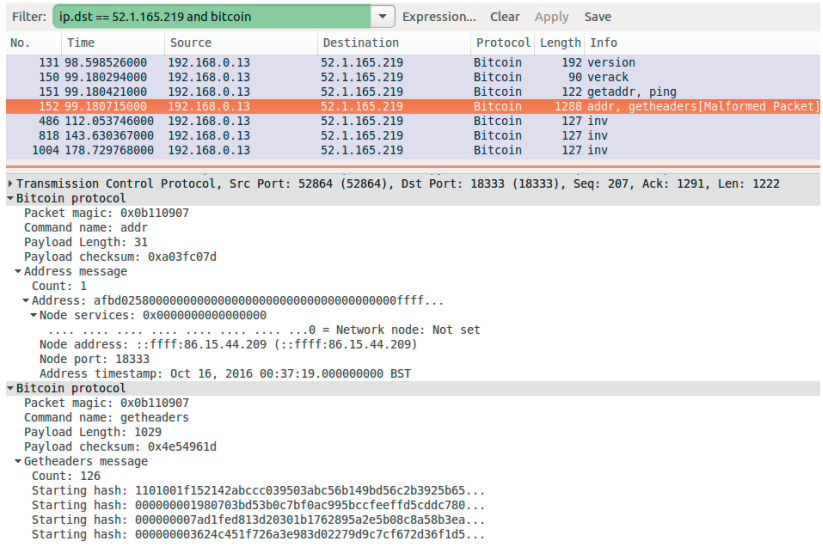
Rysunek 5.21. Układ FPGA

Rysunek 5.22. Układy ASIC

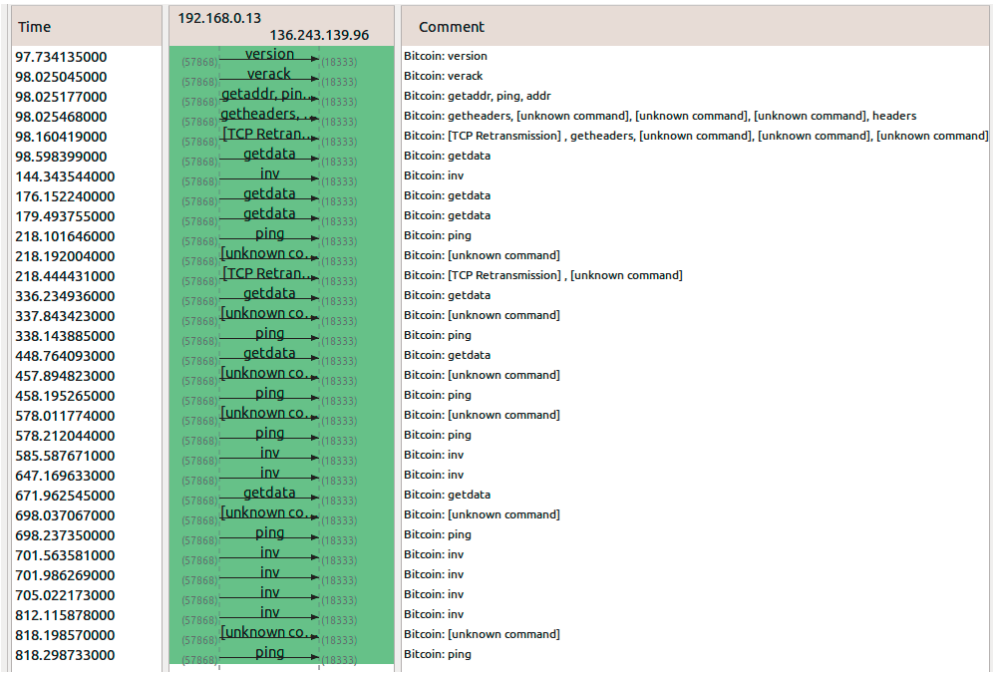


Rysunek 5.23. Kopalnie i ich moc obliczeniowa (szybkość obliczania skrótów) na dzień 28.10.207. Źródło: https://blockchain.info/pools

Rozdział 6. Sieć Bitcoina i płatności

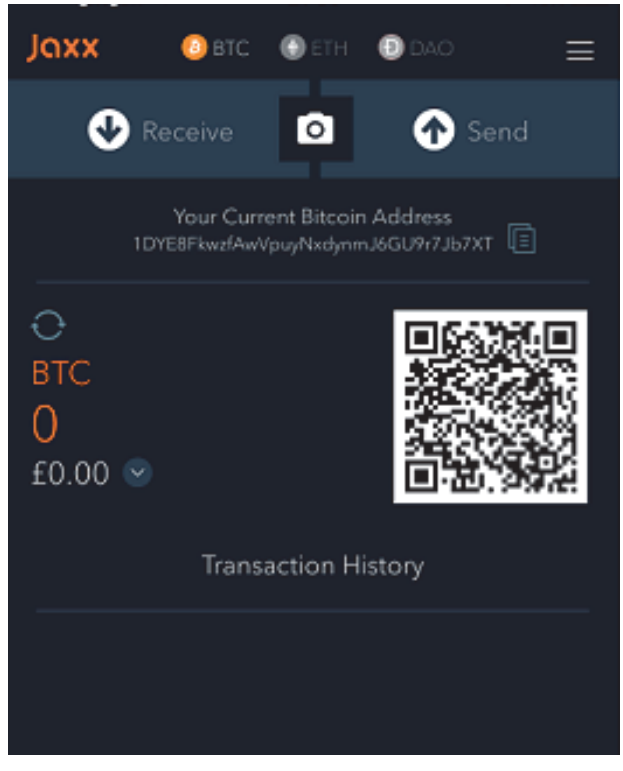


Rysunek 6.3. Przykładowy komunikat w narzędziu Wireshark



Rysunek 6.4. Protokół

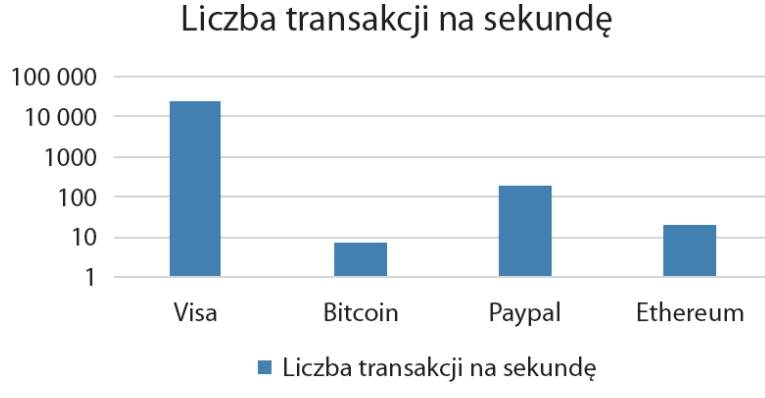
Rysunek 6.5. Portfel Trezor



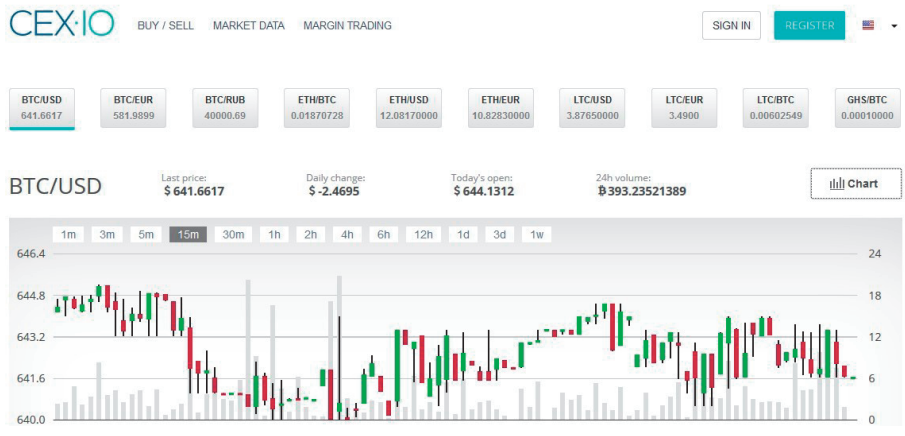
Rysunek 6.6. Portfel mobilny Jaxx

Rysunek 6.7. Logo informujące o akceptowaniu płatności w bitcoinach

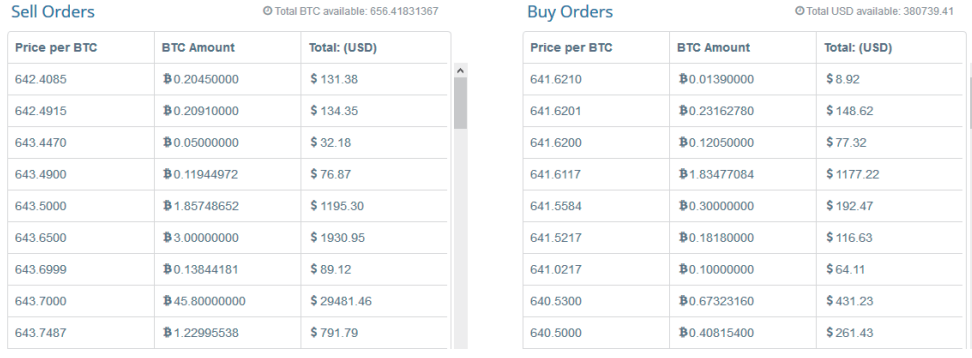
Rysunek 6.8. Terminal kasowy firmy 34 Bytes



Rysunek 6.9. Szybkość przetwarzania transakcji w sieci Bitcoina i w innych sieciach (skala logarytmiczna)

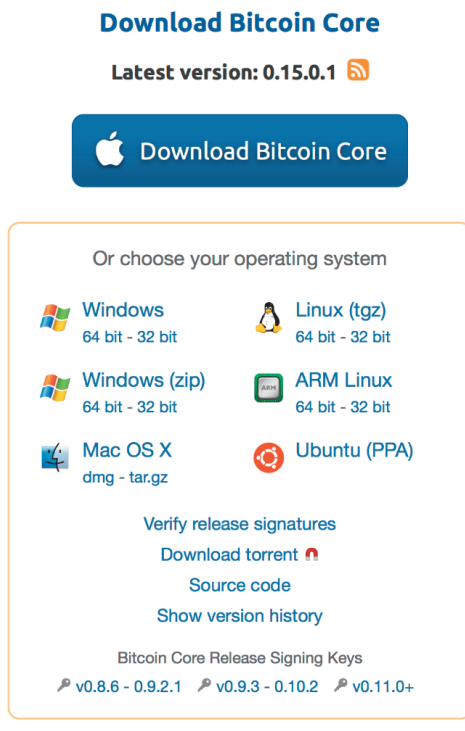


Rysunek 6.10. Przykładowa giełda bitcoinów cex.io

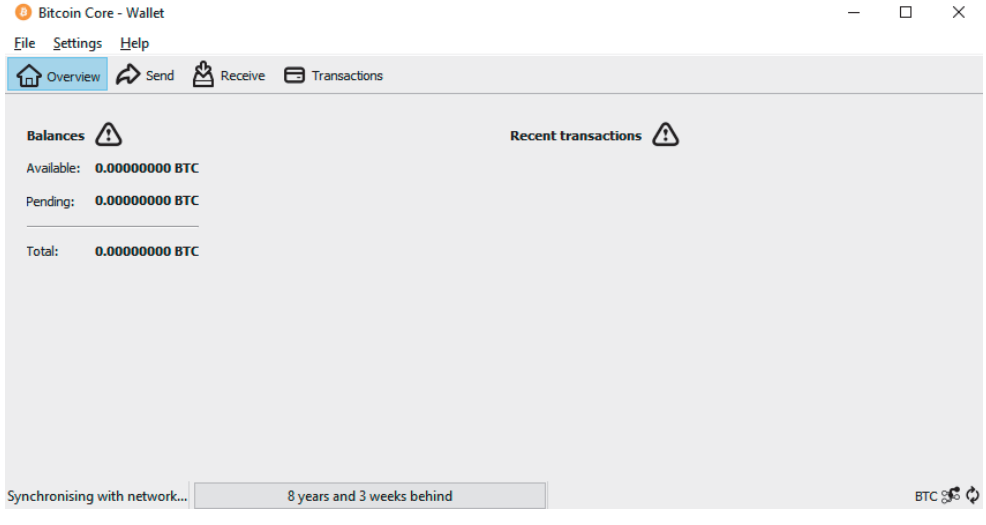


Rysunek 6.11. Przykładowy arkusz zleceń bitcoinów na giełdzie cex.io

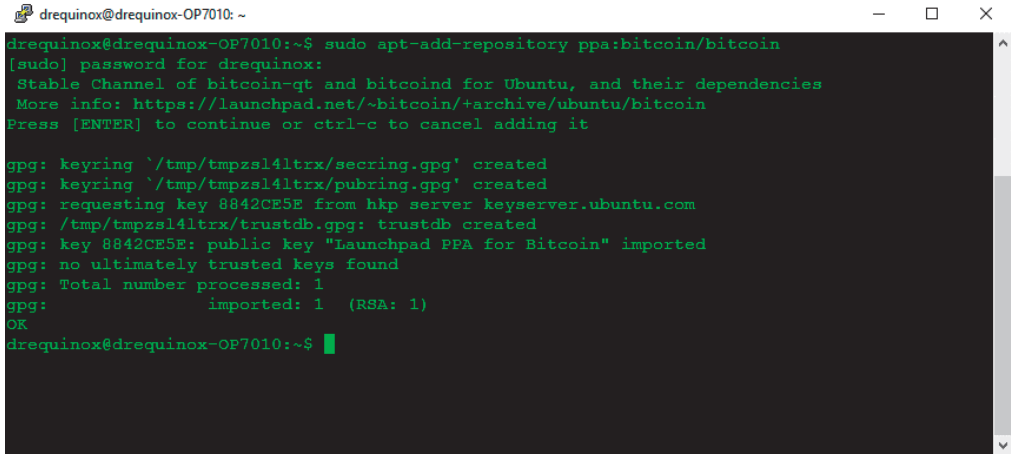
Rozdział 7. Klienty i interfejsy API Bitcoina



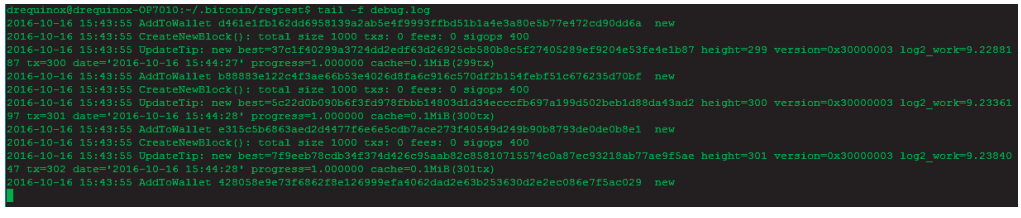
Rysunek 7.1. Pobieranie klienta Bitcoin Core



Rysunek 7.2. Klient Bitcoin Core QT po instalacji, informujący, że łańcuch bloków nie jest zsynchronizowany



Rysunek 7.3. Instalowanie Bitcoina



Rysunek 7.4. Komunikaty z dziennika debugowania Bitcoina

```
drequinox@drequinox-OP7010:~$ bitcoin-cli getinfo
{
  "version": 130000,
  "protocolversion": 70014,
  "walletversion": 130000,
  "balance": 0.00000000,
  "blocks": 433948,
  "timeoffset": 0,
  "connections": 8,
  "proxy": "",
  "difficulty": 258522748404.5154,
  "testnet": false,
  "keypoololdest": 1475534258,
  "keypoolsize": 100,
  "paytxfee": 0.00000000,
  "relayfee": 0.00001000,
  "errors": ""
}
drequinox@drequinox-OP7010:~$
```

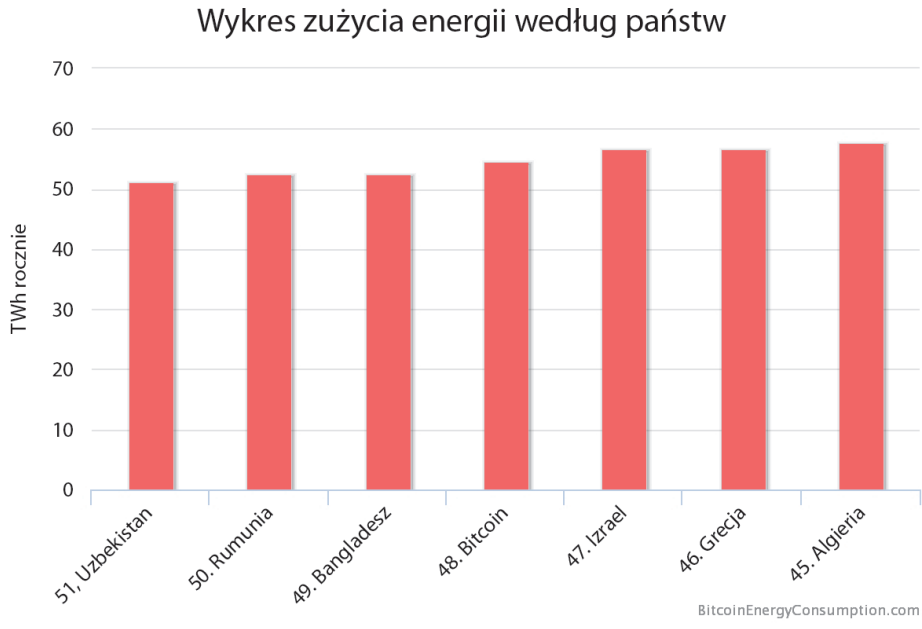
Rysunek 7.5. Przykładowe wywołanie instrukcji getinfo za pomocą interfejsu bitcoin-cli. Ten sam format można wykorzystać do wywoływania innych poleceń

```
drequinox@drequinox-OP7010:~$ bitcoin-cli -testnet help | more
== Blockchain ==
getbestblockhash
getblock "hash" ( verbose )
getblockchaininfo
getblockcount
getblockhash index
getblockheader "hash" ( verbose )
getchaintips
getdifficulty
getmempoolancestors txid (verbose)
getmempooldescendants txid (verbose)
getmempoolentry txid
getmempoolinfo
getrawmempool ( verbose )
gettxout "txid" n ( includemempool )
gettxoutproof ["txid",...] ( blockhash )
gettxoutsetinfo
verifychain ( checklevel numblocks )
verifytxoutproof "proof"

== Control ==
getinfo
help ( "command" )
stop
```

Rysunek 7.6. Używanie interfejsu bitcoin-cli do sieci testnet. Pokazano tu tylko kilka pierwszych wierszy danych wyjściowych. Rzeczywista lista zawiera wiele poleceń

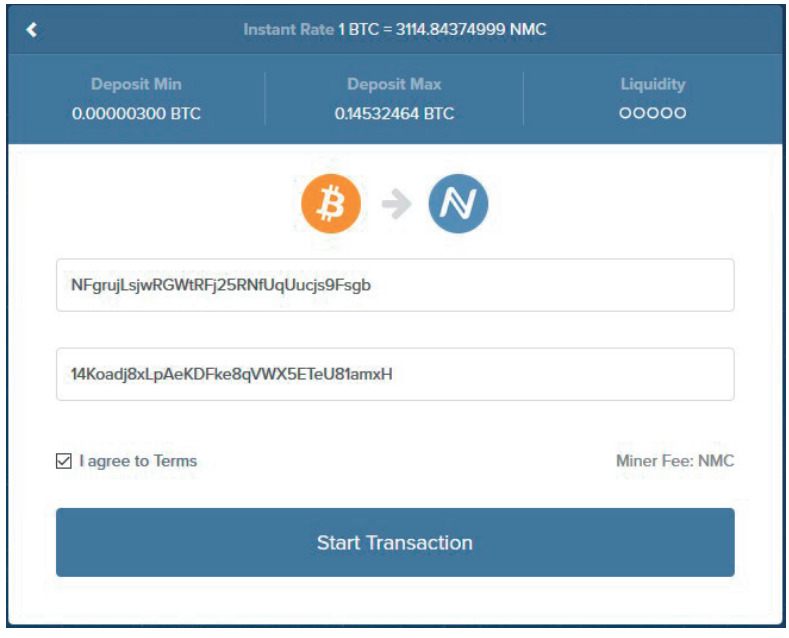
Rozdział 8. Inne kryptowaluty

Rysunek 8.1. Ten wykres pokazuje, że w czasie, gdy powstaje ta książka, łączna kapitalizacja rynku alternatywnych kryptowalut przekracza 200 miliardów dolarów

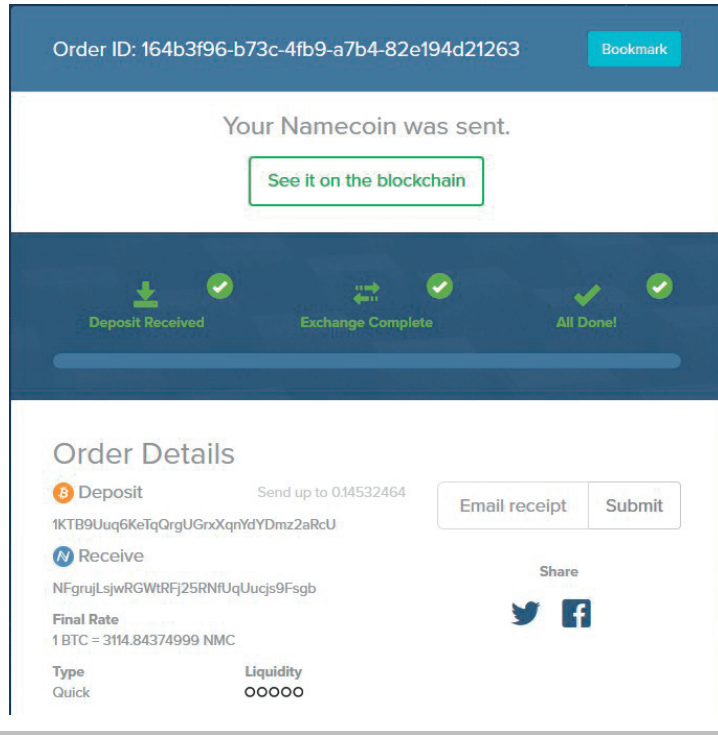


Rysunek 8.2. Zużycie energii według państw

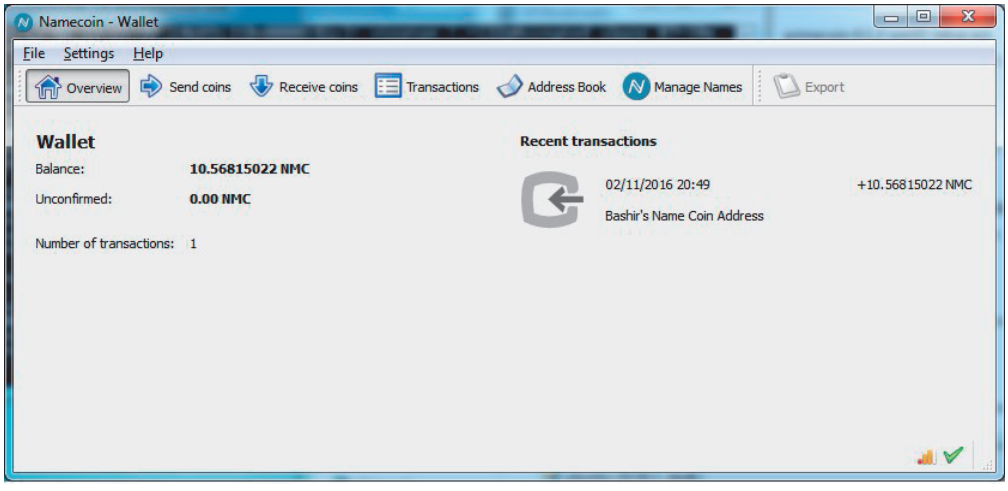
Rysunek 8.6. Poziom trudności w Namecoinie od grudnia 2016 r.
(za: https://bitinfocharts.com/comparison/difficulty-nmc.html)



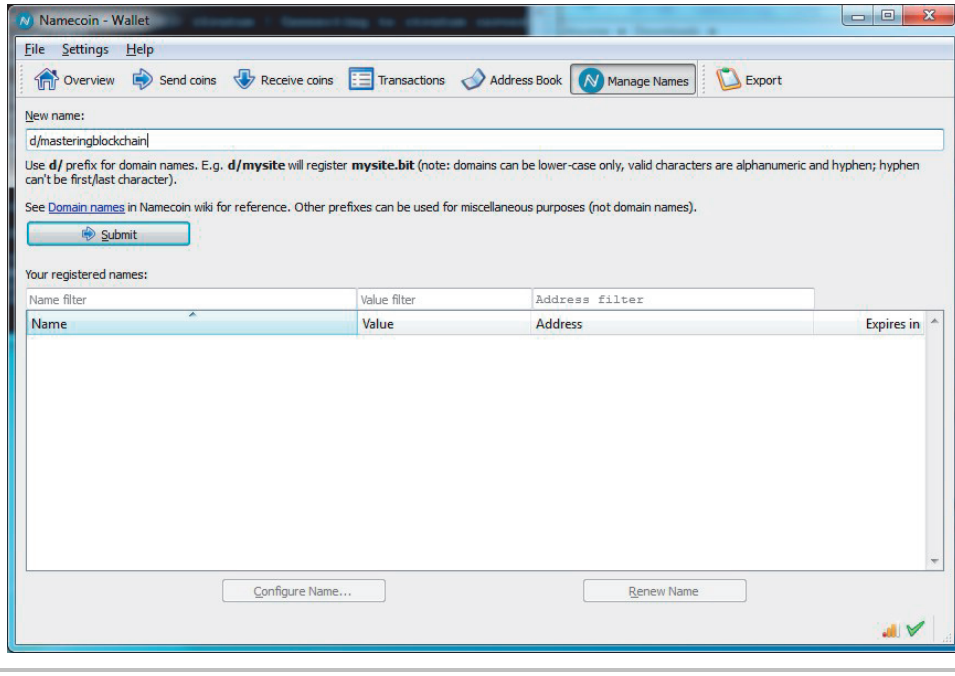
Rysunek 8.7. Wymiana bitcoinów na namecoiny



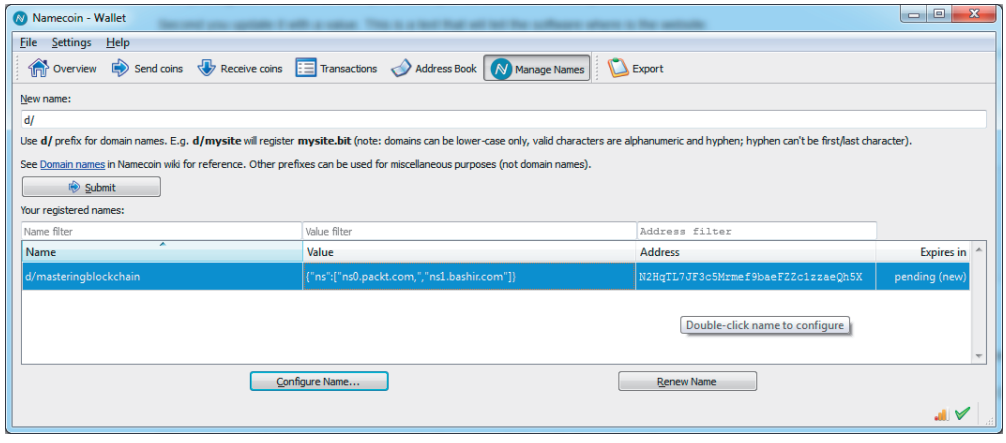
Rysunek 8.8. Powiadomienie o dostarczeniu namecoinów



Rysunek 8.9. Portfel Namecoina

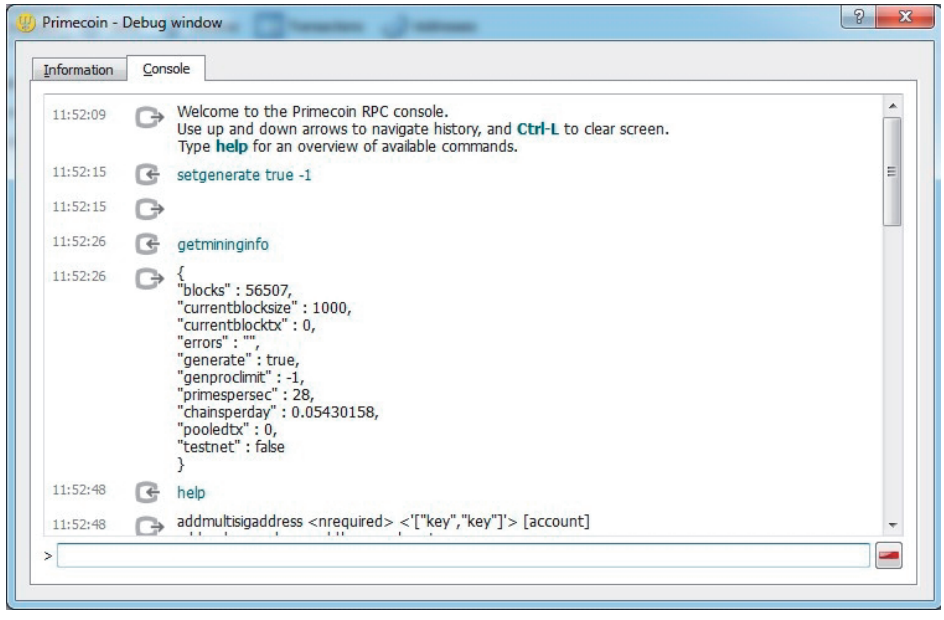


Rysunek 8.10. Portfel Namecoina — konfigurowanie nazwy domeny

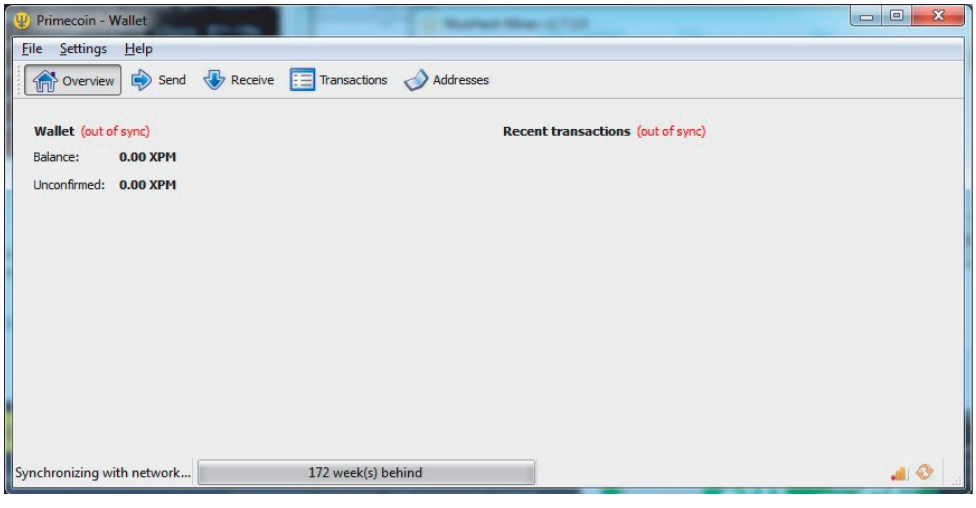


Rysunek 8.11. Portfel Namecoina — wyświetlanie zarejestrowanej nazwy

Rysunek 8.14. Wykres ze statystykami na temat Primecoina; źródło danych: https://coinmarketcap.com/currencies/primecoin/



Rysunek 8.15. Wydobywanie primecoinów



Rysunek 8.16. Oprogramowanie portfela Primecoina synchronizujące się z siecią

Rysunek 8.17. Wartość rynkowa i cena zcashów

```
drequinox@drequinox-OP7010:~$ git clone https://github.com/zcash/zcash.git
Cloning into 'zcash'...
remote: Counting objects: 56593, done.
remote: Total 56593 (delta 0), reused 0 (delta 0), pack-reused 56593
Receiving objects: 100% (56593/56593), 42.78 MiB | 2.11 MiB/s, done.
Resolving deltas: 100% (43020/43020), done.
Checking connectivity... done.
drequinox@drequinox-OP7010:~$ cd zcash/
drequinox@drequinox-OP7010:~/zcash$ git checkout v1.0.0
Note: checking out 'v1.0.0'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at 1feaefa... Update network magics for 1.0.0
```

Rysunek 8.18. Klonowanie repozytorium Zcasha z serwisu Git

```
drequinox@drequinox-OP7010:~/zcash$ ./zcutil/fetch-params.sh
Zcash - fetch-params.sh

This script will fetch the Zcash zkSNARK parameters and verify their
integrity with sha256sum.

The parameters are currently just under 911MB in size, so plan accordingly
for your bandwidth constraints. If the files are already present and
have the correct sha256sum, no networking is used.

Creating params directory. For details about this directory, see:
/home/drequinox/.zcash-params/README

Retrieving: https://z.cash/downloads/sprout-proving.key
--2016-10-28 21:46:21--  https://z.cash/downloads/sprout-proving.key
Resolving z.cash (z.cash)... 104.236.171.172
Connecting to z.cash (z.cash)|104.236.171.172|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://s3.amazonaws.com/zcashfinalmpc/sprout-proving.key [following]
--2016-10-28 21:46:22--  https://s3.amazonaws.com/zcashfinalmpc/sprout-proving.key
Resolving s3.amazonaws.com (s3.amazonaws.com)... 54.231.40.114
Connecting to s3.amazonaws.com (s3.amazonaws.com)|54.231.40.114|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 910173851 (868M) [application/octet-stream]
Saving to: '/home/drequinox/.zcash-params/sprout-proving.key.dl'

     0K ........ ........ ........ ........  3% 2.71M 5m8s
 32768K ........ ........ ........ ........  7% 3.58M 4m20s
 65536K ........ ........ ........ ........ 11% 2.53M 4m28s
 98304K ........ ........ ........ ........ 14% 1.75M 4m59s
131072K ........ ........ ........
```

Rysunek 8.19. Pobieranie parametrów dowodów ZK-SNARK w trakcie konfigurowania Zcasha

```
drequinox@drequinox-OP7010:~/zcash/src$ ./zcash-cli getinfo
{
    "version" : 1000050,
    "protocolversion" : 170002,
    "walletversion" : 60000,
    "balance" : 0.00000000,
    "blocks" : 601,
    "timeoffset" : 0,
    "connections" : 8,
    "proxy" : "",
    "difficulty" : 13748.56014152,
    "testnet" : false,
    "keypoololdest" : 1477688856,
    "keypoolsize" : 101,
    "paytxfee" : 0.00000000,
    "relayfee" : 0.00005000,
    "errors" : "WARNING: abnormally high number of blocks generated, 190 blocks received in the last 4 hours (96 expected)"
}
drequinox@drequinox-OP7010:~/zcash/src$
```

Rysunek 8.20. Zrzut ilustrujący dane wyjściowe instrukcji getinfo

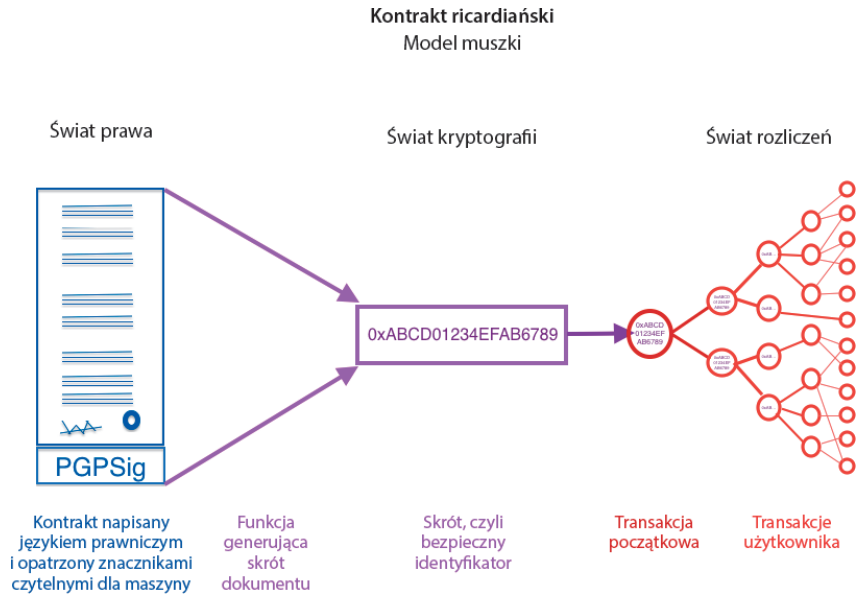
```
drequinox@drequinox-OP7010:~/nheqminer/nheqminer/build$ ./nheqminer -l eu -u 1PL6gsm49xCFMvrXqgGcee5cdrG119GoWN.worker1 -t 6 -od 0
Equihash CPU Miner for NiceHash v0.1c
Thanks to Zcash developers for providing most of the code
Special thanks to tromp for providing optimized CPU equihash solver

Setting log level to 2
[09:28:53][0x00007f51009cd700] stratum | Connecting to stratum server equihash.eu.nicehash.com:3357
[09:28:53][0x00007f51009cd700] stratum | Connected!
[09:28:53][0x00007f51009cd700] stratum | Starting miner
[09:28:53][0x00007f50fafce700] miner#1 | Starting thread #1
[09:28:53][0x00007f50fb7cf700] miner#0 | Starting thread #0
[09:28:53][0x00007f50f8fca700] miner#5 | Starting thread #5
[09:28:53][0x00007f50fa7cd700] miner#2 | Starting thread #2
[09:28:53][0x00007f50f97cb700] miner#4 | Starting thread #4
[09:28:53][0x00007f50f9fcc700] miner#3 | Starting thread #3
[09:28:54][0x00007f51009cd700] stratum | Subscribed to stratum server
[09:28:54][0x00007f51009cd700] miner | Extranonce is 5000e5b8000000000000000005000e5b9ab
[09:28:54][0x00007f51009cd700] stratum | Authorized worker 1PL6gsm49xCFMvrXqgGcee5cdrG119GoWN.worker1
[09:28:54][0x00007f51009cd700] stratum | Target set to 01e1e1e1e1e0000000000000000000000000000000000000000000000000000000
[09:28:54][0x00007f51009cd700] stratum | Received new job #000000329b82d287
[09:28:55][0x00007f50fa7cd700] stratum | Submitting share #4, nonce 020000000000000000000000000000
[09:28:55][0x00007f51009cd700] stratum | Accepted share #4
[09:28:55][0x00007f51009cd700] stratum | Ignoring non-clean job #000000329b82d2cc
[09:28:57][0x00007f50fafce700] stratum | Submitting share #5, nonce 010000000000000000000000000001
[09:28:57][0x00007f51009cd700] stratum | Accepted share #5
[09:28:59][0x00007f50f97cb700] stratum | Submitting share #6, nonce 040000000000000000000000000005
```

Rysunek 8.21. Używanie adresu Bitcoina do otrzymywania płatności za sprzedaż mocy obliczeniowej

```
C:\nheqminer_v0.3a>nheqminer_zcash.exe -l eu -u t1YiNeyxoLZcjDHnaMtq5WLEbKquTY11gK8.miner1 -t 6 -od 0

        ========== www.nicehash.com ==========
            Equihash CPU&GPU Miner for NiceHash v0.3a
        Thanks to Zcash developers for providing base of the code.
            Special thanks to tromp and xenoncat for providing
                optimized CPU and CUDA equihash solvers.
        ========== www.nicehash.com ==========

Setting log level to 2
[2016-10-29 22:37:18.196024][0x00001404]: Using SSE2: YES
[2016-10-29 22:37:18.203024][0x00001404]: Using AVX: YES
[2016-10-29 22:37:18.207025][0x00001404]: Using AVX2: YES
[2016-10-29 22:37:18.222026][0x00001590]: stratum | Starting miner
[2016-10-29 22:37:18.228026][0x000015c4]: miner#0 | Starting thread #0 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:18.228026][0x00000524]: miner#1 | Starting thread #1 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:18.243027][0x00000b20]: miner#3 | Starting thread #3 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:18.243027][0x00001570]: miner#2 | Starting thread #2 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:20.354147][0x00001330]: miner#5 | Starting thread #5 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:20.354147][0x000015f8]: miner#4 | Starting thread #4 (CPU-XENONCAT-AVX2)
[2016-10-29 22:37:20.354147][0x00001590]: stratum | Connecting to stratum server stratum.eu.zcash.nicehash.com:3357
[2016-10-29 22:37:20.354147][0x00001108]: miner#6 | Starting thread #6 (OCL_XMP) TODO
<info> found 1 devices
Using device 0 as GPU 0
<info> compiling ...
<info> source: 41162 bytes
[2016-10-29 22:37:28.796630][0x00001590]: stratum | Connected!
[2016-10-29 22:37:28.892636][0x00001590]: stratum | Subscribed to stratum server
[2016-10-29 22:37:28.899636][0x00001590]: miner | Extranonce is 1ffff7da0000000000000001ffff7d9
[2016-10-29 22:37:28.999642][0x00001590]: stratum | ←[35mTarget set to 03c3c3c0000000000000000000000000000000000000000000000000000000←[0m
[2016-10-29 22:37:29.016643][0x00001590]: stratum | ←[36mReceived new job #6←[0m
[2016-10-29 22:37:29.290659][0x00001590]: stratum | Authorized worker t1YiNeyxoLZcjDHnaMtq5WLEbKquTY11gK8.miner1
[2016-10-29 22:37:33.311889][0x00001404]: ←[33mSpeed [300 sec]: 0.154357 I/s, 0.231535 Sols/s←[0m
[2016-10-29 22:37:40.476298][0x00000524]: stratum | Submitting share #4, nonce 0100000000000000000000000000004
[2016-10-29 22:37:40.605306][0x00001590]: stratum | ←[32mAccepted share #4←[0m
[2016-10-29 22:37:43.411466][0x00001404]: ←[33mSpeed [300 sec]: 0.824045 I/s, 1.43124 Sols/s←[0m
```

Rysunek 8.22. Używanie adresu T Zcasha do otrzymywania płatności za sprzedaż mocy obliczeniowej

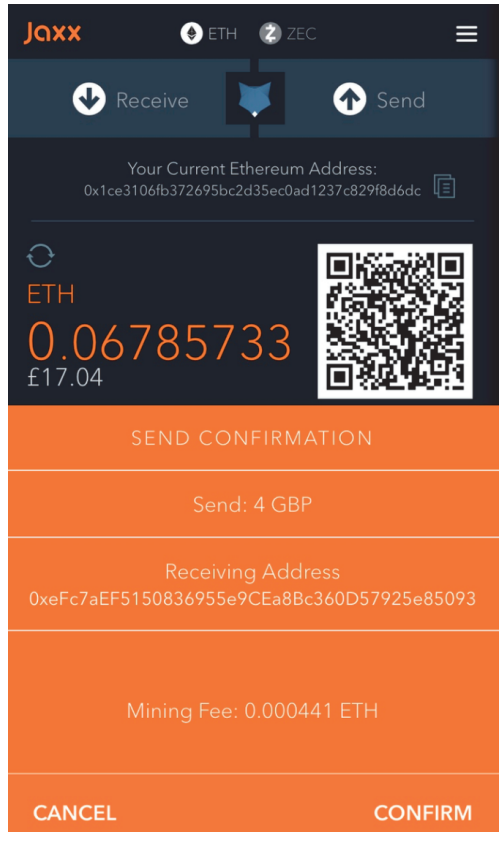
Rozdział 9. Inteligentne kontrakty



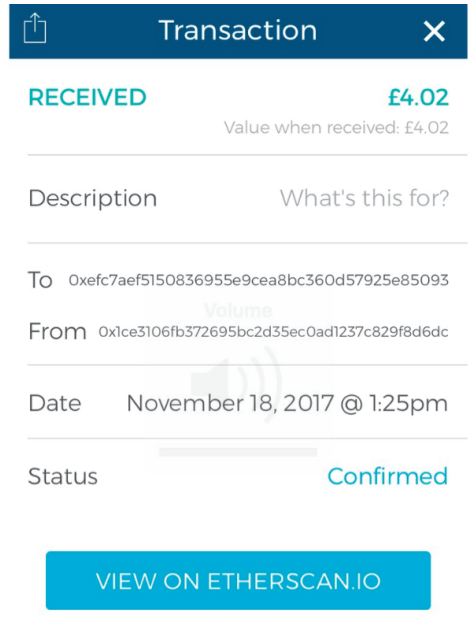
Rysunek 9.1. Kontrakty ricardiańskie, model muszki

Rysunek 9.2. Rysunek pokazujący, że kwestie semantyki i wykonywania są niezależne od siebie (za Ianem Griggiem). Ilustrację zmodyfikowano, aby na osiach pokazać inne rodzaje kontraktów

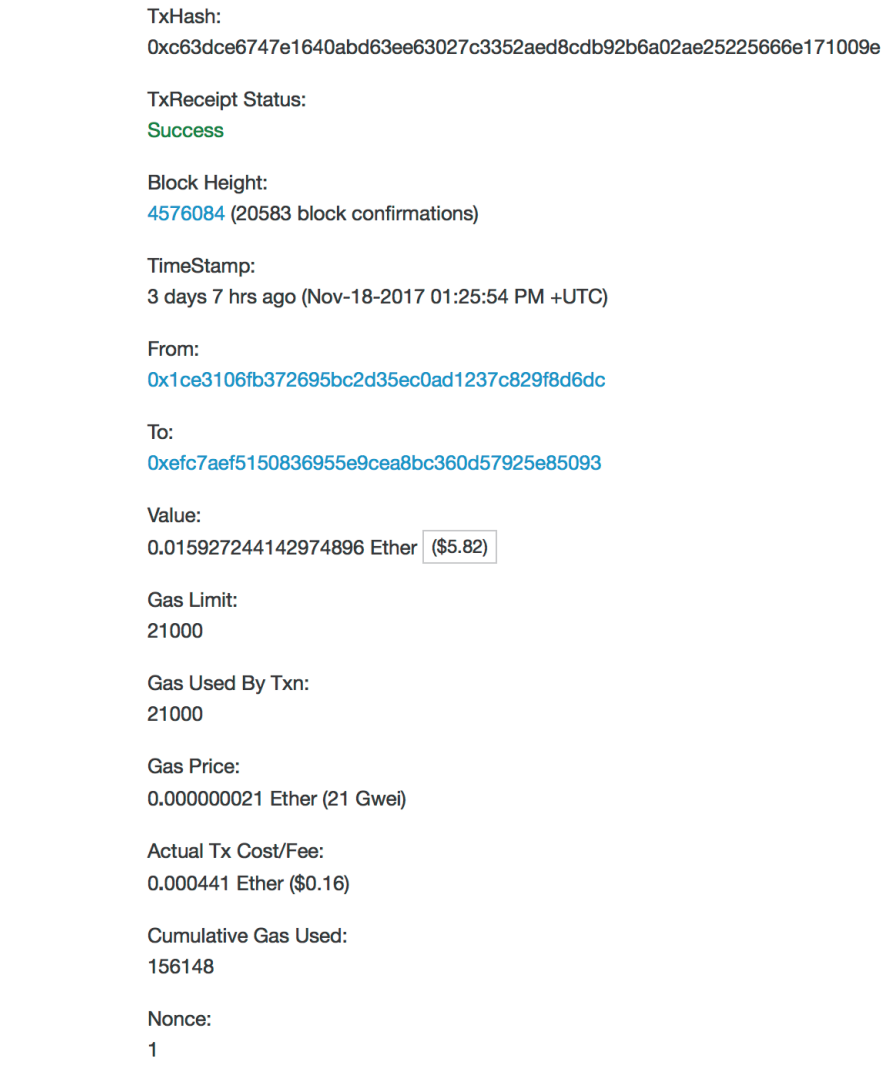
Rozdział 10. ABC łańcucha bloków Ethereum



Rysunek 10.3. Przesyłanie przelewu środków od Bashira w portfelu Jaxx



Rysunek 10.4. Transakcja w portfelu łańcucha bloków należącym do Irshad



Rysunek 10.5. Eksplorator bloków z łańcucha bloków Ethereum

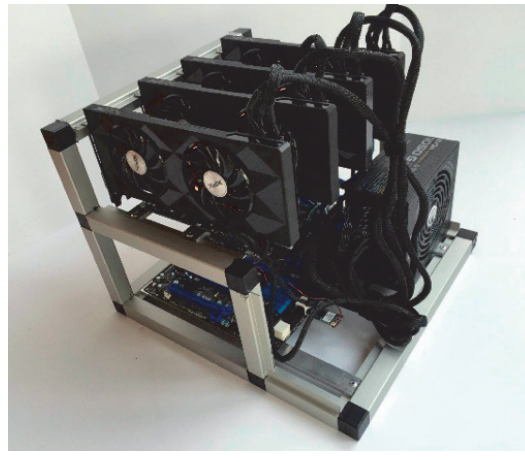
Rozdział 11. Jeszcze o Ethereum

```
drequinox@drequinox-OP7010:~$ ethminer -G
[OPENCL]:No OpenCL platforms found
No GPU device with sufficient memory was found. Can't GPU mine. Remove the -G argument
drequinox@drequinox-OP7010:~$
```

Rysunek 11.3. Błąd zwracany w sytuacji, gdy nie można znaleźć odpowiednich kart graficznych

```
drequinox@drequinox-OP7010:~$ ethminer -M -C
  ◇  22:43:30.560 ethminer  #00004000…
Benchmarking on platform: 8-thread CPU
Preparing DAG...
  □  22:43:30.561 miner0  Loading full DAG of seedhash: #00000000…
Warming up...
Trial 1... 0
Trial 2... DAG  22:43:38.310 miner0  Generating DAG file. Progress: 0 %
0
Trial 3... 0
Trial 4... DAG  22:43:45.336 miner0  Generating DAG file. Progress: 1 %
0
```

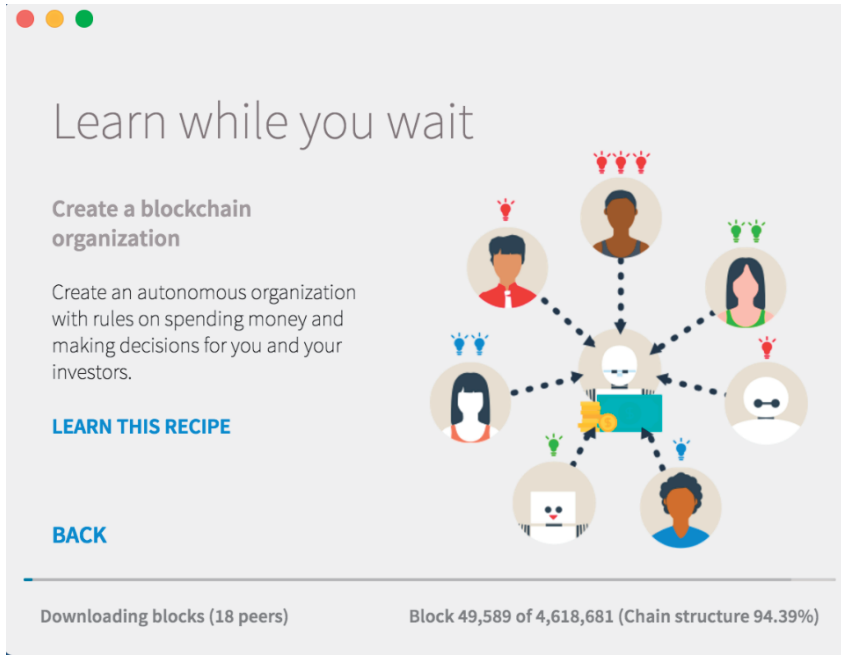
Rysunek 11.4. Testy porównawcze z użyciem zwykłych procesorów



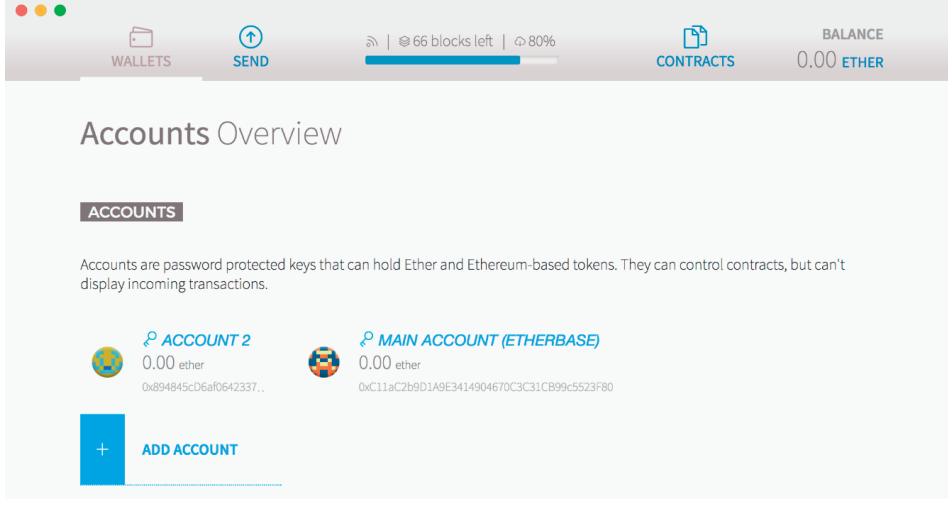
Rysunek 11.5. Koparka Ethereum wystawiona na sprzedaż w serwisie eBay

```
drequinox@drequinox-OP7010:~$  ethminer -C -F http://ethereumpool.co/?miner=0.1@0x024a20cc5feba7f3dc3776075b3e60c20eb1459c@DrEquinox
miner  23:50:52.046 ethminer  Getting work package...
```

Rysunek 11.6. Zrzut z programu Ethminer



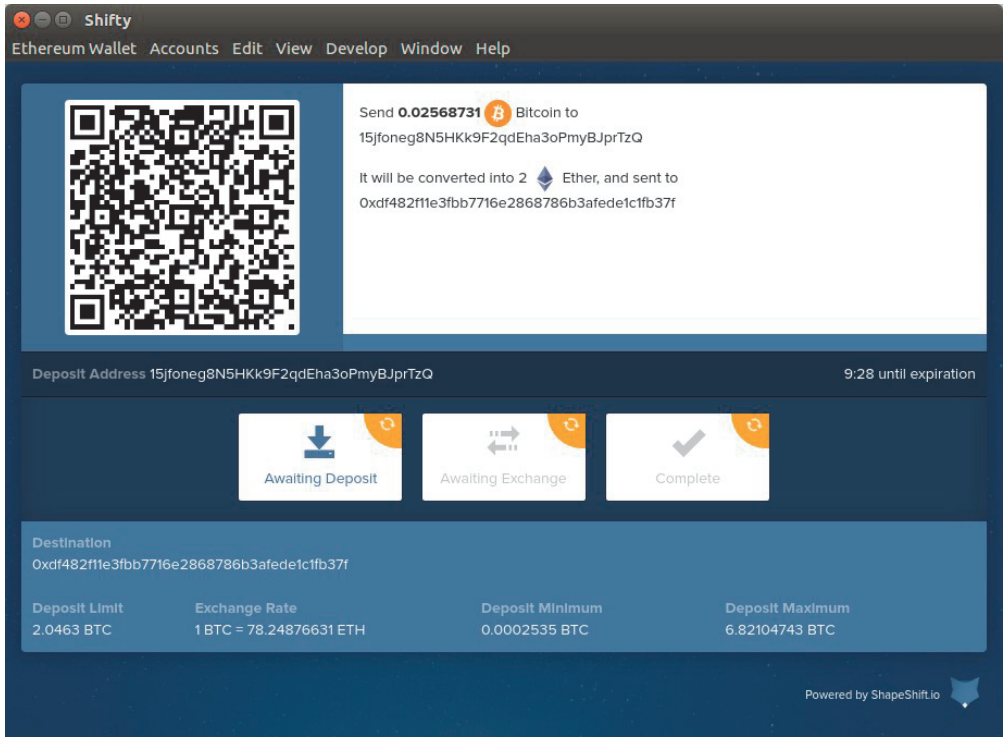
Rysunek 11.7. Przeglądarka Mist w trakcie uruchamiania i synchronizowania się z siecią mainnet



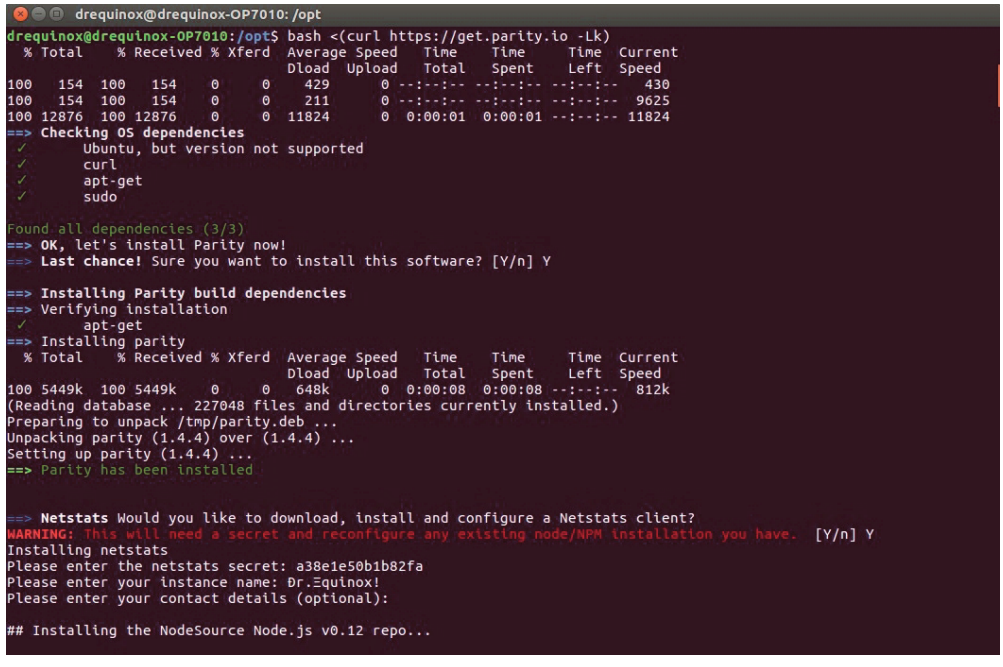
Rysunek 11.8. Przeglądarka Mist

```
drequinox@drequinox-OP7010:~$ geth attach
Welcome to the Geth JavaScript console!

instance: Parity//v1.4.4-beta-a68d52c-20161118/x86_64-linux-gnu/rustc1.13.0
coinbase: 0x0000000000000000000000000000000000000000
at block: 2718377 (Tue, 29 Nov 2016 22:52:52 GMT)
 modules: eth:1.0 net:1.0 parity:1.0 parity_accounts:1.0 personal:1.0 rpc:1.0 traces:1.0 web3:1.0

>
```

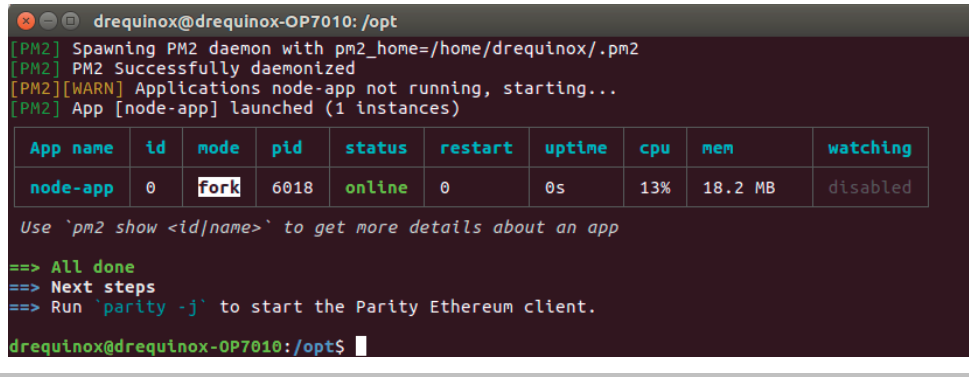
Rysunek 11.10. Klient Geth



Rysunek 11.11. Wymiana bitcoinów na ethery



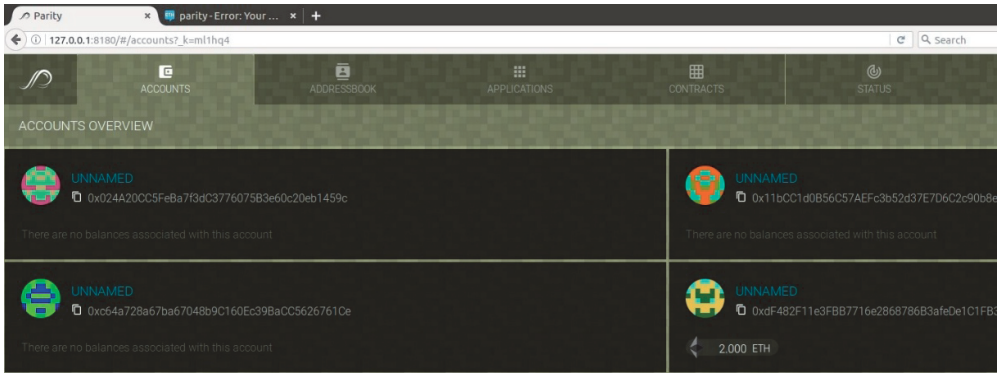
Rysunek 11.12. Instalowanie klienta Parity



Rysunek 11.13. Uruchamianie klienta Parity

Rysunek 11.14. Statystyki na temat sieci Ethereum w witrynie ethstats.net

Rysunek 11.15. Klienty wymienione na stronie https://ethstats.net/



Rysunek 11.16. Interfejs użytkownika klienta Parity

Rysunek 11.18. Wartość rynkowa etherów w przeszłości (źródło: etherscan.io)

Rozdział 12. Środowisko programistyczne Ethereum

```
imran@drequinox-OP7010:~$ geth --testnet
I1204 16:03:32.759308 cmd/utils/flags.go:613] WARNING: No etherbase set and no accounts found as default
I1204 16:03:32.759415 ethdb/database.go:83] Allotted 128MB cache and 1024 file handles to /home/imran/.ethereum/testnet/geth/chaindata
I1204 16:03:32.807292 ethdb/database.go:176] closed db:/home/imran/.ethereum/testnet/geth/chaindata
I1204 16:03:32.807589 node/node.go:175] instance: Geth/v1.5.2-stable-c8695209/linux/go1.7.3
I1204 16:03:32.807603 ethdb/database.go:83] Allotted 128MB cache and 1024 file handles to /home/imran/.ethereum/testnet/geth/chaindata
I1204 16:03:32.814016 eth/backend.go:280] Successfully wrote custom genesis block: 0cd786a2425d16f152c658316c423e6ce1181e15c3295826d7c99
04cba9ce303
I1204 16:03:32.814076 eth/db_upgrade.go:346] upgrading db log bloom bins
I1204 16:03:32.814112 eth/db_upgrade.go:354] upgrade completed in 36.513µs
I1204 16:03:32.814128 eth/backend.go:193] Protocol Versions: [63 62], Network Id: 2
I1204 16:03:32.814363 core/blockchain.go:214] Last header: #0 [0cd786a2…] TD=131072
I1204 16:03:32.814375 core/blockchain.go:215] Last block: #0 [0cd786a2…] TD=131072
I1204 16:03:32.814382 core/blockchain.go:216] Fast block: #0 [0cd786a2…] TD=131072
I1204 16:03:32.814840 p2p/server.go:336] Starting Server
I1204 16:03:37.983847 p2p/discover/udp.go:217] Listening, enode://fa838ec3fee8a26d75755b55f7cbdd80efacc4a98b5291acd5a23aea5465b794c84aff
e7be633524d2895768a2122a25e87cf97bd369895ace9f48f868eaef18@[::]:30303
I1204 16:03:37.983960 p2p/server.go:604] Listening on [::]:30303
I1204 16:03:37.984963 node/node.go:340] IPC endpoint opened: /home/imran/.ethereum/testnet/geth.ipc
I1204 16:04:17.984160 eth/downloader/downloader.go:326] Block synchronisation started
```

Rysunek 12.1. Dane wyjściowe z polecenia nawiązania połączenia z siecią testową Ethereum w kliencie geth

```
WARN [12-13|19:19:11] No etherbase set and no accounts found as default
INFO [12-13|19:19:11] Allocated cache and file handles         database=/Users/drequinox/etherprivate/geth/chaindata cache=16 handles=16
INFO [12-13|19:19:11] Writing custom genesis block
INFO [12-13|19:19:11] Successfully wrote genesis state         database=chaindata                                     hash=6650a0…b5c158
INFO [12-13|19:19:11] Allocated cache and file handles         database=/Users/drequinox/etherprivate/geth/lightchaindata cache=16 handles=16
INFO [12-13|19:19:11] Writing custom genesis block
INFO [12-13|19:19:11] Successfully wrote genesis state         database=lightchaindata                                hash=6650a0…b5c158
```

Rysunek 12.2. Inicjowanie sieci prywatnej

```
WARN [12-13|19:20:11] No etherbase set and no accounts found as default
INFO [12-13|19:20:11] Starting peer-to-peer node               instance=Geth/v1.7.3-stable-4bb3c89d/darwin-amd64/go1.9.2
INFO [12-13|19:20:11] Allocated cache and file handles         database=/Users/drequinox/etherprivate/geth/chaindata cach
WARN [12-13|19:20:11] Upgrading database to use lookup entries
INFO [12-13|19:20:11] Initialised chain configuration          config="{ChainID: 786 Homestead: 0 DAO: <nil> DAOSupport:
INFO [12-13|19:20:11] Disk storage enabled for ethash caches   dir=/Users/drequinox/etherprivate/geth/ethash count=3
INFO [12-13|19:20:11] Disk storage enabled for ethash DAGs     dir=/Users/drequinox/.ethash                   count=2
INFO [12-13|19:20:11] Initialising Ethereum protocol           versions="[63 62]" network=786
INFO [12-13|19:20:11] Database deduplication successful        deduped=0
INFO [12-13|19:20:11] Loaded most recent local header          number=0 hash=6650a0…b5c158 td=1024
INFO [12-13|19:20:11] Loaded most recent local full block      number=0 hash=6650a0…b5c158 td=1024
INFO [12-13|19:20:11] Loaded most recent local fast block      number=0 hash=6650a0…b5c158 td=1024
INFO [12-13|19:20:11] Regenerated local transaction journal    transactions=0 accounts=0
INFO [12-13|19:20:11] Starting P2P networking
INFO [12-13|19:20:13] UDP listener up                          self=enode://5c53ec0755806bc92432728f22a55f169a8c63df307fc
7a55d635@86.15.44.209:30303
INFO [12-13|19:20:13] RLPx listener up                         self=enode://5c53ec0755806bc92432728f22a55f169a8c63df307fc
7a55d635@86.15.44.209:30303
INFO [12-13|19:20:13] IPC endpoint opened: /Users/drequinox/etherprivate/geth.ipc
INFO [12-13|19:20:14] Mapped network port                      proto=udp extport=30303 intport=30303 interface="UPNP IGDv
INFO [12-13|19:20:14] Mapped network port                      proto=tcp extport=30303 intport=30303 interface="UPNP IGDv
```

Rysunek 12.3. Uruchamianie klienta geth dla sieci prywatnej

```
I0211 23:58:50.380089 eth/backend.go:479] Automatic pregeneration of ethash DAG ON (ethash dir: /home/imran/.e
thash)
I0211 23:58:50.380097 miner/miner.go:136] Starting mining operation (CPU=2 TOT=3)
I0211 23:58:50.380138 eth/backend.go:486] checking DAG (ethash dir: /home/imran/.ethash)
I0211 23:58:50.380257 miner/worker.go:542] commit new work on block 1 with 0 txs & 0 uncles. Took 139.49µs
I0211 23:58:50.380292 vendor/github.com/ethereum/ethash/ethash.go:259] Generating DAG for epoch 0 (size 107373
9904) (0000000000000000000000000000000000000000000000000000000000000000)
I0211 23:58:51.166755 vendor/github.com/ethereum/ethash/ethash.go:276] Done generating DAG for epoch 0, it too
k 786.458657ms
```

Rysunek 12.5. Generowanie acyklicznego grafu skierowanego

```
I1204 22:38:02.373804 miner/worker.go:438] ⛏ ≃  Mined 5 blocks back: block #487
I1204 22:38:02.373908 miner/worker.go:542] commit new work on block 493 with 0 txs & 0 uncles. Took 86.005µs
I1204 22:38:02.637297 miner/worker.go:344] ⛏  Mined block (#493 / 9a95245e). Wait 5 blocks for confirmation
I1204 22:38:02.637415 miner/worker.go:542] commit new work on block 494 with 0 txs & 0 uncles. Took 91.009µs
I1204 22:38:02.637436 miner/worker.go:438] ⛏ ≃  Mined 5 blocks back: block #488
I1204 22:38:02.639064 miner/worker.go:542] commit new work on block 494 with 0 txs & 0 uncles. Took 1.609044ms
I1204 22:38:03.538525 miner/worker.go:344] ⛏  Mined block (#494 / cb89cccd). Wait 5 blocks for confirmation
I1204 22:38:03.538719 miner/worker.go:542] commit new work on block 495 with 0 txs & 0 uncles. Took 158.751µs
I1204 22:38:03.538745 miner/worker.go:438] ⛏ ≃  Mined 5 blocks back: block #489
I1204 22:38:03.538860 miner/worker.go:542] commit new work on block 495 with 0 txs & 0 uncles. Took 95.822µs
I1204 22:38:03.548923 miner/worker.go:344] ⛏  Mined block (#495 / 539d8079). Wait 5 blocks for confirmation
I1204 22:38:03.549064 miner/worker.go:542] commit new work on block 496 with 0 txs & 0 uncles. Took 120.447µs
I1204 22:38:03.549082 miner/worker.go:438] ⛏ ≃  Mined 5 blocks back: block #490
I1204 22:38:03.549159 miner/worker.go:542] commit new work on block 496 with 0 txs & 0 uncles. Took 64.047µs
```

Rysunek 12.6. Dane wyjściowe w procesie wydobywania bloków

```
>
Array               Math                TypeError           constructor         hasOwnProperty      parseFloat          toString
BigNumber           NaN                 URIError            debug               inspect             parseInt            txpool
Boolean             Number              Web3                decodeURI           isFinite            personal            undefined
Date                Object              _setInterval        decodeURIComponent  isNaN               propertyIsEnumerable unescape
Error               RangeError          _setTimeout         encodeURI           isPrototypeOf       require             valueOf
EvalError           ReferenceError      admin               encodeURIComponent  jeth                rpc                 web3
Function            RegExp              clearInterval       escape              loadScript          setInterval
Infinity            String              clearTimeout        eth                 miner               setTimeout
JSON                SyntaxError         console             eval                net                 toLocaleString
>
```

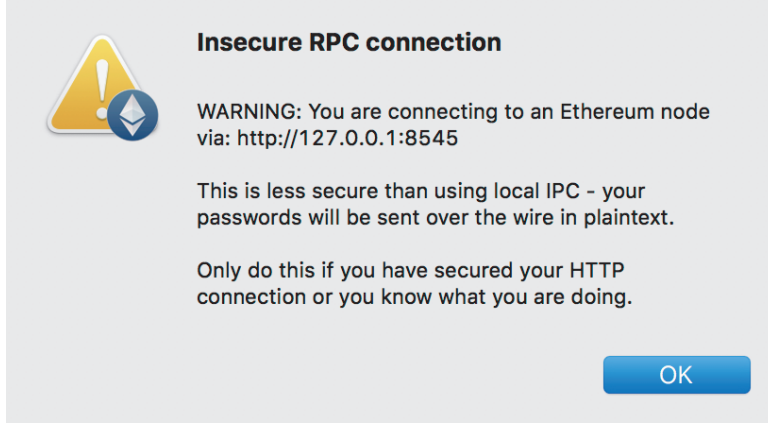
Rysunek 12.7. Dostępne obiekty

```
> personal.
personal._requestManager personal.getListAccounts personal.lockAccount     personal.sign
personal.constructor      personal.importRawKey    personal.newAccount      personal.unlockAccount
personal.ecRecover        personal.listAccounts    personal.sendTransaction
> net.
net._requestManager net.getListening    net.getVersion      net.peerCount
net.constructor     net.getPeerCount    net.listening       net.version
```

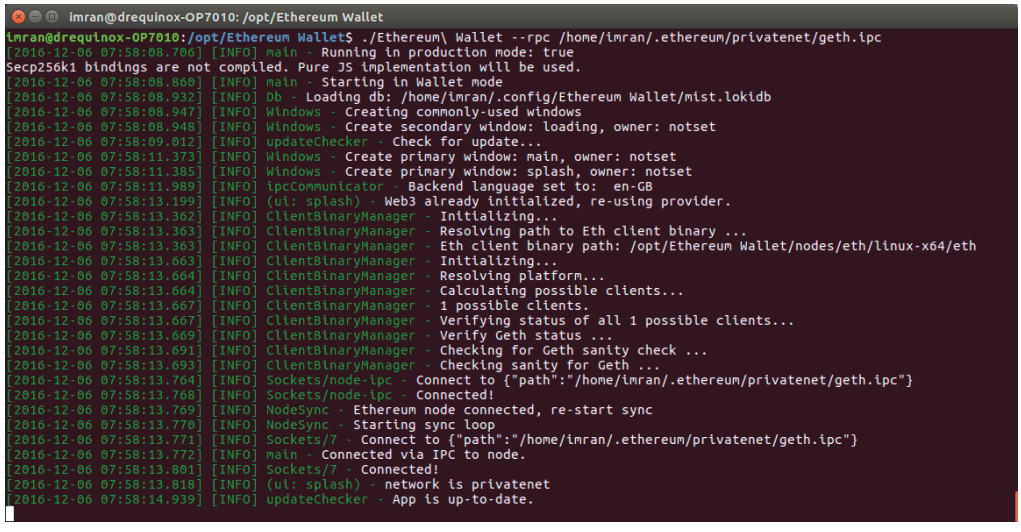
Rysunek 12.8. Dostępne metody

```
> net;
{
  listening: true,
  peerCount: 0,
  version: "786",
  getListening: function(callback),
  getPeerCount: function(callback),
  getVersion: function(callback)
}
>
```

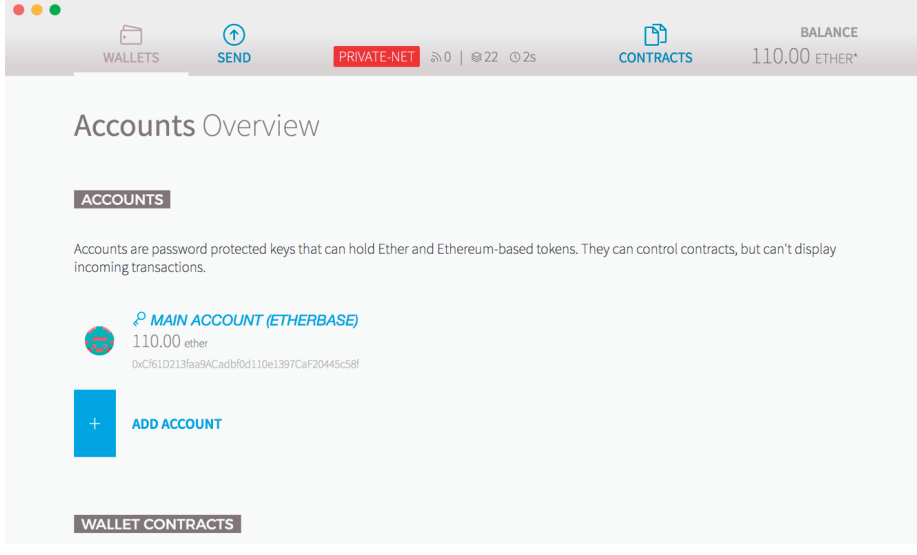
Rysunek 12.9. Lista metod



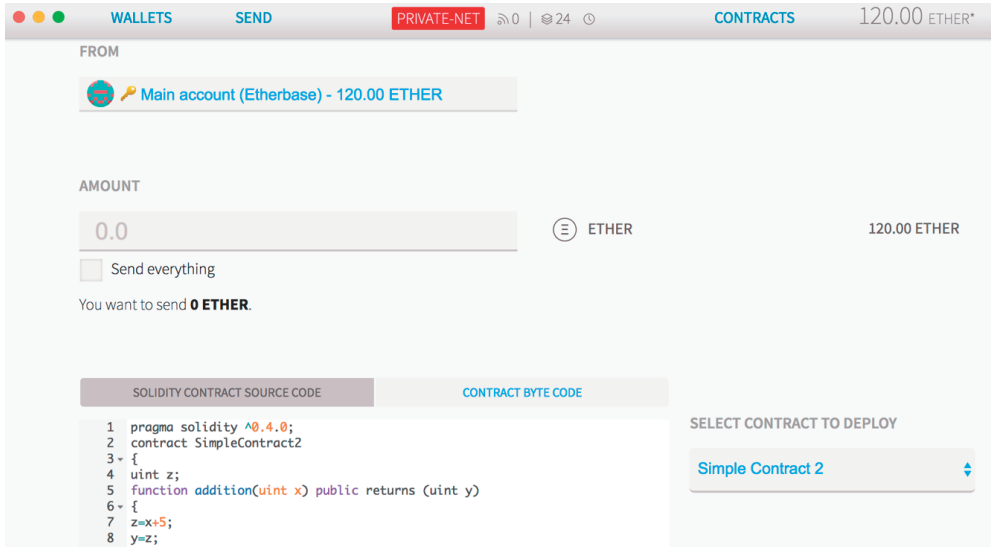
Rysunek 12.10. Niezabezpieczone połączenie RPC



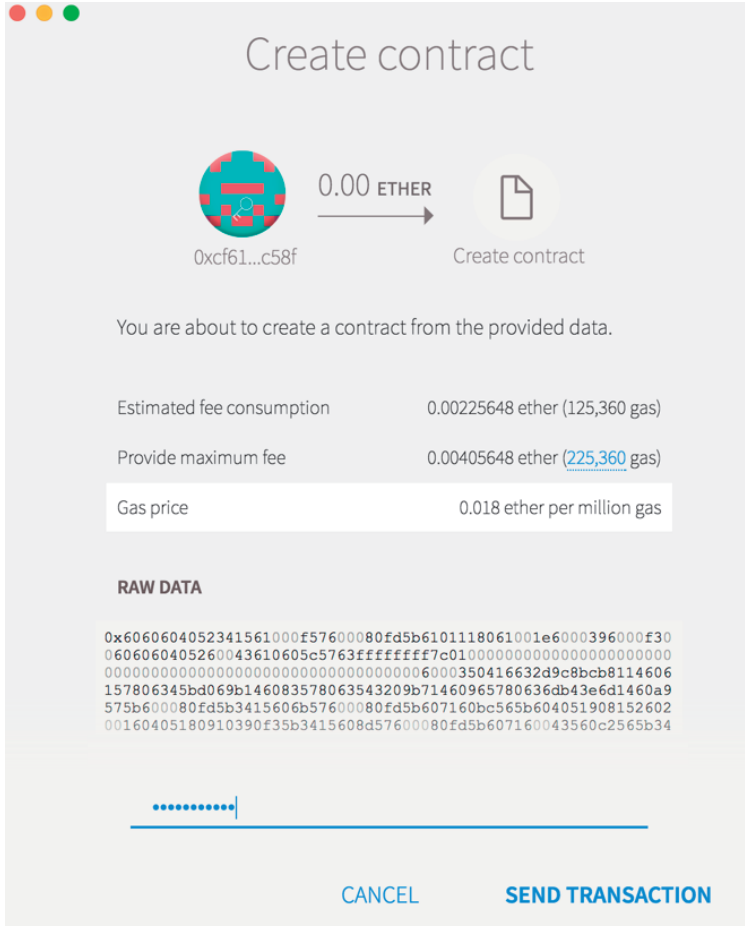
Rysunek 12.11. Uruchamianie portfela Ethereum łączącego się z prywatną siecią za pomocą komunikacji IPC



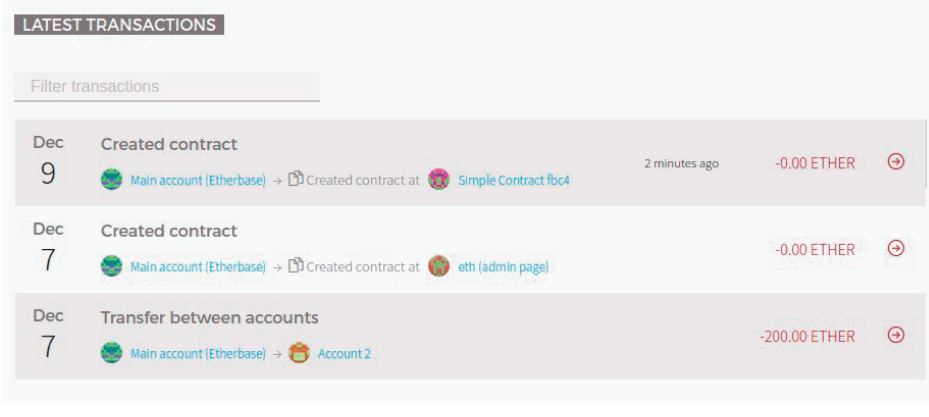
Rysunek 12.12. Przeglądarka Mist w sieci prywatnej



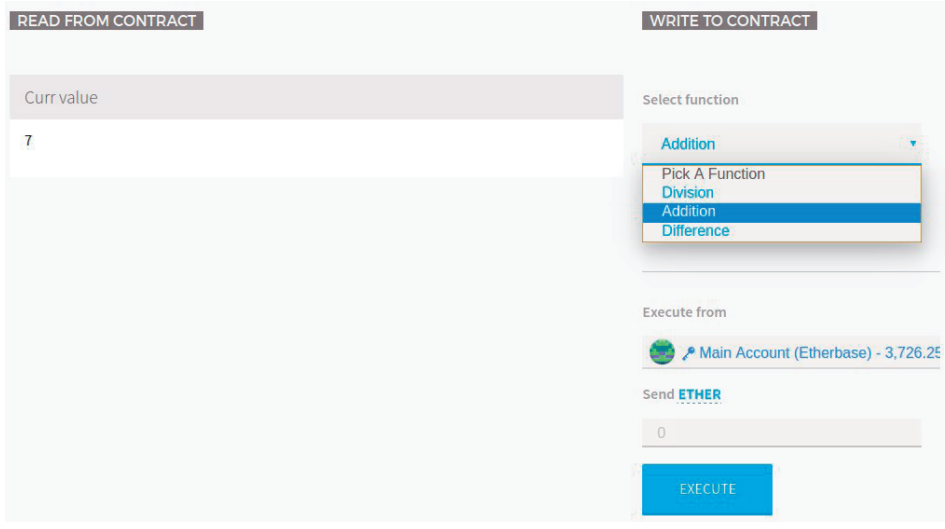
Rysunek 12.13. Dodawanie kontraktu w przeglądarce Mist



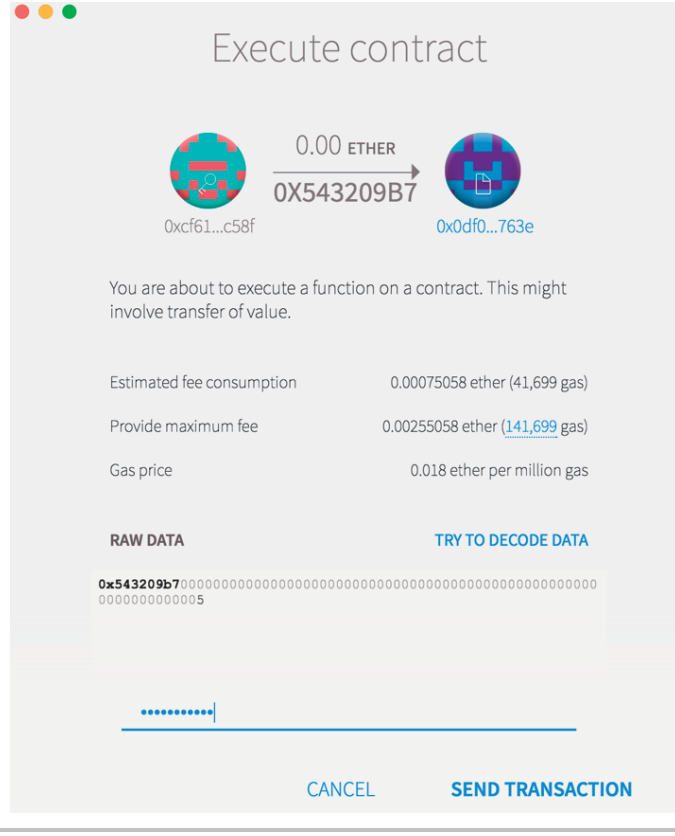
Rysunek 12.14. Tworzenie kontraktu za pomocą przeglądarki Mist



Rysunek 12.15. Lista transakcji po utworzeniu kontraktu w przeglądarce Mist

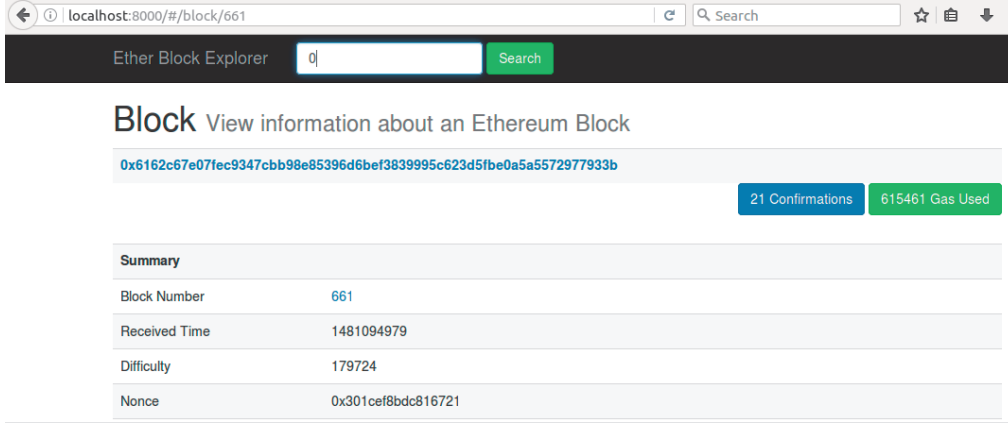


Rysunek 12.16. Interakcje z kontraktem za pomocą opcji odczytu i zapisu w przeglądarce Mist

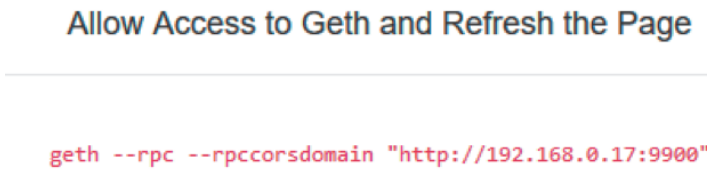


Rysunek 12.17. Wykonywanie kontraktu w przeglądarce Mist

Rysunek 12.18. Serwer HTTP eksploratora Ethereum

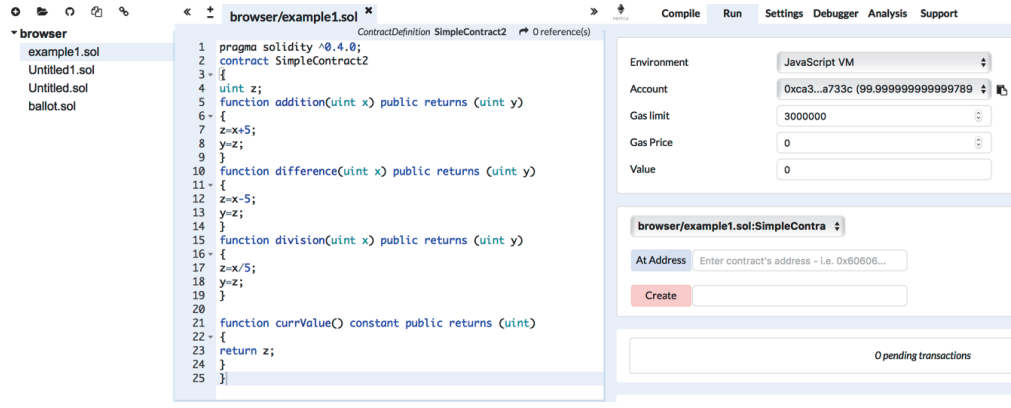


Rysunek 12.19. Eksplorator bloków

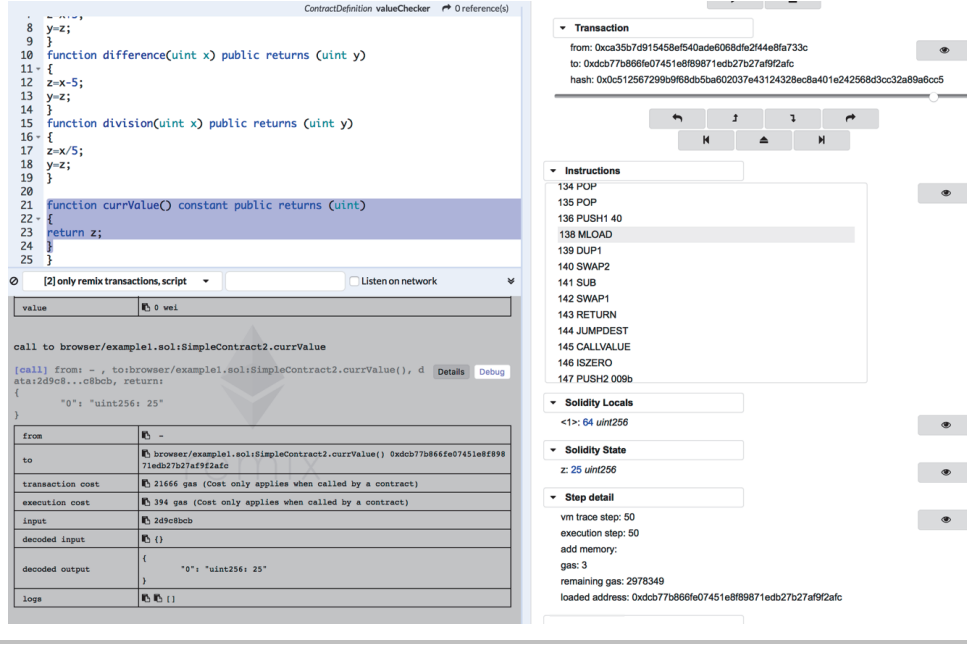


Rysunek 12.20. Komunikat o błędzie

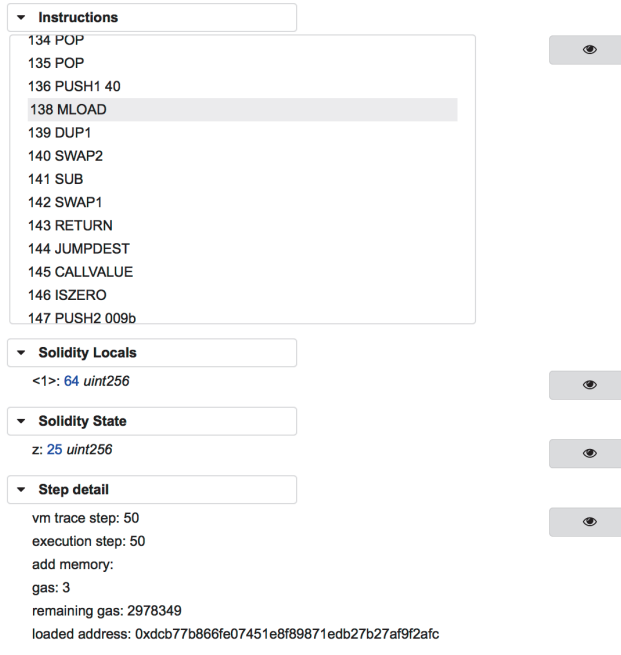
Rozdział 13. Narzędzia i platformy programistyczne



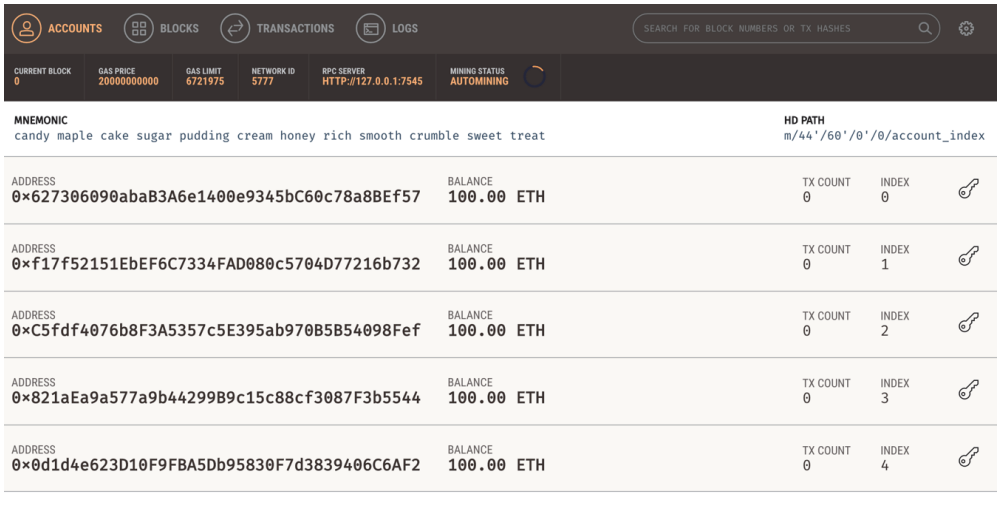
Rysunek 13.5. Środowisko IDE Remix



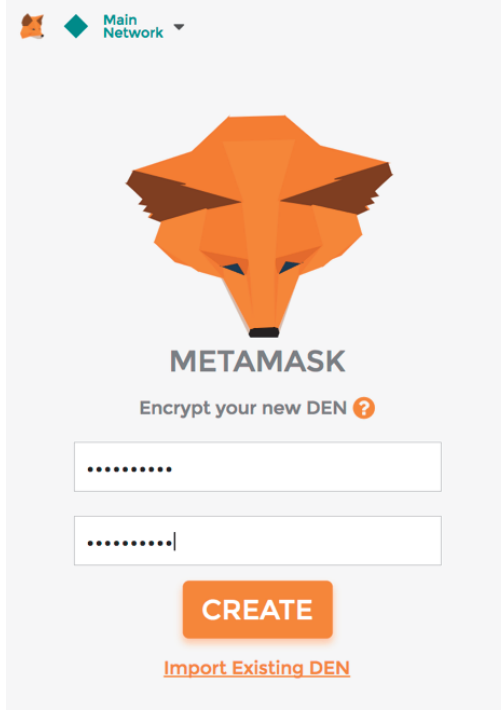
Rysunek 13.6. Debugowanie w środowisku IDE Remix



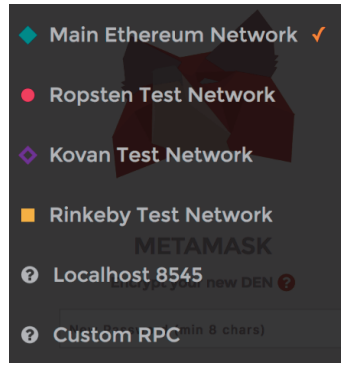
Rysunek 13.7. Debuger w środowisku Remix



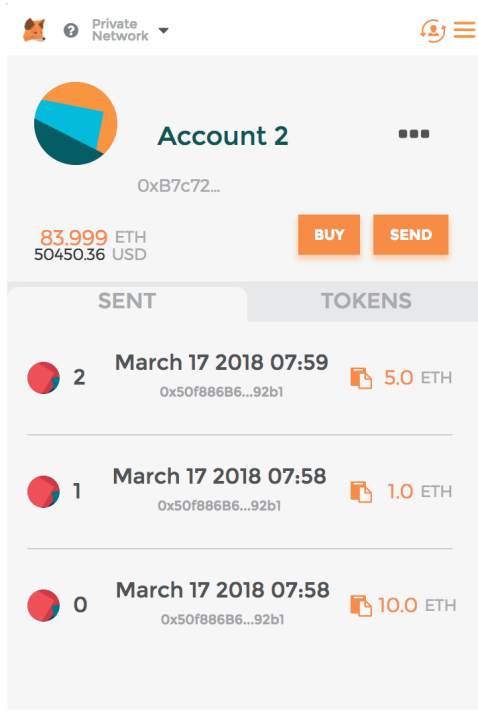
Rysunek 13.9. Ganache — osobisty łańcuch bloków Ethereum



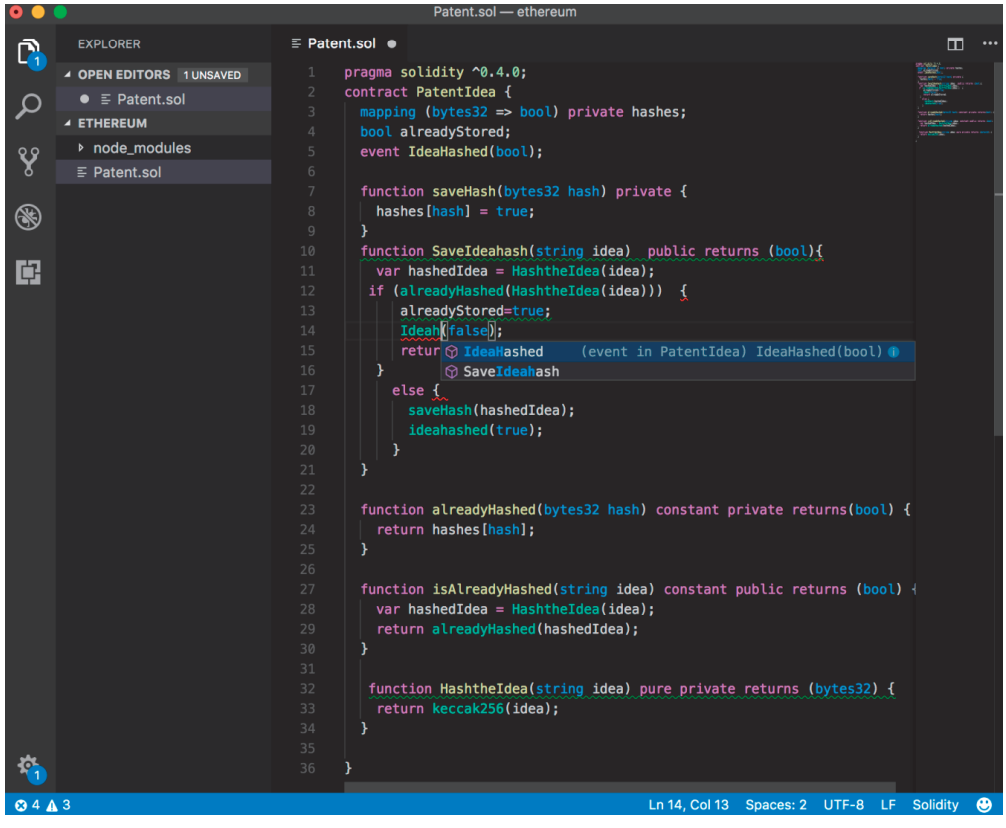
Rysunek 13.10. MetaMask



Rysunek 13.11. Sieci Ethereum widoczne w interfejsie użytkownika narzędzia MetaMask



Rysunek 13.12. Konta i podgląd transakcji w narzędziu MetaMask



Rysunek 13.14. Edytor Visual Studio Code

```
pragma solidity ^0.4.0; // Określa wersję kompilatora.
/*
To prosty kontrakt do sprawdzania wartości. Sprawdza
podaną wartość i zwraca wartość logiczną na podstawie
przetworzenia wyrażenia warunkowego.
*/
import "dev.oraclize.it/api.sol";
contract valuechecker {
   uint price=10;
   //Jest to deklaracja zmiennej price inicjowanej wartością 10.
   event valueEvent(bool returnValue)
   function Matcher (uint8 x) returns (bool)
   {
      if ( x >= price )
      {
         valueEvent(true);
         return true;
      }
   }
}
```

Rysunek 13.16. Przykładowy program w języku Solidity widoczny w środowisku IDE Remix

Rozdział 14. Wprowadzenie do Web3

```
> web3.version
{
  api: "0.15.3",
  ethereum: "0x3f",
  network: "786",
  node: "Geth/v1.5.2-stable-c8695209/linux/go1.7.3",
  whisper: undefined,
  getEthereum: function(callback),
  getNetwork: function(callback),
  getNode: function(callback),
  getWhisper: function(callback)
}
>
```

Rysunek 14.1. Używanie biblioteki Web3 za pomocą klienta geth

```
1  pragma solidity ^0.4.0;
2  contract valueChecker
3  {
4      uint price=10;
5  event valueEvent(bool returnValue);
6  function Matcher (uint8 x) public returns (bool)
7  {
8      if (x>=price){valueEvent(true);
9      return true;
10
11     }}}
```

Start to compile Auto compile

valueChecker Details Publish on Swarm

valueChecker

Rysunek 14.2. Kod w środowisku Remix

WEB3DEPLOY

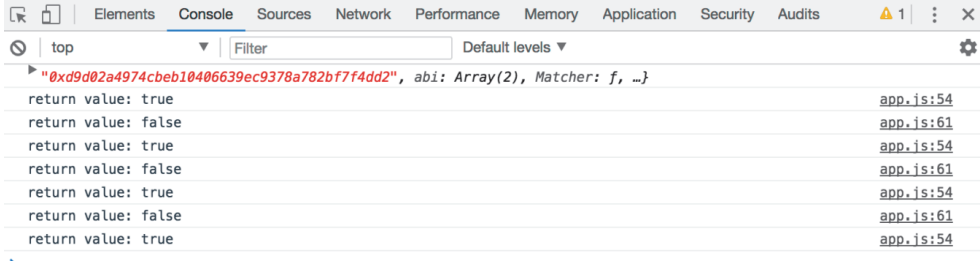
Copy value to clipboard

```
var valuecheckerContract = web3.eth.contract([{"constant":false,"inpu
ts":[{"name":"x","type":"uint8"}],"name":"Matcher","outputs":[{"name"
:"","type":"bool"}],"payable":false,"stateMutability":"nonpayable","t
ype":"function"},{"anonymous":false,"inputs":[{"indexed":false,"name"
:"returnValue","type":"bool"}],"name":"valueEvent","type":"event"}]);
var valuechecker = valuecheckerContract.new(
   {
     from: web3.eth.accounts[0],
     data: '0x6060604052600a600055341561001457600080fd5b6101038061002
36000396000f300606060405260043610603f576000357c010000000000000000000
000000000000000000000000000000000000000900463ffffffff168063f9d55e2114604
4575b600080fd5b3415604e57600080fd5b60656004808035560ff1690602001909190
5050607f565b604051808215151515815260200191505060405180910390f35b60008
0548260ff1610151560d1577f3eb1a229ff7995457774a4bd31ef7b13b6f4491ad1eb
b8961af120b8b4b6239c60016040518082151515158152602001915050604051809010
390a16001905060d2565b5b9190505600a165627a7a723058205b1d9d0f31b39806b7
782fdb9360af93d5b5f66a36f6f4023ee1aa9ca12782b70029',
     gas: '4700000'
   }, function (e, contract){
    console.log(e, contract);
    if (typeof contract.address !== 'undefined') {
         console.log('Contract mined! address: ' + contract.address +
 ' transactionHash: ' + contract.transactionHash);
    }
 })
```

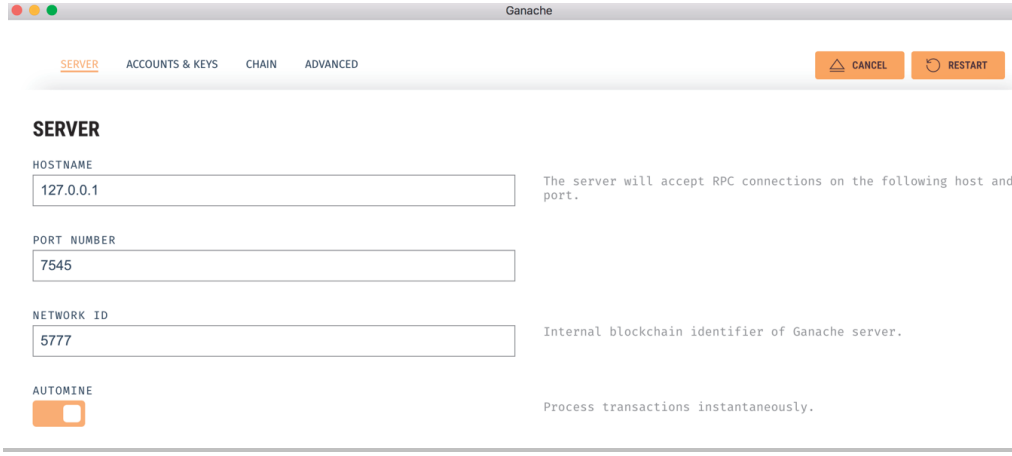
Rysunek 14.3. Skrypt instalacyjny biblioteki Web3

```
> valuechecker.
valuechecker.Matcher          valuechecker.abi            valuechecker.allEvents      valuechecker.transactionHash
valuechecker._eth             valuechecker.address        valuechecker.constructor    valuechecker.valueEvent
> valuechecker.abi
[{
    constant: false,
    inputs: [{
        name: "x",
        type: "uint8"
    }],
    name: "Matcher",
    outputs: [{
        name: "",
        type: "bool"
    }],
    payable: false,
    stateMutability: "nonpayable",
    type: "function"
}, {
    anonymous: false,
    inputs: [{
        indexed: false,
        name: "returnValue",
        type: "bool"
    }],
    name: "valueEvent",
    type: "event"
}]
> valuechecker.address
"0xbd663c5136155cb6d7ed55446888271dcd5092bc"
>
```

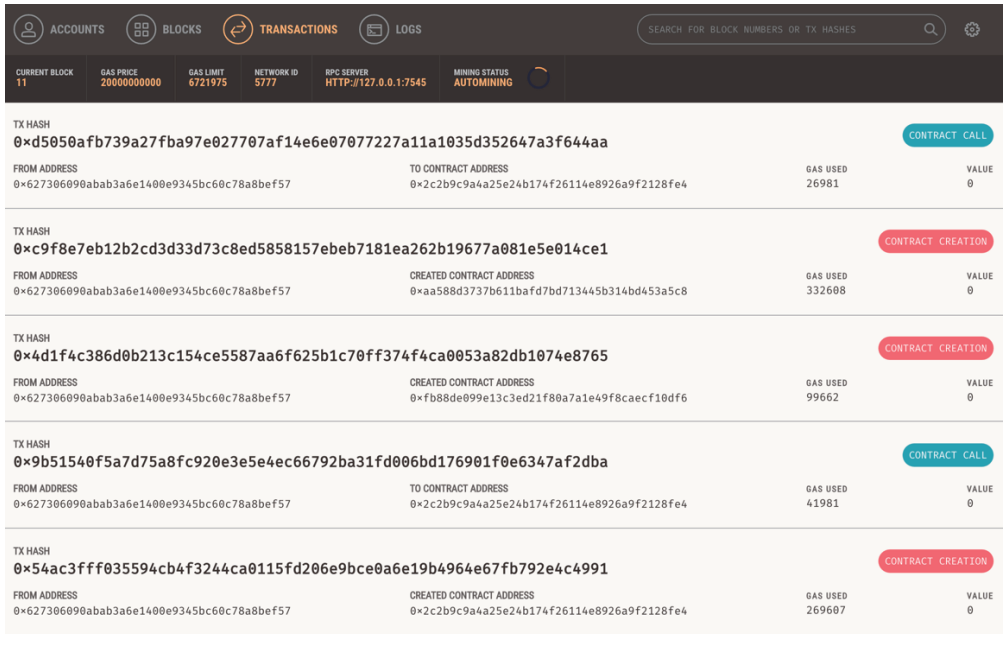
Rysunek 14.5. Atrybuty kontraktu valuechecker



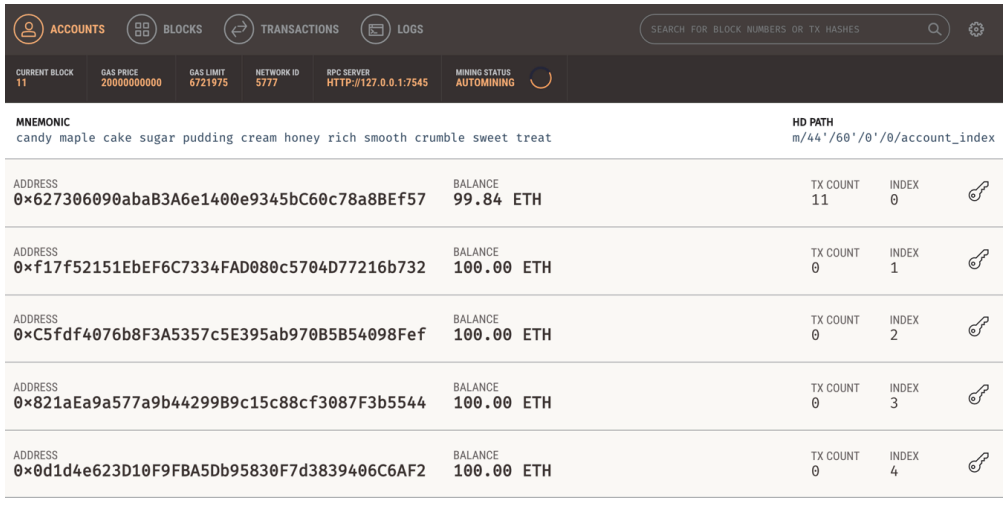
Rysunek 14.7. Interakcja z kontraktem



Rysunek 14.8. Ustawienia narzędzia Ganache



Rysunek 14.10. Transakcje wyświetlone w Ganache

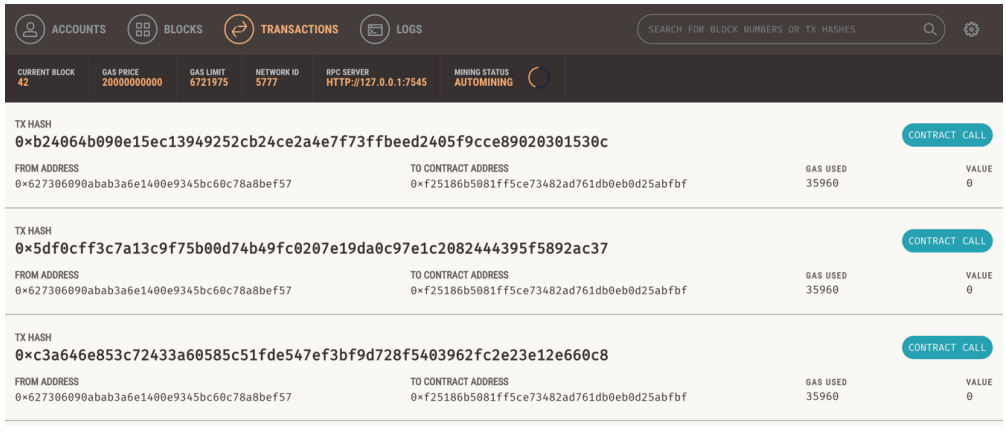


Rysunek 14.11. Wyświetlanie kont w Ganache

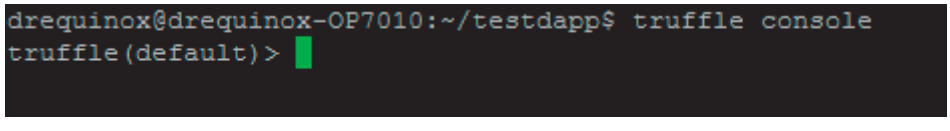
```
> truffle-init-webpack@0.0.2 dev /Users/drequinox/dapp1
> webpack-dev-server

Project is running at http://localhost:8080/
webpack output is served from /
Hash: 157d6514272a12586aba
Version: webpack 2.7.0
Time: 6702ms
     Asset       Size  Chunks                    Chunk Names
    app.js    1.65 MB       0  [emitted]  [big]  main
index.html  925 bytes          [emitted]
chunk    {0} app.js (main) 1.63 MB [entry] [rendered]
   [71] ./app/javascripts/app.js 3.64 kB {0} [built]
   [72] (webpack)-dev-server/client?http://localhost:8080 7.95 kB {0} [built]
   [73] ./build/contracts/MetaCoin.json 23.8 kB {0} [built]
  [111] ./~/loglevel/lib/loglevel.js 7.86 kB {0} [built]
  [117] ./~/querystring-es3/index.js 127 bytes {0} [built]
  [119] ./~/strip-ansi/index.js 161 bytes {0} [built]
  [122] ./app/stylesheets/app.css 905 bytes {0} [built]
  [163] ./~/truffle-contract/index.js 2.64 kB {0} [built]
  [197] ./~/url/url.js 23.3 kB {0} [built]
  [199] ./~/web3/index.js 193 bytes {0} [built]
  [233] (webpack)-dev-server/client/overlay.js 3.73 kB {0} [built]
  [234] (webpack)-dev-server/client/socket.js 1.05 kB {0} [built]
  [235] (webpack)/hot nonrecursive ^\.\/log$ 160 bytes {0} [built]
  [236] (webpack)/hot/emitter.js 77 bytes {0} [built]
  [237] multi (webpack)-dev-server/client?http://localhost:8080 ./app/javascripts/app.js 40 bytes {0} [built]
    + 223 hidden modules
webpack: Compiled successfully.
```

Rysunek 14.12. Uruchamianie pakietu Webpack

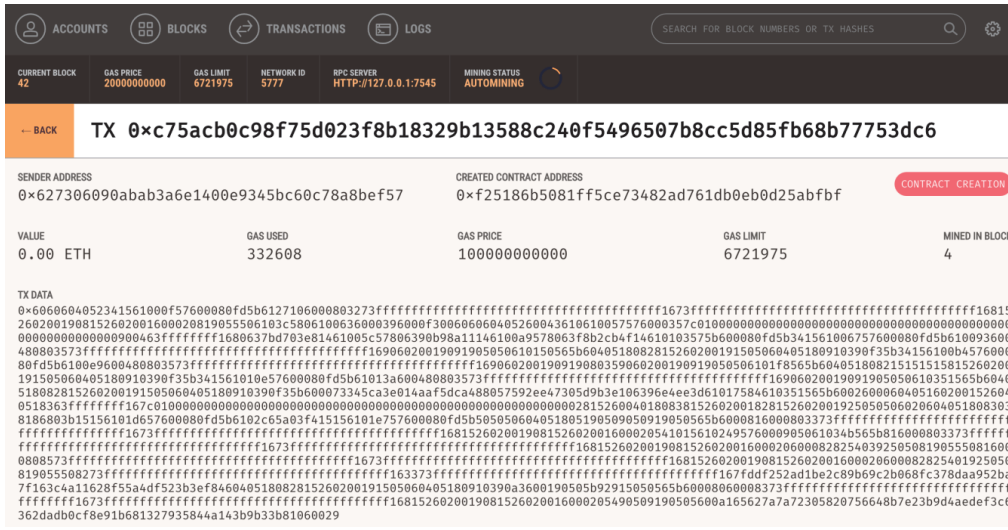
Rysunek 14.13. Fronton przykładowej aplikacji MetaCoin



Rysunek 14.14. Szczegółowe informacje o transakcjach w Ganache



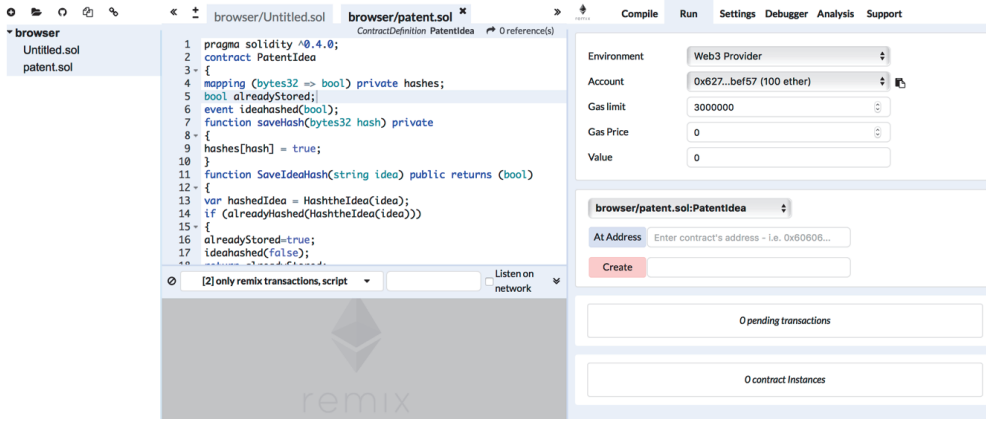
Rysunek 14.15. Konsola platformy Truffle



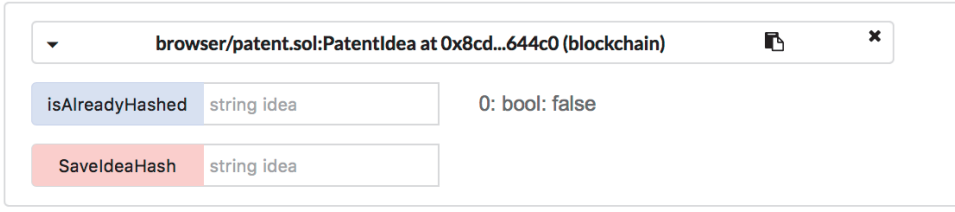
Rysunek 14.17. Transakcja tworząca kontrakt pokazana w Ganache

```
[truffle(development)> Addition.at(Addition.address).addx(100,100)
{ tx: '0xd58f346b76c1d878b67e7318bd4f2f4acb53418d4632efafa35bf939f0181cac',
  receipt:
   { transactionHash: '0xd58f346b76c1d878b67e7318bd4f2f4acb53418d4632efafa35bf939f0181cac',
     transactionIndex: 0,
     blockHash: '0x9215c0d4f7e5f2861b68bc8d097e017c65751dbaf4fe73c1a5729f97ac4315de',
     blockNumber: 41,
     gasUsed: 27131,
     cumulativeGasUsed: 27131,
     contractAddress: null,
     logs: [],
     status: 1 },
  logs: [] }
[truffle(development)> Addition.at(Addition.address).retrievex()
{ [String: '200'] s: 1, e: 2, c: [ 200 ] }
truffle(development)>
```

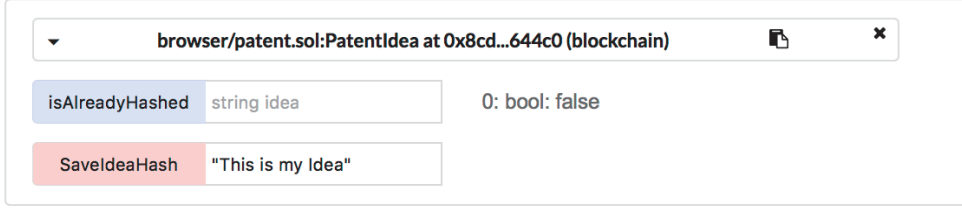
Rysunek 14.18. Konsola platformy Truffle — interakcja z kontraktem



Rysunek 14.19. Tworzenie kontraktu za pomocą przeglądarki dla języka Solidity

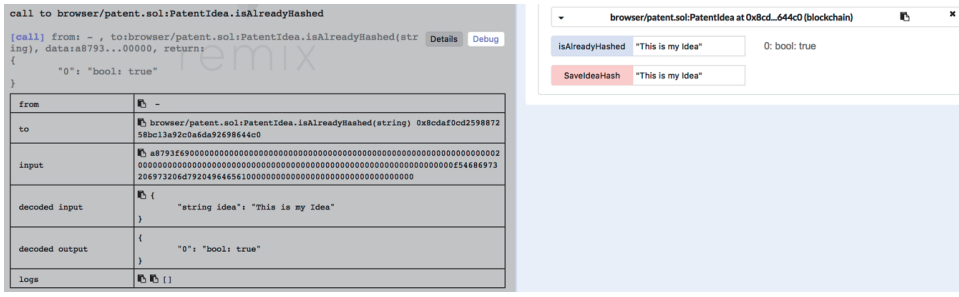


Rysunek 14.20. Udostępnianie dwóch metod

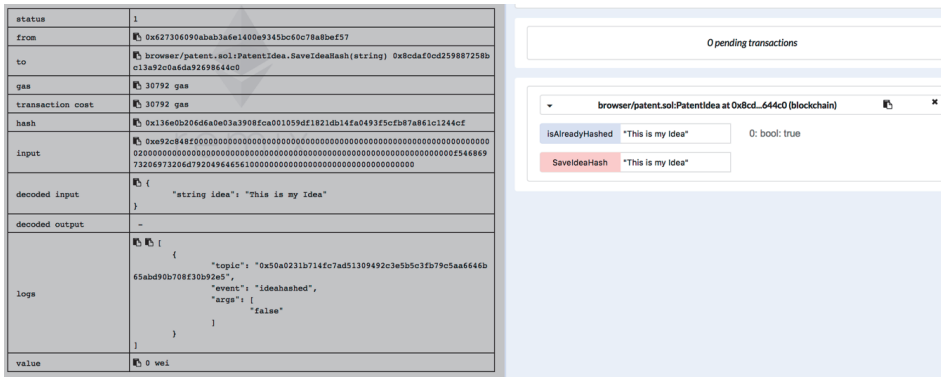


Rysunek 14.21. Wywołanie funkcji SaveIdeaHash

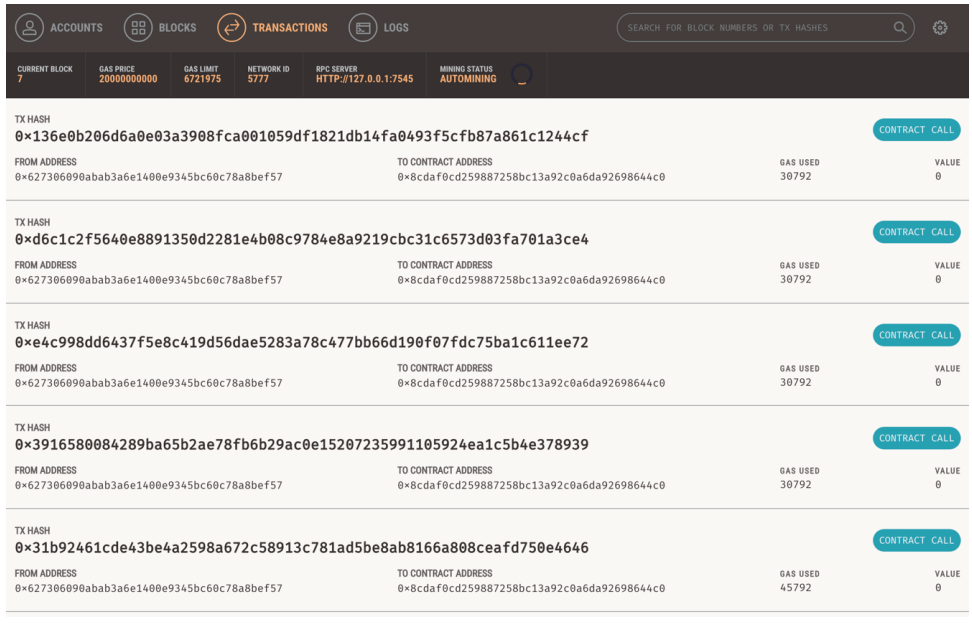
Rysunek 14.22. Dzienniki



Rysunek 14.23. Wykonywanie funkcji isAlreadyHashed



Rysunek 14.24. Wykonywanie funkcji SaveIdeaHash



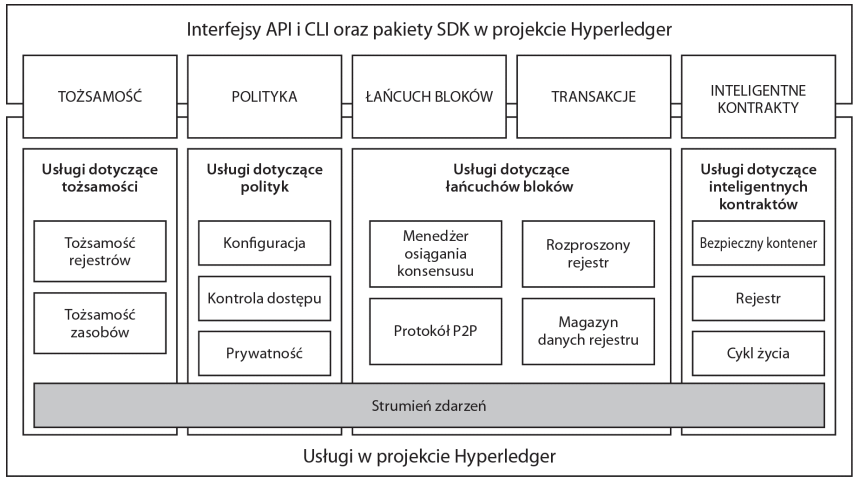
Rysunek 14.25. Narzędzie Ganache wyświetla wszystkie transakcje wywołane za pomocą środowiska IDE Remix

```
[truffle(development)> PatentIdea.deployed().then(function(instance){app = instance})
undefined
[truffle(development)> app.SaveIdeaHash("hello1")
{ tx: '0x597fffed0f859877676b158077c402b09a0ba1cd213c1ba28298e72a1f48af80',
  receipt:
   { blockHash: '0xaf21334ea79cc7f507eca9e6c93f043253fd371d987f0d3593ef3e270e4c77d6',
     blockNumber: 4046,
     contractAddress: null,
     cumulativeGasUsed: 45634,
     from: '0xcf61d213faa9acadbf0d110e1397caf20445c58f',
     gasUsed: 45634,
     logs: [ [Object] ],
     logsBloom: '0x000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000400000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000040000200000000000000000000000000000000000000000000000
     root: '0xc9b84ede2a53eefa6e933d697068230565e16570bffeb3ad21fe71c6ee62454b',
     to: '0x0694d9db791803c83e10175163d8a58df76a2a84',
     transactionHash: '0x597fffed0f859877676b158077c402b09a0ba1cd213c1ba28298e72a1f48af80',
     transactionIndex: 0 },
  logs:
   [ { address: '0x0694d9db791803c83e10175163d8a58df76a2a84',
       blockNumber: 4046,
       transactionHash: '0x597fffed0f859877676b158077c402b09a0ba1cd213c1ba28298e72a1f48af80',
       transactionIndex: 0,
       blockHash: '0xaf21334ea79cc7f507eca9e6c93f043253fd371d987f0d3593ef3e270e4c77d6',
       logIndex: 0,
       removed: false,
       event: 'ideahashed',
       args: [Object] } ] }
```

Rysunek 14.26. Wywoływanie metod dodanego kontraktu w konsoli platformy Truffle

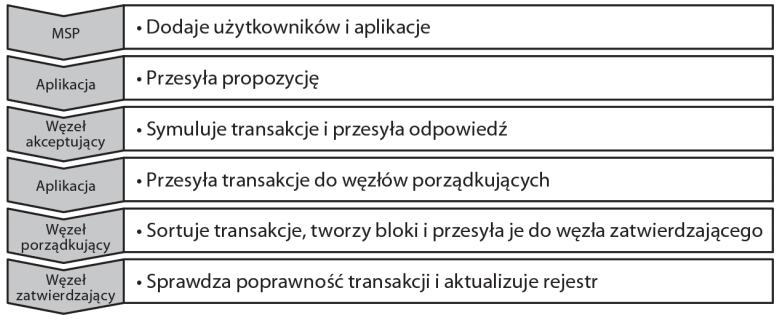
Rysunek 14.28. Instalowanie systemu IPFS

Rozdział 15. Hyperledger

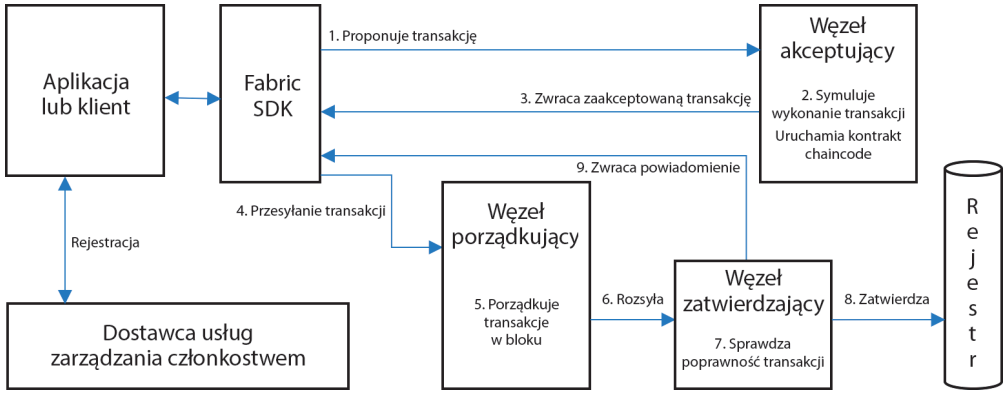


Rysunek 15.1. Architektura wzorcowa. Źródło: dokument na temat projektu Hyperledger

Rysunek 15.6. Proces osiągania konsensusu



Rysunek 15.7. Cykl życia transakcji



Rysunek 15.8. Architektura procesu obsługi transakcji

```
drequinox@drequinox-OP7010:~/project$ git clone https://github.com/IntelLedger/sawtooth-core.git
Cloning into 'sawtooth-core'...
remote: Counting objects: 12527, done.
remote: Compressing objects: 100% (964/964), done.
remote: Total 12527 (delta 452), reused 0 (delta 0), pack-reused 11515
Receiving objects: 100% (12527/12527), 9.26 MiB | 1.76 MiB/s, done.
Resolving deltas: 100% (8131/8131), done.
Checking connectivity... done.
```

Rysunek 15.11. Klonowanie projektu Sawtooth z serwisu GitHub

```
drequinox@drequinox-OP7010:~/project/sawtooth-core/tools$ vagrant up
Could not determine vagrant user.
VAGRANT_BOX = ubuntu/xenial64
VAGRANT_FORWARD_PORTS = true
VAGRANT_MEMORY = 2048
VAGRANT_CPUS = 2
Proxyconf plugin not found
Install: vagrant plugin install vagrant-proxyconf
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'ubuntu/xenial64' could not be found. Attempting to find and install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'ubuntu/xenial64'
    default: URL: https://atlas.hashicorp.com/ubuntu/xenial64
==> default: Adding box 'ubuntu/xenial64' (v20161221.0.0) for provider: virtualbox
    default: Downloading: https://atlas.hashicorp.com/ubuntu/boxes/xenial64/versions/20161221.0.0/providers/virtualbox.bo
x
    default: Progress: 1% (Rate: 1709k/s, Estimated time remaining: 0:04:04)
```

Rysunek 15.12. Wykonywanie polecenia vagrant up

```
ubuntu@ubuntu-xenial:/project/sawtooth-core$ /project/sawtooth-core/docs/source/tutorial/genesis.sh
writing file: /home/ubuntu/sawtooth/keys/base000.wif
writing file: /home/ubuntu/sawtooth/keys/base000.addr
```

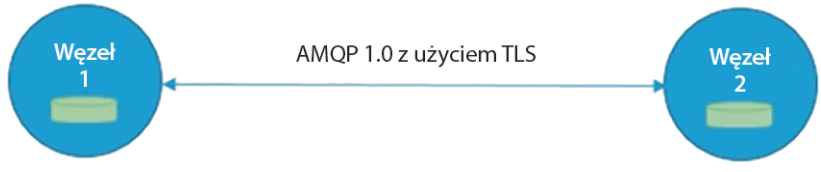
Rysunek 15.13. Generowanie bloku początkowego i kluczy

```
ubuntu@ubuntu-xenial:/project/sawtooth-core$ ./bin/txnvalidator -v -F ledger.transaction.integer_key --config /home/ubuntu/sawto
oth/v0.json
[22:08:22 INFO    validator_cli] validator started with arguments: ['./bin/txnvalidator', '-v', '-F', 'ledger.transaction.intege
r_key', '--config', '/home/ubuntu/sawtooth/v0.json']
[22:08:22 INFO    validator_cli] read signing key from /home/ubuntu/sawtooth/keys/base000.wif
[22:08:24 WARNING validator_cli] validator pid is 10937
[22:08:24 INFO    gossip_core] listening on IPv4Address(UDP, '0.0.0.0', 33713)
[22:08:24 INFO    global_store_manager] create blockstore from file /home/ubuntu/sawtooth/data/base000_state.dbm with flag c
[22:08:24 INFO    validator] set administration node to None
[22:08:24 INFO    validator] starting ledger base000 with id 1KSRNedZ at network address ('127.0.0.1', 33713)
[22:08:24 INFO    web_api] listen for HTTP requests on (ip='localhost', port=8800)
[22:08:24 INFO    validator_cli] adding transaction family: ledger.transaction.integer_key
[22:08:24 INFO    journal_core] restore ledger state from persistence
[22:08:24 INFO    global_store_manager] add block 60af3ec894fa1cb0 to the queue for loading
[22:08:24 INFO    global_store_manager] load block 60af3ec894fa1cb0 from storage
[22:08:24 INFO    journal_core] commit head: 60af3ec894fa1cb0
[22:08:26 INFO    validator] ledger connections using RandomWalk topology
[22:08:26 INFO    random_walk] initiate random walk topology update
[22:08:29 INFO    validator] ledger initialization complete
[22:08:29 INFO    journal_core] process initial transactions and blocks
[22:08:29 INFO    validator] register endpoint 1KSRNedZ with name base000
[22:08:29 INFO    journal_core] build transaction block to extend 60af3ec8 with 1 transactions
[22:08:29 INFO    wait_timer] wait timer created; TIMER, 5.00, 33.69, HE2DQNJWGI2DCNJQ
```

Rysunek 15.14. Uruchamianie mechanizmu sprawdzania poprawności transakcji

```
ubuntu@ubuntu-xenial:/project/sawtooth-core$ ./bin/mktclient --name market --keyfile validator/keys/mkt.wif
//UNKNOWN> help

Documented commands (type help <topic>):
========================================
EOF         dump           exit      liability    selloffer  tokenstore
account     echo           help      map          session    waitforcommit
asset       exchange       holding   offers       sleep
assettype   exchangeoffer  holdings  participant  state

Miscellaneous help topics:
==========================
symbols  names

//UNKNOWN> participant reg --name market --description "the market"
transaction ff652e63dadeaf32 submitted
//market>
```

Rysunek 15.15. Klient market utworzony na potrzeby rodziny transakcji MarketPlace



Rysunek 15.17. Dwa węzły komunikujące się w sieci Corda

Rozdział 16. Inne łańcuchy bloków

```
drequinox@drequinox-OP7010:~/Downloads$ ./pact
pact> 1234
1234
pact> (+ 1 2)
3
pact> (if (= (+ 1 2) 3 "OK" "ERROR")
(interactive):1:31: error: unexpected
    EOF, expected: ")", ";", "{",
    Boolean false, Boolean true,
    Decimal literal, Integer literal,
    String literal, Symbol literal,
    list literal, pact, sexp, space
(if (= (+ 1 2) 3 "OK" "ERROR")<EOF>
                              ^
pact> (if (= (+ 1 2) 3) "OK" "ERROR")
"OK"
pact>
```

Rysunek 16.2. Środowisko REPL języka Pact; widoczne są tu przykładowe polecenia i komunikat o błędzie

```
1  ;Rozpoczynanie transakcji o opcjonalnej nazwie.
2  (begin-tx) 'testTransaction
3  ;Podawanie danych transakcji w formacie JSON z typami języka Pact.
4  (env-data { "keyset": {"keys": ["admin"], "pred": "keys-any"}})
5  ;Definiowanie keyset jako nazwy powiązanej ze zbiorem kluczy.
6  (define-keyset 'admin-keyset (read-keyset "keyset"))
7  ;Podawanie kluczy na potrzeby podpisywania transakcji.
8  (env-keys ["admin"])
9  ;Definiowanie modułu za pomocą składni (module NAZWA KLUCZE [DOKUMENT] DEFINICJE ...
10 (module additionModule 'admin-keyset
11 ;Definiowanie funkcji przyjmującej trzy argumenty: x y z.
12 (defun addition (x y z) (+ x (+ y z))))
13 ;Zatwierdzanie transakcji.
14 (commit-tx)
15 ;Używanie funkcji addition.
16 (use 'additionModule)
17 ;Uruchomienie funkcji addition i sformatowanie wyniku.
18 (format "Wynik: {}" [(addition 100 200 300)])
```

Rysunek 16.3. Przykładowy kod w języku Pact

```
""
""
"testTransaction"
"Setting transaction data"
"Keyset defined"
"Setting transaction keys"
"Loaded module \"additionModule\", hash
\"08a0de7383896b88e7287ed869bb0f405c8d7e7371404916c41330addbf3bab0eccec3fa92bd3f
742e25beb60cd9022fef411a039c5fd1107266e69f4eccb2c7\""
""
""
"Using \"additionModule\""
"Wynik: 600"
```

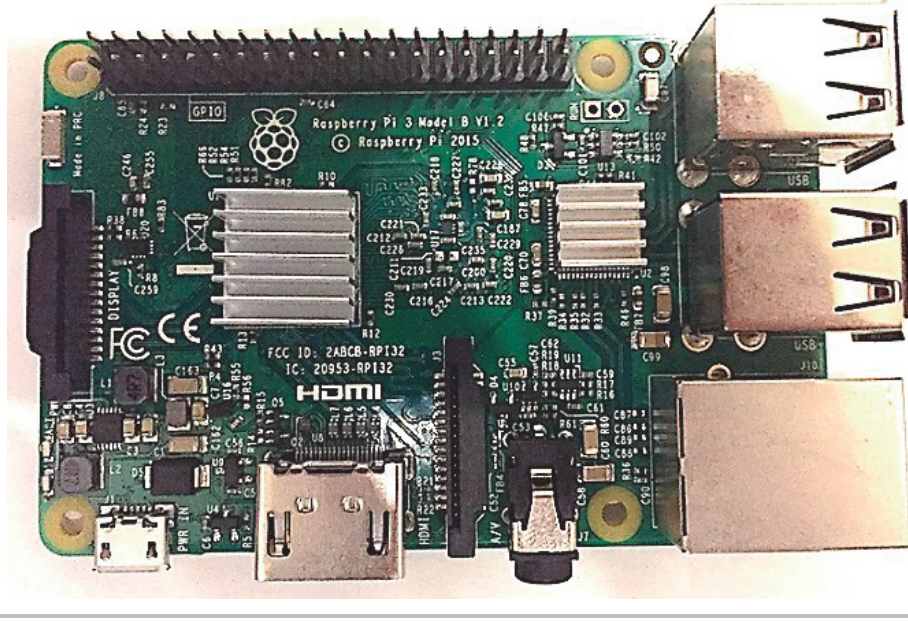
Rysunek 16.4. Dane wyjściowe kodu

Rysunek 16.5. Etapy protokołu osiągania konsensusu w systemie Ripple

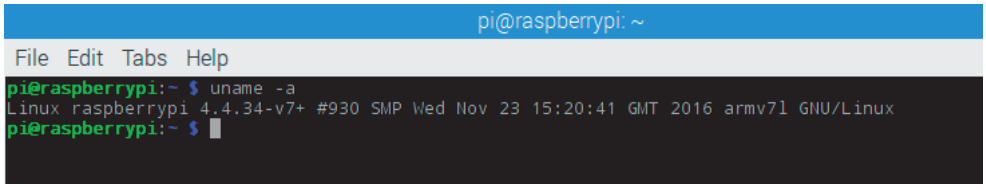
Rozdział 17. Łańcuch bloków — poza świat walut

Rysunek 17.1. Typowa sieć internetu rzeczy (źródło: IBM)

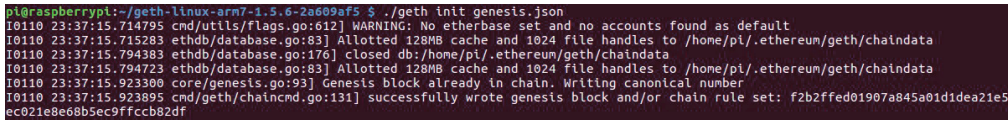
Rysunek 17.4. Model bezpośredniej komunikacji opartej na łańcuchu bloków (źródło: IBM)



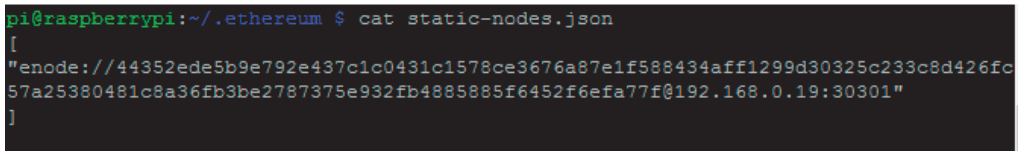
Rysunek 17.5. Raspberry Pi Model B



Rysunek 17.6. Architektura Raspberry Pi

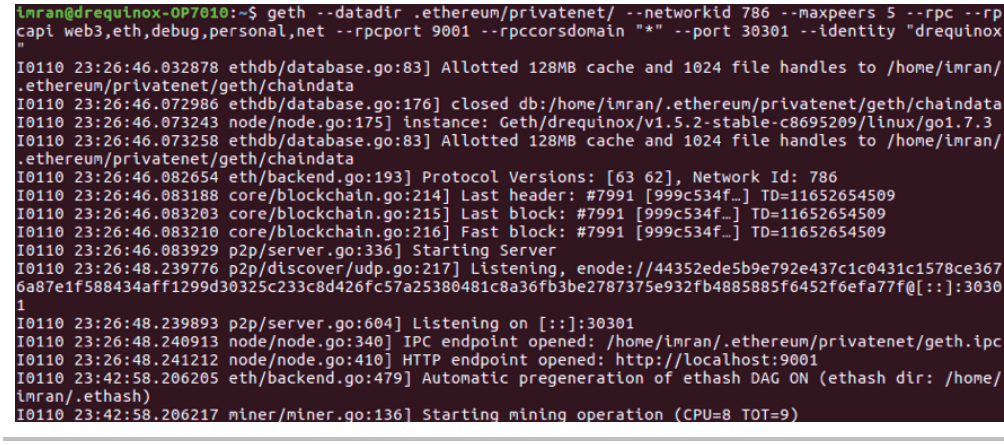


Rysunek 17.7. Inicjowanie bloku początkowego



Rysunek 17.8. Konfigurowanie węzłów statycznych

Rysunek 17.9. Instrukcja nodeInfo w kliencie Geth



Rysunek 17.10. Uruchamianie klienta Geth w pierwszym węźle

```
pi@raspberrypi:~/geth-linux-arm7-1.5.6-2a609af5 $ ./geth  --networkid 786 --maxpeers 5 --rpc --rpcapi web3,eth,debug,personal,net --
 --rpccorsdomain "*" --port 30302 --identity "raspberry"
I0110 23:38:04.654374 cmd/utils/flags.go:612] WARNING: No etherbase set and no accounts found as default
I0110 23:38:04.654776 ethdb/database.go:83] Allotted 128MB cache and 1024 file handles to /home/pi/.ethereum/geth/chaindata
I0110 23:38:04.693111 ethdb/database.go:176] closed db:/home/pi/.ethereum/geth/chaindata
I0110 23:38:04.696937 node/node.go:176] instance: Geth/raspberry/v1.5.6-stable-2a609af5/linux/go1.7.4
I0110 23:38:04.697042 ethdb/database.go:83] Allotted 128MB cache and 1024 file handles to /home/pi/.ethereum/geth/chaindata
I0110 23:38:04.847835 eth/backend.go:191] Protocol Versions: [63 62], Network Id: 786
I0110 23:38:04.849753 eth/backend.go:219] Chain config: {ChainID: 0 Homestead: <nil> DAO: <nil> DAOSupport: false EIP150: <nil> EIP1
P158: <nil>}
I0110 23:38:04.857847 core/blockchain.go:216] Last header: #2668 [6776ef24…] TD=708187563
I0110 23:38:04.858174 core/blockchain.go:217] Last block: #2668 [6776ef24…] TD=708187563
I0110 23:38:04.858349 core/blockchain.go:218] Fast block: #2668 [6776ef24…] TD=708187563
I0110 23:38:04.866705 p2p/server.go:340] Starting Server
I0110 23:38:10.223170 p2p/discover/udp.go:227] Listening, enode://98ba36ecea7ff011803d634da45752abd25101f20a62f23427afc3f280017bc134
b195ac6ed59c3b01ca2a3f14638a52697a1bb1bf967fc84274@86.15.44.209:30302
I0110 23:38:10.224031 p2p/server.go:608] Listening on [::]:30302
I0110 23:38:10.233788 node/node.go:341] IPC endpoint opened: /home/pi/.ethereum/geth.ipc
I0110 23:38:10.237027 node/node.go:411] HTTP endpoint opened: http://localhost:9002
I0110 23:38:20.225637 eth/downloader/downloader.go:326] Block synchronisation started
I0110 23:38:49.583631 core/blockchain.go:1067] imported 1 blocks,     0 txs (  0.000 Mg) in   14.018s ( 0.000 Mg/s). #2669 [76077955
I0110 23:38:49.622191 core/blockchain.go:1067] imported 5 blocks,     0 txs (  0.000 Mg) in  38.520ms ( 0.000 Mg/s). #2674 [76077955
```

Rysunek 17.11. Klient Geth w Raspberry Pi

```
> admin.peers
[{
    caps: ["eth/62", "eth/63"],
    id: "44352ede5b9e792e437c1c0431c1578ce3676a87e1f588434aff1299d30325c233c8d426fc57a25380481c8a36fb3be2787375e932f
b4885885f6452f6efa77f",
    name: "Geth/drequinox/v1.5.2-stable-c8695209/linux/go1.7.3",
    network: {
      localAddress: "192.168.0.21:56550",
      remoteAddress: "192.168.0.19:30301"
    },
    protocols: {
      eth: {
        difficulty: 11719415397,
        head: "0x2d32c90b4c9dacea9a109b0ae52c1ebf511915bb618a2d3c55a80a63852e89f6",
        version: 63
      }
    }
}]
```

Rysunek 17.12. Polecenie admin.peers w konsoli klienta geth w Raspberry Pi

```
> admin.peers
[{
    caps: ["eth/62", "eth/63"],
    id: "98ba36ecea7ff011803d634da45752abd25101f20a62f23427afc3f280017bc134833dd5ba400bb195ac6ed59c3b01
ca2a3f14638a52697a1bb1bf967fc84274",
    name: "Geth/raspberry/v1.5.6-stable-2a609af5/linux/go1.7.4",
    network: {
      localAddress: "192.168.0.19:30301",
      remoteAddress: "192.168.0.21:56512"
    },
    protocols: {
      eth: {
        difficulty: 11700366137,
        head: "0x1188f58b4900a1d771d333141ea9400d78400bb8e561494ab436519ae64e1e34",
        version: 63
      }
    }
}]
```

Rysunek 17.13. Polecenie admin.peers w konsoli klienta geth uruchomione w innym węźle

```
pi@raspberrypi:~/testled $ curl -sL https://deb.nodesource.com/setup_7.x | sudo -E bash -

## Installing the NodeSource Node.js v7.x repo...


## Populating apt-get cache...

+ apt-get update
Get:1 http://archive.raspberrypi.org jessie InRelease [22.9 kB]
```

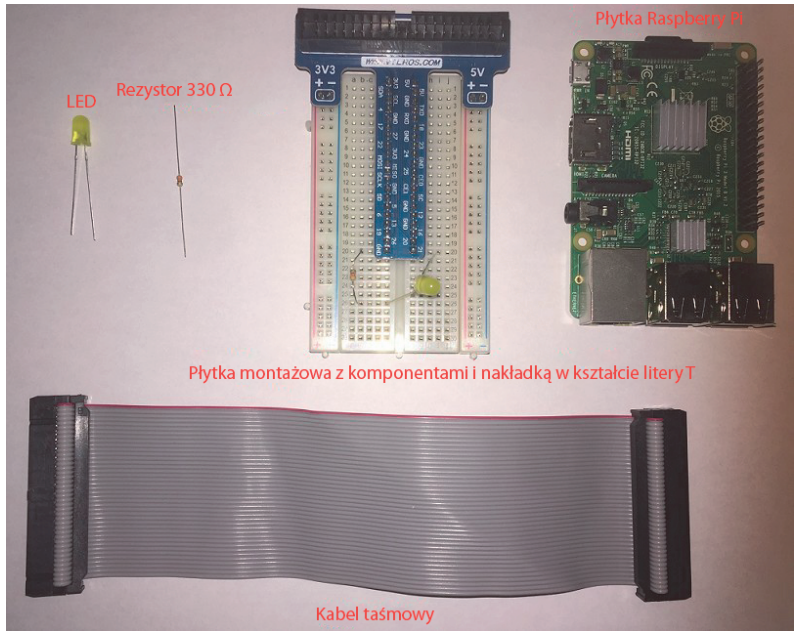
Rysunek 17.14. Instalowanie środowiska Node.js

```
pi@raspberrypi:~/testled $ npm -v
4.0.5
pi@raspberrypi:~/testled $ node -v
v7.4.0
pi@raspberrypi:~/testled $
```

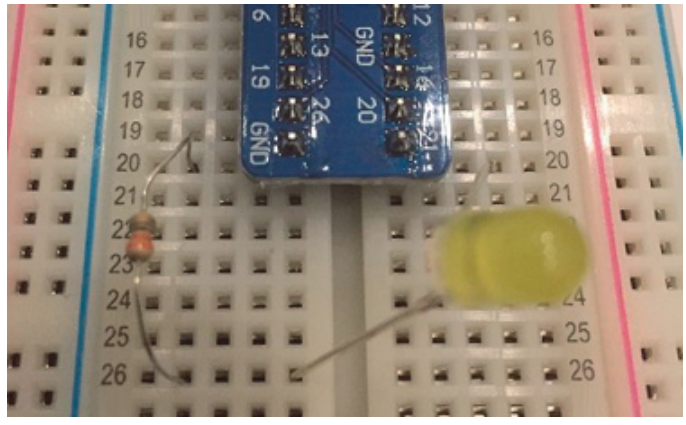
Rysunek 17.15. Sprawdzanie zainstalowanych wersji narzędzi Node.js i npm

```
pi@raspberrypi:~/testled $ npm install web3
testled@1.0.0 /home/pi/testled
└─┬ web3@0.18.0
  └── bignumber.js@2.0.7  (git+https://github.com/debris/bignumber.js.git#94d7146671b9719e00a09c29b01a691bc85048c2)

npm WARN testled@1.0.0 No repository field.
pi@raspberrypi:~/testled $
```

Rysunek 17.16. Instalowanie pakietu web3 za pomocą narzędzia npm



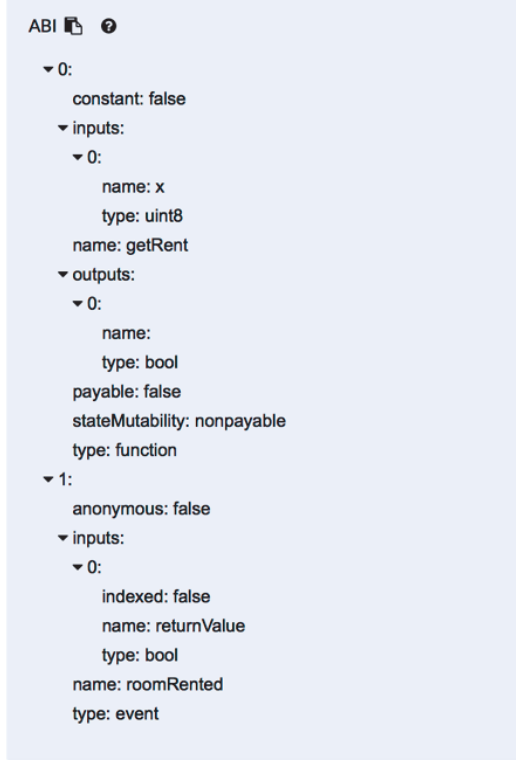
Rysunek 17.18. Potrzebne komponenty



Rysunek 17.19. Podłączenia komponentów na płytce montażowej

```
pragma solidity ^0.4.0;
contract simpleIOT {
    uint roomrent = 10;
    event roomRented(bool returnValue);
    function getRent (uint8 x) public returns (bool) {
        if (x==roomrent) {
            roomRented(true);
            return true;
        }
    }
}
```

Rysunek 17.20. Kod w języku Solidity dla prostego urządzenia dla internetu rzeczy



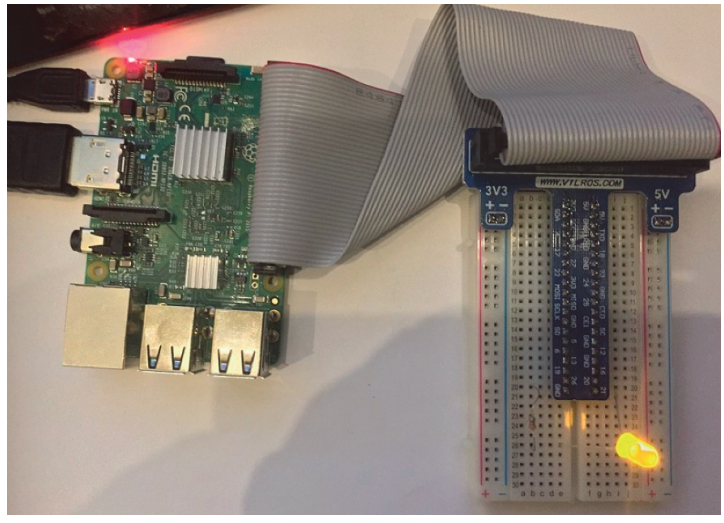
Rysunek 17.21. Interfejs ABI w środowisku IDE Remix

```
imran@drequinox-OP7010:~/iotcontract$ truffle migrate --reset
Running migration: 1_initial_migration.js
  Deploying Migrations...
  Migrations: 0xdd8a88072aa4ff49b62c25d6f6f2207b731aee76
Saving successful migration to network...
Saving artifacts...
Running migration: 2_deploy_contracts.js
  Deploying simpleIOT...
  simpleIOT: 0x151ce17c28b20ce554e0d944deb30e0447fbf78d
Saving successful migration to network...
Saving artifacts...
```

Rysunek 17.24. Dodawanie kontraktu za pomocą narzędzia Truffle

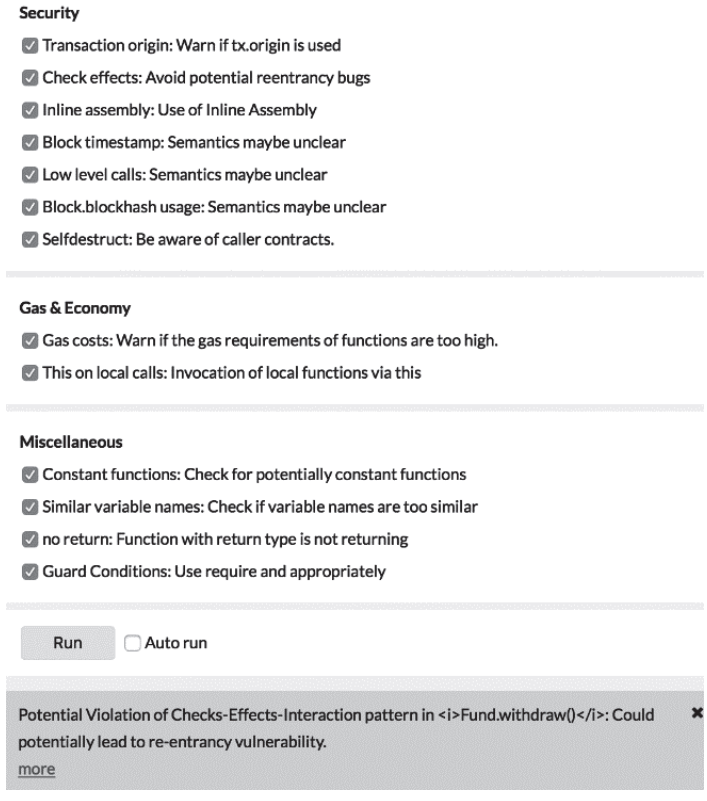
```
[truffle(development)> simpleiot.getRent(10)
 '0x71f550949a4c5168af7b9f7f84fada99bccc20a123779642e5e8c0c0127266ee'
```

Rysunek 17.25. Interakcja z kontraktem

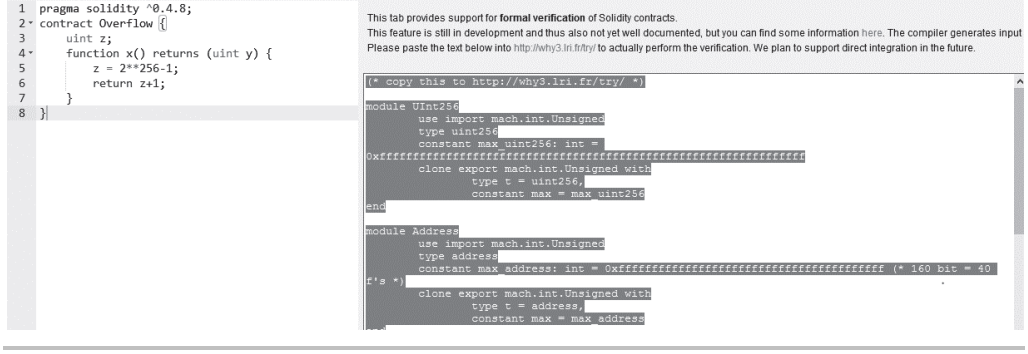


Rysunek 17.26. Raspberry Pi sterujące diodą LED

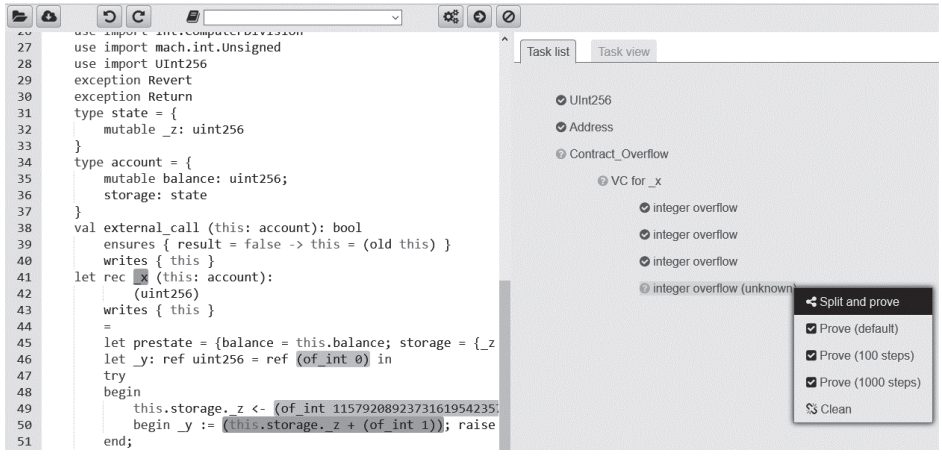
Rozdział 18. Skalowalność i inne problemy



Rysunek 18.2. Opcje analiz w środowisku IDE Remix



Rysunek 18.3. Internetowy kompilator kodu w języku Solidity oferujący formalne sprawdzanie poprawności

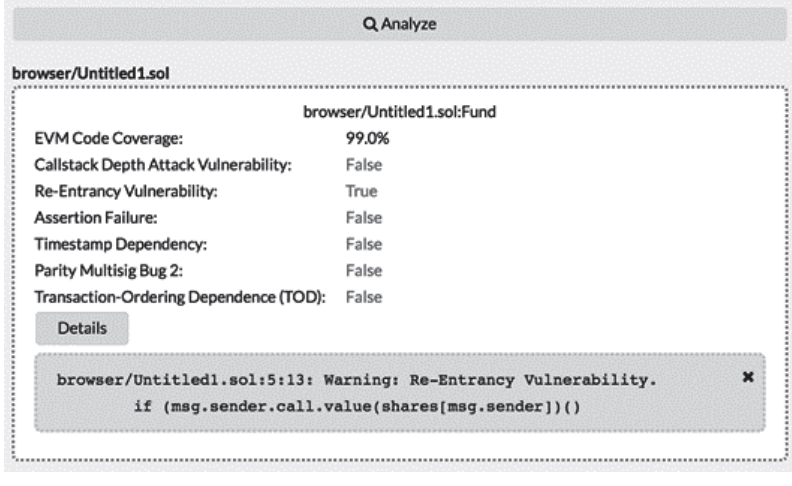


Rysunek 18.4. Why3

```
pragma solidity ^0.4.0;
contract Fund {
    mapping(address => uint) shares;
    function withdraw() public {
        if (msg.sender.call.value(shares[msg.sender])())
            shares[msg.sender] = 0;
    }
}
```

Rysunek 18.5. Kontrakt z błędem związanym z ponownym wywoływaniem kodu. Źródło: dokumentacja języka Solidity

Rysunek 18.6. Narzędzie Oyente wykrywające błędy w języku Solidity



Rysunek 18.7. Analizy w narzędziu Oyente