

# World Compensation System (WCS)

**Keywords:** #2020, #Blockchain, #InternetOfValue, #RSK, #DeFi, #DeFi-App, #DFApp, #wcsDFApp #Unbanked

**Classification:** DeFi : EcoSystem : @rif\_os : **Version:** 0.0.1 (2020-01-15)

**Status:** *in work*

## Purpose

Simple. Eliminate money invisibility.

## User-story

I dare you to throw out all your money, all your papers and coins and individual national currencies, and start over.

Develop an international monetary system that is wide open, totally visible, immediately traceable, completely accountable. Establish a Worldwide Compensation System by which people would be given Credits for services rendered and products produced, and Debits for services used and products consumed.

Under the new Worldwide Compensation System, WCS, the transfer of Debits and Credits would be immediate and totally visible. That is, anybody and everybody could inspect the account of any other person or organization at any time. Nothing would be kept secret, nothing would be ‘private’

Everything would be on the system of Credits and Debits. Returns on investments, inheritances, winnings of wagers, salaries and wages, tips and gratuities, everything. The WCS would deduct 10 percent of all earnings each year from the income of those *voluntarily requesting* such a deduction. Everyone in the society would be able to observe who was choosing to offer the 10 percent for the general good of all, and who was not. And everyone’s records would be open to everyone else. And nothing could be purchased without Credits. There would be no other negotiable currency. ([source:#CWG, @realNealWealsh](http://ISBN))

**Short-name:** WCS, DeFiApp, wcsDFApp

**Disclaimer:** All quoted phrases are verbatim copies found at Conversations with God. Book Two (see Bibliography)

**Bibliography:** - Conversations with God (c) 1997 Neale Donald Walsch. ISBN 978 0 340 76544 9

# World Compensation Operating (Eco-)System

## Use-cases

- Value creation (energy-flow)
- Development of services
- Production or products
- Labour/Work

## Extended Use-cases

- Energy eXchange

## Decentralised Finances (DeFi)

- Taxation
- Borrows/Loans

## Community

- [[@WorldCompensationSystem \(Twitter\)](https://twitter.com/WorldCompensationSystem)](https://twitter.com/WorldCompensationSystem)

## License

Code is under the The Unlicensed. Documentation is under the Creative Commons Attribution license.

## Contributing

Please read our Project Contribution Guide and [Code of Conduct]

## Donations

BTC-Address: | X1v34..7xKt |  
ETH-Address: | 0x123..0000 |

# Whitepaper

## Table-of-contents

### Operating (Eco-)System Concept

1. wcsO(E)S - WCS Operating Eco-System
  1. Platform
  2. Use-cases
    1. Notation
    2. Welcome home
    3. Help
      1. View description manual
    4. User management
    5. File-System (Minimum commands)
      1. Present working directory (pwd)
      2. Listing files (ls)
    6. Applications, Services, Commands and Tools
      1. Running local tool system-service
      2. Running local command system-service
      3. Running local user-application
      4. Running remote user-application
      5. Commands
      6. Tools
    7. Local Services
      1. Financial
      2. Asset Management
    8. Value Operations
      1. Transactions
    9. Distributed Services
    10. Group citizenship
    11. Work – Get Idle Task (according to current citizenship)
    12. Donate
    13. Contribute to Nation's taxes
    14. Value Creation (out-of-thin-air)
    15. Credits
    16. Value Transfer - Request for service
    17. Value Transfer - Investment
      1. Communities
    18. Running DApplication in debug-mode
    19. User-management
  3. User-Application development
    1. DApplication development help
  4. User management
    1. Returning home

- 5. Financial services
  - 1. Lend User-service
  - 2. Claim lend User-service
- 2. Application Notes
  - 1. Borrow service
  - 2. Service production and consumption
  - 3. Wallet
  - 4. Snapshot
  - 5. Application management

## (Eco-)System Application notes

```

homeland$
homeland$help
homeland$man ver
homeland$ver
homeland$home
homeland$user create user1
homeland$pwd
homeland$ls
homeland$stat
homeland$cmd1 --verbose
homeland$app1 --verbose
homeland$dapp1 --verbose
homeland$commands
homeland$tools
homeland$app
homeland$wallet
homeland$credits
homeland$debts
homeland$assets
homeland$send 2 user1
homeland$credit tetris
homeland$neighborhood
homeland$discover
homeland$connect
homeland$ls
homeland$ls nations
homeland$citizen
homeland$citizen federation1
homeland$idle federation1
homeland$donate -idle neighborhood
homeland$tax 8
homeland$offer -idle federation1
homeland$offer 8

```

```

homeland$offer 6
homeland$credits
homeland$tetris
homeland$credits
homeland$invest kernel.org 2
homeland$assets --all
homeland$ls communities
homeland$greetings community1 me --verbose
homeland$offers
homeland$accept 1
homeland$credits
homeland$value dapp1
homeland$stat dapp1
homeland$dapp1 --verbose --debug
homeland$login
homeland$
homeland$ls
homeland$rate dapp1 ***
homeland$SMS user1
homeland$share dapp1 user1 2
homeland$lend dapp1
homeland$claim dapp1
homeland$read ebook1 1
homeland$read ebook1 1
homeland$man wallet
homeland$wallet create
homeland$man snap
homeland$ls -al /dapps
homeland$lend 3 user1 2020-06-31 0.001
homeland$gift 3 user3 /user3/birthday

```

## Applications

/apps:

```

/apps/App1 -- Local App1 executable
/apps/<Domain>/App1/spec -- (external) interface specification
/apps/<Domain>/App1/src -- source code
/apps/<Domain>/App1/test -- test

```

## List of supported Application(s)

- wcsUTelnet - Telnet (status:*planned*)
- wcsPFtp - wcs:Protocol interpreter : File-Transfer (status:*planned*)
- wcsPHttp - HTML eXchanger (status:*planned*)

**Dependencies** /lang - implementation language specific files

## Application Notes

### Digital signatures

49bf9fabbb31ca80119c72109b914708 ./apps/README.md

Document	MD5-Checksum
README.md	49bf9fabbb31ca80119c72109b914708 apps/README.md

## Distributed Apps

/dapps:

/dapps/DApp1 -- Distributed App1 executable  
/dapps/<Domain>/DApp1/spec -- (external) interface specification  
/dapps/<Domain>/DApp1/src -- source code  
/dapps/<Domain>/DApp1/test -- test

### List of native DApp(s)

- wcsDAppDeductor {src, dst} - Deduces 10% from src *monthly* into dst (status:*planned*)
- wcsDAppTemplate - WCS network DApp Template (status:*planned*)

### List of registered third-party DApp(s)

## Application Notes

### Digital signatures

8bac8aaca025fbaed5d9753a66a1b7cd ./dapps/README.md

Document	MD5-Checksum
README.md	8bac8aaca025fbaed5d9753a66a1b7cd dapps/README.md

/commands:

### Digital signatures

Document	MD5-Checksum
GLOSSARY.md README.md	

/operations:

### Digital signatures

Document	MD5-Checksum
GLOSSARY.md README.md	

### Tools

/tools:

#### List of native (off-the-shelf) Tool(s)

- wcsUPing - wcs:Util : Ping (status:*planned*)
- wcsUEcho - Echo (status:*planned*)

### Application Notes

#### Digital signatures

d89cf07d2b13a22deec08c0c4ed9266e ./tools/README.md

Document	MD5-Checksum
README.md	d89cf07d2b13a22deec08c0c4ed9266e tools/README.md

### Services

/services:

#### List of supported System-Service(s)

- wcsServer - World Compensation System server (status:*planned*)

## Digital signatures

086f662610232b3d3c823e5af2138bce    ./services/README.md

Document	MD5-Checksum
README.md	086f662610232b3d3c823e5af2138bce services/README.md

## Architecture

/arch:

### Concept

Create a World Compensation Ecosystem based on Decentralised Financial Applications.

### Implementation:

- Operation System (including fs, dfs, time-shared applications)

wcsOS – linux based distribution

### Layers:

1. Distributed peer-2-peer (P2P) Network (Blockchain based)
2. Distributed File system (dfsWcs)
3. Nodes are Servers
4. Servers
  1. run System- and Users-services
  2. route User- and System- interactions (transactions)
5. Users are Clients
6. Clients decide to participate or not (mounting/unmounting) as service suppliers in the network
7. Clients interact with other Clients
8. Clients request services from Servers (service suppliers)
  1. Via Remote Procedure Call (RPC) returning values in JSON format
9. Clients transfer value-assets to single or multiple-users or services
10. Light-Clients connect and use the network only for short-time (SMS, PPP)
11. Value-assets are represented via Addresses in the Distributed File system
12. Clients and Servers interact via read/write file operations with eachother
13. Servers providing User-services are debted certain agreed amount per-use
14. Servers providing System-services are debted an agreed amount per-use, daily, monthly or yearly on donation basis



## Network

**Topology:** Flower or Tree-of-Life (sacred geometry star 1:N, N:=6) \*  
[https://en.wikipedia.org/wiki/Overlapping\\_circles\\_grid#Modern\\_usage](https://en.wikipedia.org/wiki/Overlapping_circles_grid#Modern_usage)

### Nodes:

#### Full-nodes:

- store the complete history of command-blocks (analog to batch-files (a.k.a transactions))

#### Light-nodes:

- store, validate and reconstruct environment from all nodes in local network (only) -- bis

(light-Node) User : reads, modifies and stores state in Blockchain

- In browser (js Linux machine)
- Web App
- Mobile App

full-Node (service provider server) : offers (shares) execution time

- USB live CD
- Floppy disk

## Local File-system

Refer to /arch/fs

## Remote (distributed) File-system

Refer to /arch/dfs

## Realisation

TCP/IP Server :port UNIX's "Everything is a File" -> (name:Address) - Network  
(distributed) File-System

## Support Tools

- neo4j - Graph Database
- jslinux - Web-Browwer Linux
- 128-bit OS - 128-bit RISC OS
- quickjs - Embeddable Javascript engine

Refer to support tools.

## Network Startup

```
$wcsStart &  
World Compensation System server (wcsc) running  
Listening on port:280182
```

```
$wcsStatus  
Status: OK
```

## Command, Services and Tools

- bin - Local commands, services and tools
- dbin - Distributed (Remote) user-commands and user-tools

## Application Notes

- apps - Local commands, services and tools
- dapps - Distributed (Remote) User-services

## Digital signatures

aa4209ef73003fc5ed561c333358fa92 ./arch/README.md

Document	MD5-Checksum
README.md	aa4209ef73003fc5ed561c333358fa92 arch/README.md

## 2020 (CC) Creative Common License

59631d4266e7bb2116db1f7c09b6f47a ./whitepaper/README.md

## Digital signatures

d89cf07d2b13a22deec08c0c4ed9266e ./tools/README.md  
2002ca1741c76f6377bbeeee61871f2c ./dbin/README.md  
6eec0042d8bf26b963570b205d79b536 ./bin/README.md  
94063115eb82858ccfd15ef5a3b21814 ./development/integration.md  
4c5388669c4868fb9a92f0e8cef58327 ./development/README.md  
68f05ceb68281268217108fb55876082 ./development/deployment.md  
5cd4aa50a1a9f8d1b46b0b63c9d82e27 ./development/CONTRIBUTING.md  
9c060f1741bc37163838ead55b73c8ab ./development/workproducts.md  
3582056e21f163e556a92a29f26da4bc ./GLOSSARY.md  
8bac8aaca025fbaed5d9753a66a1b7cd ./dapps/README.md

```

ca8f6611e7334b5878a412f6908fab36 ./whitepaper/platform.md
35610a8fc82e44c38b636eb7a8c4403f ./whitepaper/wcsOES.md
d3777eb628218cf79d50e576d5c95bbd ./whitepaper/customer.md
f80bb3d6110f0bb9d5b829e1c45b0fd0 ./whitepaper/ecosystem.md
ff424f581783a3931c1c6f8b67e60a8c ./whitepaper/README.md
8bdd07641a7f2ad579cc6ab27cc90147 ./README.md
cabaa25d0481b17f4fe563869c86a467 ./arch/ledger/README.md
bdb604d4b1a062ad395e255c5fe46ca6 ./arch/dfs/README.md
b6a84991d4f8957e69ccfd6d3e935e02 ./arch/dfs/dFSwcs/README.md
7641cae0c8a83ae31000144d576289e2 ./arch/networking/README.md
93224d53add26ee895587864ebdf3c00 ./arch/agreement/README.md
227488574263a442dfc666513ef23f0c ./arch/operations/README.md
aa4209ef73003fc5ed561c333358fa92 ./arch/README.md
f16bab90fe5bf837c86b04e89f7dbb86 ./arch/lang/README.md
2eaaf2bbe0e2dae25cbc17345d4ba75a ./arch/lang/c/README.md
626a048f9af6cb0a4326140a20f61c35 ./arch/commands/README.md
beb28c51736bf8eb435fbc02654657f6 ./arch/fs/README.md
e598ee8dfcac71c6efdc5a3c56954ac0 ./users/README.md
c15908d22552696e8d67679f60717794 ./users/user1/README.md
49bf9fabbb31ca80119c72109b914708 ./apps/README.md
a3c6c1e9fbc0dd9e6723f73f7402b08a ./services/wcsServer/README.md
086f662610232b3d3c823e5af2138bce ./services/README.md

```

Whitepaper. WCS - published by: wcs:root : Sun Jan 19 16:45:29 CET 2020

## Glossary

- term - short explanation

## Definitions

### Assets:

Work-product (document, software application, drawing (diagram), picture) that was created and can be: - stored, - used (executed, applied to), or - exchanged (ownership transfer: lend/borrow, inheritance, etc.)

### Value:

The intrinsic 'cost' of producing and using a given asset - *universal* unit of credit: cost (node operation cost) - present value: (development + production cost) - future value: (service cost) – reputation based

**Transaction (Asset (Credits or Debts)-value Transfer):** - Blockchain->Browser->Distributed File-System->Local File-System->Blockchain->Remote File-System

1. User->Client (Application)
2. Client->Browser
3. Browser (Peer)-to-(Peer)-Network (Blockchain)
4. ...

**Credits:**

credit - a given amount granted in exchange of local-resource-time (normally computing power or manual labour) consumption/usage - run a service in a remote:node

**Debits:**

debit - crediting unit referring the used time during required computation -  
debiting-time - time used during computation \* fix cost (internet, electricity) -  
visibility (production cost, service price), e.g. (2,34)

**Services:**

service - an action executed virtually (software) or physically (hardware) -  
eg. (software:x386:server) hashvalueof bigfile.bin - (hardware:raspberrypi:io)  
getoutsidetemp