

Spaaw – Spreadsheet as a Wallet

By [Donald McIntyre](#)

Abstract. When Bitcoin started to grow in popularity, one of its great promises was [the ability to program money and accounts](#), enabling agreements, using smart contracts. The problem was that [the limits of Bitcoin's language and its tiny memory](#) turned out to be a barrier to support such functionality. Then, platforms like [Ethereum Classic \(ETC\)](#) came up with a solution creating a blockchain that could store smart contracts and execute them in a safe environment. However, the new problem was that to write and deploy such smart contracts in ETC remained a difficult task, especially for consumers and businesses. Spaaw solves this problem and makes smart contract management available to the general public on a global scale. On the Spaaw platform, smart contracts are as easy as using a spreadsheet.

1. Introduction

Spaaw (Spreadsheet as a Wallet) is a web, desktop and mobile application that serves as an Ethereum Classic wallet and smart contracts management system (SCMS). The distinction, and hence its name, is that the UI is a spreadsheet application format very similar to Excel or Google Sheets.

However, the difference to a full service spreadsheet is that, in addition to serving as an ETC wallet, Spaaw has preset templates that serve different functions which may, once populated with data, formulas, calculations and ETC wallet addresses, be converted into smart contracts, that contain the logic of the spreadsheets, and deployed on the ETC network.

Once deployed, these smart contracts may be called to execute their code and the results are constantly represented in the corresponding Spaaw sheets as if they were local calculations.

All normal transactions in ETC network accounts and smart contracts such as `transfer`, `deploy contract` or `contract call` can be sent from the Spaaw spreadsheets using cell functions directly. Read actions such as `balance`, `activity`, `sends` and `receives` can be posted as data to use in the spreadsheets using cell functions as well.

Spaaw sheets can be shared with other Spaaw users or businesses to create and manage agreements, to make bilateral or multilateral deals or projects, or manage team and business finances or property with specific rules and permissions.

Spaaw sheet templates can be created by developers or users, and can be found in the “Spaaw template store” or created directly on new sheets in the user’s application.

If successful, the Spaaw platform may eventually become a large platform connecting users, businesses, financial services, and developers.

2. Sheet Templates

There will be some default standard sheet templates and sheet templates that can be acquired from independent developers in a Spaaw sheet template store. The purpose of sheet templates is for users to have simple, standard and tested smart contracts to deploy on the ETC network that represent the agreements, cashflows and values they entered.

2.1. Default Standard Sheet Templates

The four standard default sheets will be:

1. A wallet management sheet template: A spreadsheet that shows the user's balances and activity with the same components as a normal ETC wallet. This sheet will usually be the "home" when users open their Spaaw app.
2. A simple contractor sheet template: A spreadsheet that lets users start a contractor relationship with another Spaaw user, where one is the customer and the other is the provider.
3. A small business cashflow sheet template: A one column budget with editable income and expense lines.
4. A small business twelve month cashflow sheet template: A twelve column budget with editable income and expense lines.

2.2. Sample Wallet Management Sheet Template

The Spaaw application default home will be similar to opening a normal online spreadsheet application, but it will automatically open some default sheets as well, with the wallet sheet first:

The screenshot shows the Spaaw application interface. At the top, there is a menu bar with options: File, Edit, View, Insert, Format, Data, Tools, Add-ons, Help. Below the menu bar is a toolbar with various icons for undo, redo, print, and formatting. The main area is a spreadsheet with columns A through E and rows 1 through 21. The spreadsheet is divided into sections: 'My ETC wallets' (rows 3-11) and 'Activity' (rows 13-19). The 'My ETC wallets' section lists three accounts with their respective balances. The 'Activity' section lists transactions with their respective amounts.

	A	B	C	D	E
1					
2					
3		My ETC wallets		Balance	
4					
5		Account 1	0x29ab23cdd98133940095351f32f06C7c3aAD4Fa875deff	33	
6					
7		Account 2	0x940095351f29ab23cdd9813332f06C7c3aAD4Fa998fdee	4.5	
8					
9		Account 3	0xaAD4Fa875deff29095351f32f06C7c3ab23cdd98133940	12	
10					
11			Total	49.5	
12					
13		Activity			
14					
15		Send	0xf35f0b5F754e81D1FdfE0ca4e9D8B83fF2d03ef4	1	
16		Receive	0x3ff2d03ef4f35f0bfE0ca4e9D8B85F754e81D1Fd	2	
17		Receive	0xA332D8B83fF2d03ef4f35f0b5F754e81D1FdfE0ca	3.34	
18		Send	0x35f0b5F754e81DfE0ca4e9D8B83fF2d03ef4fccc	0.25	
19		Send	0x4e9D8B83fF2d03ef4f35f0b5F754e81D1FdfE0ca	0.45	
20					
21					

At the bottom of the application, there is a navigation bar with tabs: '+', 'Wallet', 'Simple Contractor', 'Small biz CF', and 'Small biz 12 mo'.

Using the classic spreadsheet tabs at the bottom of the application, users can navigate through the sheets or reorganize them or add or create new ones.

2.3. Sample Simple Contractor Sheet Template

The Simple Contractor template has the function of putting two Spaaw users to buy and sell a service delivered in a digital format.

2.3.1. Terms Box

In the following example, Alice hires Bob to deliver a mockup, in PDF format, of a design of a website. The image below is the “terms box” with the terms of the Simple Project between Alice and Bob:

Simple Contractor Terms Box

Spaaw Simple Contractor Terms	
Customer	Alice
Provider	Bob
Product	Mockup
Price	\$800
Format	PDF
Agreement date	1/15/2020
Delivery date	3/15/2020
Tolerance	6/15/2020
Customer collateral deposit	100%
Payment mode	PVD

- Alice and Bob are entered in the cells with their handles on the Spaaw app. Following traditional spreadsheet formats, the function to enter the names in the cells may be with the format `=@handle`, so in the case of this contract it would be `=@Alice` and `=@Bob`. This action, automatically adds their ETC public keys or wallets where Bob gets paid to, and Alice will pay from.
- The “product” line is filled with text of the name of the project Alice and Bob agreed on.
- The “price” is the price agreed by Alice and Bob for the project.
- The “format” is the delivery file format agreed by Alice and bob.
- The “agreement date”, “delivery date” and “tolerance” are the key dates of the smart contract. If the PDF is delivered between the “agreement date” and the “delivery date”, the smart contract will pay automatically. If the delivery is made between the “delivery date” and the “tolerance” date, the smart contract will also pay. If the delivery is not done before the “tolerance” final date, then the money is returned to the customer.

- The “customer collateral deposit” is the percentage of the money the customer must deposit in the smart contract in advance so the provider can start to work on the project. The deposit date sets the “agreement date”.
- The “payment mode” is the way in which the payments will occur, in this case PVD (payment versus deliver) was selected, where the full payment is made upon delivery of the PDF document.

2.3.2. Smart Contract Perspective

The “smart contract perspective” is a view of the drivers, cashflow and verified events of the contract between Alice and Bob from the smart contract’s standpoint. The smart contract may be understood as a third entity, related to the simple contractor deal, on the ETC blockchain with its independent inflows, outflows and verified events, which correspond to the respective inflows, outflows and verified events of Alice and Bob:

Simple Contractor Smart Contract Perspective						
Drivers	Jan-2020	Feb-2020	Mar-2020	Apr-2020	May-2020	Jun-2020
Customer	1	0	0	0	0	0
Provider	1	0	0	0	0	0
Units	1	0	0	0	0	0
Price	\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Customer collateral deposit %	100%	0%	0%	0%	0%	0%
Delivery verification	0	0	0	1	0	0
Cashflow						
Customer collateral deposit \$	\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Payment to provider	\$0.00	\$0.00	\$0.00	-\$800.00	\$0.00	\$0.00
Return to customer	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Net	\$800.00	\$0.00	\$0.00	-\$800.00	\$0.00	\$0.00
Balance	\$800.00	\$800.00	\$800.00	\$0.00	\$0.00	\$0.00

- The “drivers” section contains the items of the contractor terms that will drive the execution of the smart contract, thus the flows of money and verification events. In this case, there is one customer and one provider, one unit sold at a price of \$800, and the collateral expected to be deposited is 100% of the sale value, \$800. All these parameters were set for January of 2020 and did not change, nor were there any other or new items, during the rest of the lifetime of the deal.
- The “delivery verification” line is the place where it is shown when and if the items of the contract were delivered. In this case a PDF file, which was delivered in April of 2020. The action of the delivery of the agreed product or service is called a “verified event”, a concept that will be explained below in this document.
- The “cashflow” section contains the inflows and outflows that occurred in the smart contract, which happen to correspond to the outflows and inflows from Alice and Bob’s perspectives as well, as will be seen below. These are activated by contract rules and verified events. In this case,

Alice deposited the required collateral in January, another type of “verified event”, and the smart contract paid \$800 to Bob in April, when it was verified that the PDF was delivered.

- The “balance” line is the cash position of the smart contract at every time. As it can be seen, from January to April the smart contract had a position of \$800 because Alice sent the required deposit, but when the delivery of the PDF occurred, then the \$800 were paid to Bob, thus the position of the smart contract became \$0.

2.3.3. Provider Perspective

The “provider perspective” is a view of the drivers, cashflow and verified events of the contract between Alice and Bob from the provider’s standpoint:

Simple Contractor Provider Perspective						
Drivers	Jan-2020	Feb-2020	Mar-2020	Apr-2020	May-2020	Jun-2020
Customer	1	0	0	0	0	0
Provider	1	0	0	0	0	0
Units	1	0	0	0	0	0
Price	\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Customer collateral deposit %	100%	0%	0%	0%	0%	0%
Delivery verification	0	0	0	1	0	0
Customer collateral deposit \$	\$800.00	\$800.00	\$800.00	-\$800.00	\$0.00	\$0.00
Cashflow						
Revenue	\$0.00	\$0.00	\$0.00	\$800.00	\$0.00	\$0.00
Net	\$0.00	\$0.00	\$0.00	\$800.00	\$0.00	\$0.00

- The “drivers” section is identical to the smart contract perspective, but includes the “Customer collateral deposit \$” line because the actual deposit of collateral is the driver or indicator for the provider to start working on the project.
- As seen in the image, when the provider delivers the PDF, it is verified in the smart contract and then the collateral deposit is debited from the smart contract and transferred to the provider’s account, which shows in the “cashflow” section in the “revenue” line.

2.3.4. Customer Perspective

The “customer perspective” is a view of the drivers, cashflow and verified events of the contract between Alice and Bob from the customer’s standpoint:

Simple Contractor Customer Perspective						
Drivers	Jan-2020	Feb-2020	Mar-2020	Apr-2020	May-2020	Jun-2020
Customer	1	0	0	0	0	0
Provider	1	0	0	0	0	0
Units	1	0	0	0	0	0
Price	\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Customer collateral deposit %	100%	0%	0%	0%	0%	0%
Delivery verification	0	0	0	1	0	0
Customer collateral deposit \$	\$800.00	\$800.00	\$800.00	-\$800.00	\$0.00	\$0.00
Cashflow						
To customer collateral deposit \$	-\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Return to customer	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Net	-\$800.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00

- The “drivers” section is identical to the provider perspective. In the “Customer collateral \$” line, the customer will be able to see the deposit sent from the customer’s account.
- In the “cashflow” section, the customer account is debited when the collateral is sent. This can be done by entering the value in the spreadsheet in the “Customer collateral \$” and accepting to send the transaction when prompted.
- As seen in the image, when the smart contract verifies that the PDF was sent, the funds in the collateral account will be debited.
- If the provider had never sent the PDF, the smart contract would have debited the collateral account and returned the funds to the customer account at the end of the tolerance date, 6/15/2020, which would have been seen in the customer’s “cashflow” section” as an inflow in the “Return to customer” line.

2.4. Sample Small Business Cashflow Sheet Template

The Small Business Cashflow sheet template has the function of helping businesses or team leaders to budget and automate across trust boundaries the inflows and out flows of their small business or team.

As may be seen in the image below, the small business cashflow has the drivers, revenues, expenses, and net income sections:

Small biz CF

Drivers	2020
Customers	7800
Units per customer	6.25
Units Sold	56000
Price	\$11.67
Materials costs per unit	\$3.00
Production costs per unit	\$3.50
Income tax rate	20%

Gross revenues	\$704,900
Materials	\$192,200
Production	\$213,100
Net Revenues	\$299,600

Wages Alice	\$60,000
Wages Bob	\$48,000
Wages Charlie	\$36,000
Marketing expenses	\$6,000
Office WeWork	\$9,600
Utilities	\$3,000
AWS	\$3,000
Travel	\$7,200
Total Expenses	\$172,800

Income before taxes	\$126,800
Taxes (20% of income)	\$32,460
Net income	\$94,340

- In the drivers section, the data of amount of customers, units per customer, units sold and price can be captured automatically as they occur, especially if they are transactions that happen within the blockchain.
- The materials and production costs can be derived from separate smart contracts entered with the business providers, thus linking smart contracts of different functions and parts of the business, or even external smart contracts.
- The revenues section is the sum of transactions that generate the gross revenues, related expenses and payments to materials and production providers, and the resulting net revenues. The payments of materials and production costs can be automated as transactions occur or made according to the specific smart contracts entered with the corresponding providers.
- The expenses section is the sum of expenses, which include wages, marketing, utilities and other typical expenses that small businesses or teams incur in. All types of expenses may be paid as transactions occur or according to smart contracts with each team member or provider. For example, Alice's wages can be paid monthly, weekly, or in any fashion agreed with her. Payments to WeWork, Utilities or AWS can be made as they send their bills or in a fixed schedule.
- The income section includes the income before taxes, the tax that must be paid and the net income. The taxes may be paid as a fixed amount or percentage as transactions occur, as a ratio based on earnings, or according to specific smart contracts with the corresponding governments and taxing authorities.

2.5. Sample Small Business Twelve Month Cashflow Sheet Template

The Small Business Twelve Month Cashflow sheet template has the same function as the cashflow template, but is spread over a twelve month period.

The twelve month cashflow template has the same basic structure as the cashflow template:

Small biz CF 12 mo												
Drivers	Jan-2020	Feb-2020	Mar-2020	Apr-2020	May-2020	Jun-2020	Jul-2020	Aug-2020	Sep-2020	Oct-2020	Nov-2020	Dec-2020
Customers	100	200	300	400	500	600	700	800	900	1000	1100	1200
Units per customer	3	3	5	6	5	6	7	8	8	7	8	9
Units Sold	300	600	1500	2400	2500	3600	4900	6400	7200	7000	8800	10800
Price	\$10	\$10	\$10	\$11	\$11	\$11	\$12	\$12	\$12	\$13	\$14	\$14
Materials costs per unit	\$2	\$2	\$2	\$3	\$3	\$3	\$3	\$3	\$3	\$4	\$4	\$4
Production costs per unit	\$3	\$3	\$3	\$3	\$3	\$3	\$4	\$4	\$4	\$4	\$4	\$4
Income tax rate	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
Gross revenues	\$3,000	\$6,000	\$15,000	\$26,400	\$27,500	\$39,600	\$58,800	\$76,800	\$86,400	\$91,000	\$123,200	\$151,200
Materials	\$600	\$1,200	\$3,000	\$7,200	\$7,500	\$10,800	\$14,700	\$19,200	\$21,600	\$28,000	\$35,200	\$43,200
Production	\$900	\$1,800	\$4,500	\$7,200	\$7,500	\$10,800	\$19,600	\$25,600	\$28,800	\$28,000	\$35,200	\$43,200
Net Revenues	\$1,500	\$3,000	\$7,500	\$12,000	\$12,500	\$18,000	\$24,500	\$32,000	\$36,000	\$35,000	\$52,800	\$64,800
Wages Alice	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000	\$5,000
Wages Bob	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000	\$4,000
Wages Charlie	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000	\$3,000
Marketing expenses	\$500	\$500	\$500	\$500	\$500	\$500	\$500	\$500	\$500	\$500	\$500	\$500
Office WeWork	\$800	\$800	\$800	\$800	\$800	\$800	\$800	\$800	\$800	\$800	\$800	\$800
Utilities	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250
AWS	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250	\$250
Travel	\$600	\$600	\$600	\$600	\$600	\$600	\$600	\$600	\$600	\$600	\$600	\$600
Total Expenses	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400	\$14,400
Income before taxes	-\$12,900	-\$11,400	-\$6,900	-\$2,400	-\$1,900	\$3,600	\$10,100	\$17,600	\$21,600	\$20,600	\$38,400	\$50,400
Taxes (20% of income)	\$0	\$0	\$0	\$0	\$0	\$720	\$2,020	\$3,520	\$4,320	\$4,120	\$7,680	\$10,080
Net income	-\$12,900	-\$11,400	-\$6,900	-\$2,400	-\$1,900	\$2,880	\$8,080	\$14,080	\$17,280	\$16,480	\$30,720	\$40,320

- All the inflows and outflows in the twelve month cashflow sheet can be programmed.

- Recurring payments, as the wages, can be set to be paid monthly (or any other frequency).
- Non-recurring payments can be set to be paid as the bills are received from external sources, at specific dates, or at dates negotiated and agreed upon in other smart contracts with the corresponding providers.
- Data inputs from events and verified events can be entered at the sheet layer and then relayed to the underlying smart contracts (contract calls), or data can be captured by smart contracts (also contract calls) from external sources in the underlying blockchain and relayed (read) by sheets to display and/or create new transactions or contract calls.

3. Spaaw Sheet Template Store

The template store is a section in Spaaw where users can search and find useful templates, for different functions, which generate smart contracts stored and executable in the ETC network. These sheets can be free or available for a fee.

4. Sheet Cell Functions

All normal functions will be available on the Spaaw spreadsheets. In addition, to add the functionality needed to make the sheets manage accounts and build, deploy and interact with smart contracts on ETC, there will be more cell functions included. This is a list of functions that will be added:

- `=USER(ETC, @handle)`: Adds a Spaaw user, thus his/her ETC public address, to a cell or function in a sheet.
- `=ACC(ETC, 0xaccount)`: Adds an ETC public account to a cell or function.
- `=BAL(ETC, 0xaccount)`: Adds and ETC account balance to a cell or function.
- `=ACT(ETC, 0xaccount)`: Adds the activity of an account to a cell or a group of aligned cells.
- `=OUT(ETC, 0xaccount)`: Adds outgoing transfers of an account to a cell or group of aligned cells.
- `=IN(ETC, 0xaccount)`: Adds incoming transfers of an account to a cell or group of aligned cells.
- `=SEND(ETC, 0xaccount, amount, fee, date, time)`: Adds a send transaction to a cell or function, which is sent to the ETC network when executed. If user add date and/or time, then it is executed in that date and/or time.

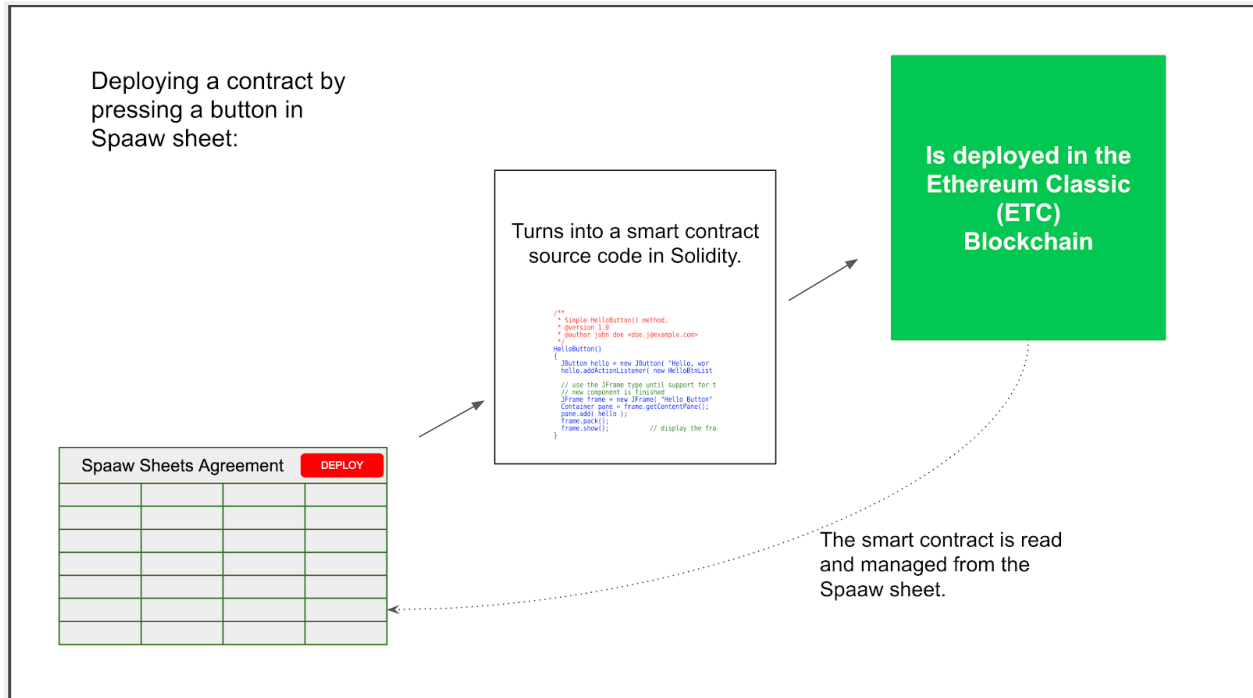
There will likely be more functions added to the Spaaw application in the future.

5. How Spaaw Sheets are Turned into Smart Contracts

As explained before, when users enter data, formulas, calculations and ETC wallet addresses, the template sheets can then be turned into smart contracts and deployed on the Ethereum Classic blockchain. The steps are as follows:

1. @Alice and @Bob (they can be in any part of the world, but both use Spaaw) enter into an agreement.
2. When they press the DEPLOY button, the sheet content (including formulas and macros) are compiled into a smart contract in Solidity source code, or other ETC compatible languages.
3. Then, that smart contract is deployed into the Ethereum Classic blockchain.

Diagram:



4. From then on, the smart contract is going to execute the code and be read and called from the Spaaw sheet, and interact with other external sources as well.
5. All data and money movements will be posted in the spreadsheet @Alice and @Bob built for their contract.

6. Verified Events

All external data that is entered into sheets or is sent to smart contracts are external events that require a person or some external source to send it. These inputs are key to execute the parts of agreements that are critical to the deals or assets being managed. For example, if a customer sends a collateral deposit, that signals the provider to start their work. If a provider delivers their work, then the smart contract must receive that information so the collateral deposit is transferred to the provider.

In the case of all external events that are generated on-chain, e.g. normal transactions and smart contract operations, the smart contracts deployed by Spaaw sheets will read them and act upon them accordingly.

In the case of off-blockchain external events, they will likely be processed on the Spaaw application where a “delivery” input/output service section will be provided so Spaaw sheets can send the data to the blockchain.

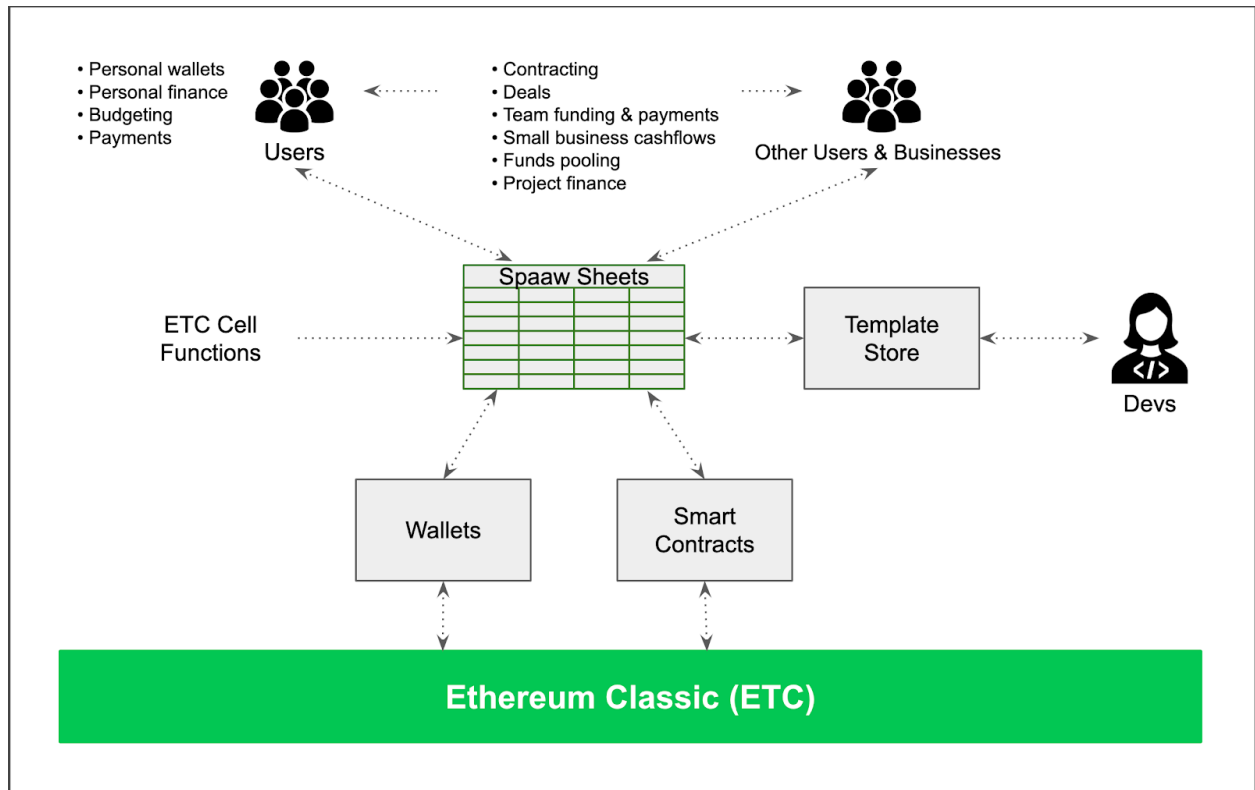
When formal oracles or other services emerge, Spaaw sheets and smart contracts will be able to read them.

7. Private Keys – Wallets Supported

Spaaw will not host private keys and will follow the model of MyCrypto, MyEtherWallet and other wallets, where users can connect their hardware wallet, Metamask, Parity Signer, Private Key, Keystore file, mnemonic phrase or other secure private key management systems that may exist in the future.

8. Business Model

The Spaaw business model is similar to WordPress, as an open source platform, and Automattic, Inc., as the company that maintains and supports it. In WordPress, the WordPress content management system (CMS) is the heart of the platform. In the case of Spaaw sheets, it is a smart contracts management system (SCMS) where Spaaw sheets are at the heart of the model:

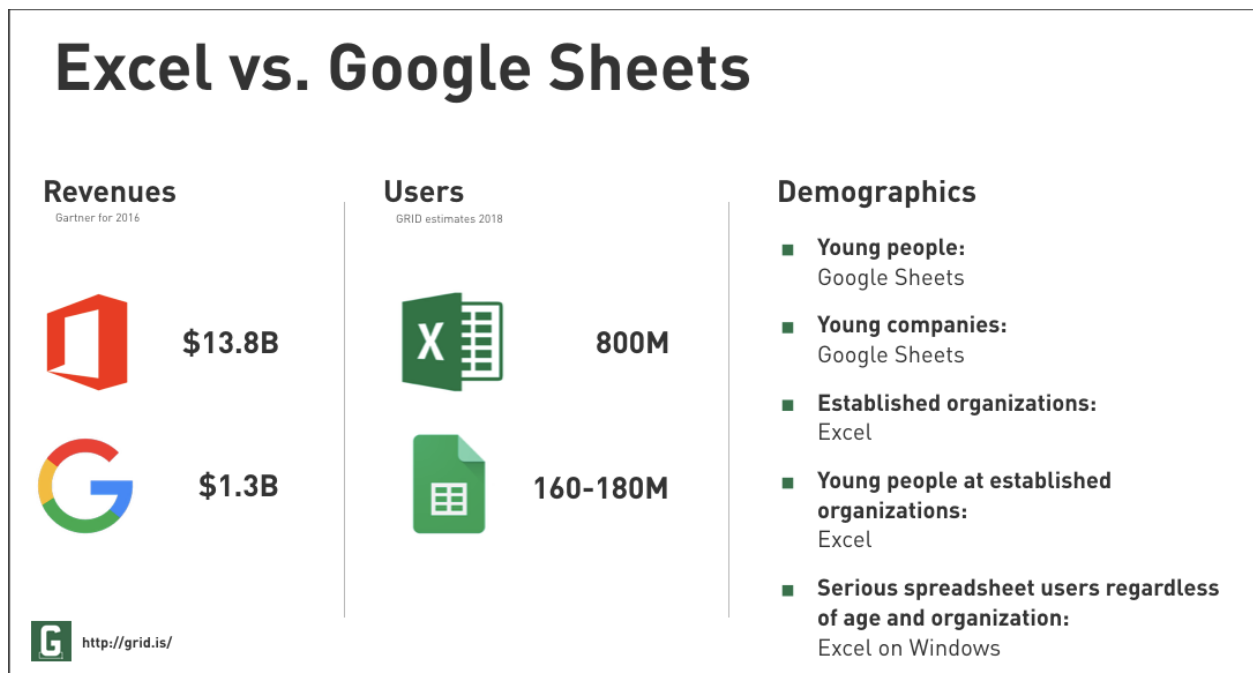


- The center of the ecosystem is the Spaaw sheets SCMS platform.
- Around the platform, a rich marketplace of smart contracts with user-to-user, user-to-business, and user-to-developer relationships may ensue and flourish.
- Users can connect with other users or businesses through Spaaw sheets to establish the terms of their agreements and then deploy them to the ETC network.
- These agreements may be: Basic contracting, other spot deals, team funding and payments, small business cashflows, funds pooling for causes or purchases, project finance, and any other agreements that may be enabled using sheets, templates and smart contracts.
- Users can also use Spaaw sheets to manage their own personal finances by tracking income and expenses, budgeting, making payments, scheduling recurring payments, etc.
- Users can also program sheets to plan for long term goals, such as college education or retirement funding.
- Teams and small businesses may use sheets for 401k style retirement plans.
- A template store where developers make sheets available for free, or charging fees, will be accessible through the application.
- The purpose of templates is to make sure users use tested and standard smart contracts rather than overly complex or flawed code.
- Future ancillary services provided by the Spaaw application may be on-chain third party financial products, such as insurance, lending, investing, etc.
- The layers of the system are: sheets -> smart contracts & wallets -> ETC network.
- For scalability, in the future the system may evolve to have a sidechain: sheets -> smart contracts & wallets -> Spaaw sidechain -> ETC network.

- Like with WordPress themes and plugins, the Spaaw Sheet Template Store may become a popular marketplace of smart contracts and an important source of revenues for smart contract developers.

9. Conclusion

Since spreadsheets are very versatile, were the first killer app of personal computers, thus a large population of around 970 million people in the world use them regularly (see image below), and are used for very diverse functions -from finance and accounting to engineering and academia- the possibilities for applying Spaaw sheets to more use cases is very large.



Source: <https://medium.com/grid-spreadsheets-run-the-world/excel-vs-google-sheets-usage-nature-and-numbers-9dfa5d1cadbd>

The opportunity of turning Spaaw sheets into a large platform connecting users, businesses, financial services, and developers is very significant.

The Spaaw Sheets Platform can be trust minimized, socially scalable and global across all trust boundaries.

Find Spaaw on Github: <https://github.com/TokenHash/Spaaw>