An attempt to give the most abstract possible definition of a blockchain.

1 BlockChain

Definition. *We call Validated Rules a function noted* V*:*

$$V : (\Sigma^*)^\times \to \mathscr{P}(\Sigma^*)$$

Remark. *Intuitively* V *is a function taking a list of block in input and returning the set of block which are valid.*

Definition. *We call* (G, V) *validated chain,where* $G \in (\Sigma^*)^\times$*, noted* $log_{G,V}$ *a function*

$$log_{G,V} : \mathbb{N} \to \Sigma^*$$

such that:

$$\begin{aligned}
&G(0) \in V(\emptyset)\\
&\forall i \in [\![0; |G|]\!], log(i) = G(i)\\
&\forall i \in \mathbb{N}^+, log(i) \in V(log_{G,V}(0), , , log_{G,V}(i-1))
\end{aligned}$$

Remark. G *is the list of genesis block to startup the system.* $log_{G,V}$ *would be an infinite chain that is valid regarding* V*.*

Notation.

$$\forall i, log_{G,V}(i)^- = (log_{G,V}(0), , , log_{G,V}(i))$$
$$LOG_{G,V} = \{f | f \text{ is a (G,V) validated chain}\}$$

Remark. *We introduce* $LOG_{G,V}$ *which really complicated but is actually necessary to deal with fork and consensus later.*

Definition. *A set of validated chain* $LOG_{G,V}$ *is said to be infinite if:*

$$\forall log_{G,V} \in LOG_{G,V}, \forall i \in \mathbb{N}, \forall b \in V(log_{G,V}(i)-), V(log_{G,V}(i)-, b) \neq \emptyset$$

Remark. *Infinite here is used in a sense that whatever instance of a* G, V *validated-chain we are dealing with we will always be able to complete it.*

Definition. *A set of validated chain* $LOG_{G,V}$ *is said secured if :*

$$\forall log_{G,V} \in LOG_{G,V}, \forall i \in \mathbb{N}^+, \forall b, b' \in \Sigma, b \neq b' \implies V(log_{G,V}(i)^-, b) \cap V(log_{G,V}(i)^-, b') = \emptyset$$

Remark. *Intuitively in order to be secured* $V(, , b)$ *should depend on* b *as bitcoin include previous hash block.*

Definition. *We call player's knowledge a tuple* (P, K_T) *where* P *is a set of player and* K_T *a function:*

$$K_T : P \times [0; T] \times \mathbb{R}^+ \to \mathscr{P}(\Sigma^* \times]0; 1])$$

such that:

$$\begin{aligned}
&\forall p \in P, \forall t \in [0; T], (b, \alpha) \in K_T(t, 0, p) \implies \alpha = 1\\
&\forall p \in P, \forall t, t' \in [0; T], t' \geq t \implies K_T(t', 0, p) \subseteq K(t, 0, p)\\
&\forall p \in P, \forall t \in [0; T], \forall \delta \in \mathbb{R}^+, K_T(t, 0, p) \subseteq K_T(t, \delta, p)\\
&\forall p \in P, \forall t \in [0; T], \forall \delta, \delta' \in \mathbb{R}^+, \delta' \geq \delta \implies \forall (b, \alpha) \in K_T(p, t, \delta), \exists (b, \alpha') \in K_T(p, t, \delta'), \alpha' \geq \alpha
\end{aligned}$$

Notation. $\forall p \in P, \forall t \in [0;T]$ *we denote* $\{b|(b,1) \in K_T(p,t,0)\} : K_T(p,t)$

Definition. *Let* $T, T' \in \mathbb{R}^+$ *such that* $T > T'$ *we say that* $K'_{T'}$ *extend* K_T *if* $\forall p, K_T(p,T) = K'_{T'}(p,T)$

Definition. *A block chain protocol is a function noted* $P_{G,V}$*:*

$$P_{G,V} : (LOG_{G,V} \times \mathbb{N}) \times (LOG_{G,V} \times \mathbb{N}) \times T \rightarrow (LOG_{G,V} \times \mathbb{N})$$

such that :

$$\forall log_{G,V}, log'_{G,V} \in LOG_{G,V}, \forall n, n' \in \mathbb{N}, \forall t \in T; P_{G,V}(log_{G,V}, n, log'_{G,V}, n', t) = (log_{G,V}, n) \vee (log_{G,V}, n')$$

Remark. $P_{G,V}$ *can be seen as the rule in case of fork and new block. Have to be improve to impose that there is no cycle (an order ?)*

Definition. *Considering an validated chain* $LOG_{G,V}$*, a player's knowledge* (P, K_T) *and a block chain protocol* $P_{G,V}$*. We denote* $S_{t,p}$ *where* $t \in [0,T]$ *and* $p \in P$ *the set of tuple :*

$$S_{t,p} = \{log_{G,V}(N)^- | log_{G,V} \in LOG_{G,V} \wedge log_{G,V}(N)^- \subseteq K_T(p,t)\}$$

We call BlockChain at time $t \in [0,T]$ *for user* $p \in P$ *noted* $BC_{t,p}$ *the tuple:*

$$BC_{t,p} \in S_{t,p}$$
$$\forall log \in S_{t,p}, P_{G,V}((log, |log|), (BC_{t,p}, |BC_{t,p}|), t) = (BC_{t,p}, |BC_{t,p}|)$$

Remark. *Intuitively the blockchain for a user* p *at a time* t *is the best chain he fully knows regarding the protocol function and the validity at time* t *(time-stamping).*

Definition. *We denote* α^* *the function*

$$\mathbb{R}^+ \times LOG_{G,V} \times N \times P \rightarrow [0,1]$$

such that :

$$\alpha^*(\delta, log_{G,V}, N, p) = max\{\alpha | \exists b; (b,\alpha) \in K_T(p,T,\delta) \cap V(log_{G,V}(N)^-)\}$$

We said that a $LOG_{G,V}$ *is alive regarding* (P, K_T) *iff:*

$$\exists p, \exists log_{G,V}, \exists N, log_{G,V}(N)^- \in K_T(p,T) \wedge V(log_{G,V}(N)^-) \cap K_T(p,T) = \emptyset \wedge lim_{\delta \to \infty} \alpha^*(\delta, log_{G,V}, N, p) = 1$$

2 Draft

Definition. *We call an alive set of validated chain a tuple* $(LOG_{G,V}, P, K_P)$ *where* $LOG_{G,V}$ *is an set of infinite validated chain and* P, K_P *an alive set of player.*

Proposition. *Let* $(LOG_{G,V}, P, K_P)$ *an alive set of validated chain then:*

$$\forall log_{G,V} \in LOG_{G,V}, \forall i \in \mathbb{N}, \exists p \in P, \exists t \in T, V(log_{G,V}(i)^-) \cap K_P(p,t) \neq \emptyset$$

Remark. *To be honest i am not sure as we are dealing with infinite number. I may have to trick things here. I want to ensure the fact that the chain will eventually move forward.*