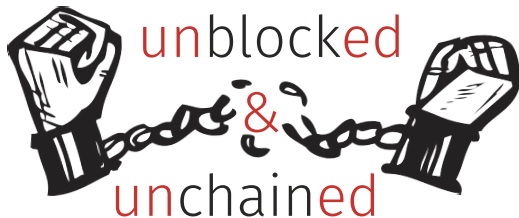


Introduction technocritique aux blockchains



Pablo Rauzy <pr@up8.edu>
pablo.rauzy.name

Pour aller plus loin : pablockchain.fr

Blockchain : quèsaco ?

Définition

- Une *blockchain* est un registre distribué unique et immuable qui ne doit nécessiter aucune confiance pour garantir ces propriétés.
 - *registre* : document où sont consignés des faits ;
 - *distribué* : chaque participant en possède une copie ;
 - *unique* : il ne doit en exister qu'une seule version ;
 - *immuable* : il doit être impossible d'y modifier une information consignée.
- Tout cela doit être mis en œuvre :
 - en l'absence de confiance envers quiconque (excepté soi-même),
 - sans pouvoir connaître l'ensemble des participants.

Définition

- Une *blockchain* est un registre distribué unique et immuable qui ne doit nécessiter aucune confiance pour garantir ces propriétés.
 - *registre* : document où sont consignés des faits ;
 - *distribué* : chaque participant en possède une copie ;
 - *unique* : il ne doit en exister qu'une seule version ;
 - *immuable* : il doit être impossible d'y modifier une information consignée.
- Tout cela doit être mis en œuvre :
 - en l'absence de confiance envers quiconque (excepté soi-même),
 - sans pouvoir connaître l'ensemble des participants.

Définition

- Une *blockchain* est un registre distribué unique et immuable qui ne doit nécessiter aucune confiance pour garantir ces propriétés.
 - *registre* : document où sont consignés des faits ;
 - *distribué* : chaque participant en possède une copie ;
 - *unique* : il ne doit en exister qu'une seule version ;
 - *immuable* : il doit être impossible d'y modifier une information consignée.
- Tout cela doit être mis en œuvre :
 - en l'absence de confiance envers quiconque (excepté soi-même),
 - sans pouvoir connaître l'ensemble des participants.

- Le registre doit servir à effectuer des *transactions* en “cryptomonnaie” :
 - les transactions y sont ajoutées par *bloc*,
 - avant chaque ajout on vérifie la validité des transactions grâce à l'historique,
 - à chaque ajout on doit garantir les propriétés (distribution, unicité, immuabilité).
- Consigner dans le registre autre chose que des transactions en “cryptomonnaie” ?
 - C'est le principe d'un nombre important de projet autour des blockchains :
 - certification de documents (actes de propriété, diplômes, ...),
 - contractualisation automatisée (notariat, contrats, ...)
 - traçabilité (supply chain, agro-industrie, ...),
 - démocratie (vote électronique)...
 - On reviendra là dessus une fois outillé-es sur le fonctionnement.

- Le registre doit servir à effectuer des *transactions* en “cryptomonnaie” :
 - les transactions y sont ajoutées par *bloc*,
 - avant chaque ajout on vérifie la validité des transactions grâce à l'historique,
 - à chaque ajout on doit garantir les propriétés (distribution, unicité, immuabilité).
- Consigner dans le registre autre chose que des transactions en “cryptomonnaie” ?
 - C'est le principe d'un nombre important de projet autour des blockchains :
 - certification de documents (actes de propriété, diplômes, ...),
 - contractualisation automatisée (notariat, contrats, ...)
 - traçabilité (supply chain, agro-industrie, ...),
 - démocratie (vote électronique)...
 - On reviendra là dessus une fois outillé-es sur le fonctionnement.

- Le registre doit servir à effectuer des *transactions* en “cryptomonnaie” :
 - les transactions y sont ajoutées par *bloc*,
 - avant chaque ajout on vérifie la validité des transactions grâce à l'historique,
 - à chaque ajout on doit garantir les propriétés (distribution, unicité, immuabilité).
- Consigner dans le registre autre chose que des transactions en “cryptomonnaie” ?
 - C'est le principe d'un nombre important de projet autour des blockchains :
 - certification de documents (actes de propriété, diplômes, ...),
 - contractualisation automatisée (notariat, contrats, ...)
 - traçabilité (supply chain, agro-industrie, ...),
 - démocratie (vote électronique)...
 - On reviendra là dessus une fois outillé-es sur le fonctionnement.

Fonctionnement d'une blockchain

- Comment assurer les propriétés d'immuabilité, de distribution, et d'unicité ?

- Chaque bloc est identifié par son contenu avec un condensat cryptographique et est *chaîné* au bloc au précédent :
 - l'intégrité du contenu de chaque bloc est vérifiable avec son identifiant,
 - chaque bloc contient l'identifiant du précédent,
 - ce “chaînage” permet de vérifier l'intégrité de l'ensemble du registre.
- On *simule* ainsi l'*immuabilité*.

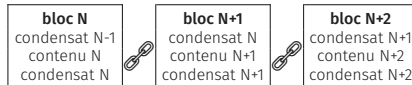
- Chaque bloc est identifié par son contenu avec un condensat cryptographique et est *chaîné* au bloc au précédent :
 - l'intégrité du contenu de chaque bloc est vérifiable avec son identifiant,
 - chaque bloc contient l'identifiant du précédent,
 - ce "chaînage" permet de vérifier l'intégrité de l'ensemble du registre.
- On *simule* ainsi l'*immuabilité*.

Condensat cryptographique

- Un *condensat cryptographique* est une empreinte numérique.
 - Grand nombre de taille fixe calculé déterministiquement.
 - Extrêmement difficile (~impossible) à inverser, grâce l'*effet d'avalanche*.
 - Exemple avec l'algorithme SHA-256 [NIS02] :
 - "Unblockchained" :
`41e14c7eae48db5da0e8f55837214bfa9c4bed2da49d222caac2ca8d7a7bacf7,`
 - "unblockchained" :
`8ddb32c8390d8ece5b7cd9fbcc00571ca3e20c9b4a2b4a4ff65cbd9e4855f999,`
 - "U" est encodé par `01110101` tandis que "u" l'est par `01010101`.
- Cela permet d'*identifier* une donnée.

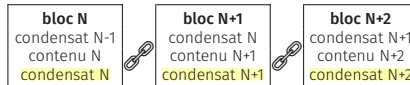
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



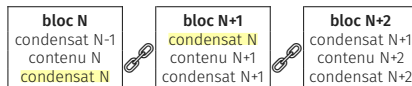
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



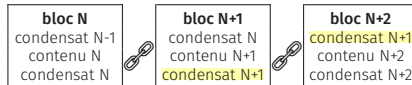
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



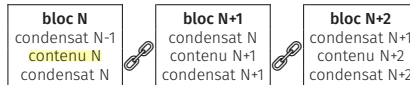
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



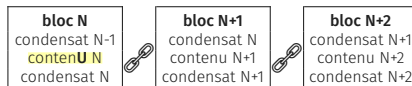
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



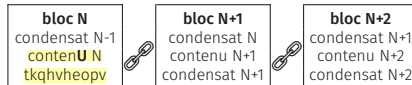
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



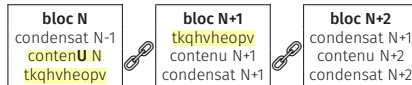
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



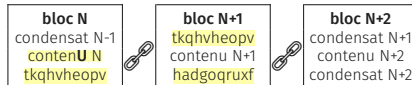
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



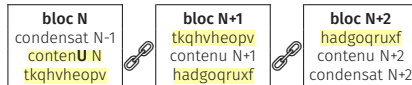
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



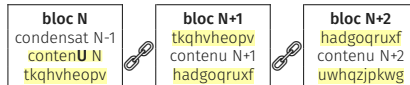
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



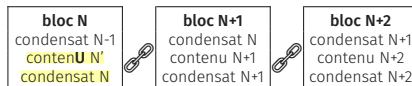
L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une collision sur le condensat du bloc modifié.



L'intégrité d'une blockchain

- Dans une blockchain, chaque bloc est identifié par son condensat.
 - On peut facilement vérifier son intégrité en recalculant son condensat.
 - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de chaîne.
 - Chaque bloc contient l'identifiant du précédent.
 - C'est un cas particulier d'*arbre de Merkle* [Mer87].
 - La modification ou la suppression d'un bloc entraîne celle de tous les suivants...
 - ... sauf si on trouve une **collision** sur le condensat du bloc modifié.



Collision

- On a une *collision* quand deux données différentes ont le même condensat.
- On est sûr que ça existe (principe des *tiroirs à chaussettes*).
- Mais trouver une collision est extrêmement difficile (~impossible).
 - D'autant plus si on cherche une collision avec un condensat particulier (exit le "*paradoxe*" des anniversaires).
 - Temps de calcul estimable en milliards de milliards de milliards d'années (sauf en cas de faille dans l'algorithme, bien sûr).

- L'immuabilité d'une blockchain est donc produite par :
 - la possibilité de vérifier son intégrité ;
 - sa distribution, pour avoir des points de comparaison.
-  L'immuabilité n'est *pas* produite par la preuve de travail (ou d'enjeu).

Distribution du registre

- Lorsque les participants se sont mis d'accord sur un bloc à ajouter au registre :
 - ce bloc est transmis en pair-à-pair à l'ensemble des participants au réseau,
 - chacun peut vérifier indépendamment la validité du bloc et de ses transactions,
 - et l'ajouter si tout va bien à sa copie locale du registre.
- On met ainsi en œuvre la *distribution*.
- Il reste à s'assurer de l'*unicité*...

Distribution du registre

- Lorsque les participants se sont mis d'accord sur un bloc à ajouter au registre :
 - ce bloc est transmis en pair-à-pair à l'ensemble des participants au réseau,
 - chacun peut vérifier indépendamment la validité du bloc et de ses transactions,
 - et l'ajouter si tout va bien à sa copie locale du registre.
- On met ainsi en œuvre la *distribution*.
- Il reste à s'assurer de l'*unicité*...

Distribution du registre

- Lorsque les participants **se sont mis d'accord** sur un bloc à ajouter au registre :
 - ce bloc est transmis en pair-à-pair à l'ensemble des participants au réseau,
 - chacun peut vérifier indépendamment la validité du bloc et de ses transactions,
 - et l'ajouter si tout va bien à sa copie locale du registre.
- On met ainsi en œuvre la *distribution*.
- Il reste à s'assurer de l'*unicité*... dans un contexte de défiance généralisée.

Unicité du registre

- Le registre sert à consigner des transactions en “cryptomonnaie” :
 - les transactions ne sont qu'un jeu d'écriture, donc
 - on ne peut pas faire sans l'unicité (double dépense, création de “monnaie”, etc.),
 - (ces problèmes ne se posent pas avec de la monnaie physique).
- On sait garantir l'unicité d'un registre quand :
 - soit il est centralisé (une seule source d'autorité),
 - soit il est distribué mais les participants coopèrent pour tenir leur version à jour.
- Dans le cas d'une blockchain, on ne fait confiance à personne par hypothèse.
 - Seule solution : simuler la centralisation du registre.
 - Pour ça il est nécessaire de résoudre le problème du *consensus distribué*.

Unicité du registre

- Le registre sert à consigner des transactions en “cryptomonnaie” :
 - les transactions ne sont qu'un jeu d'écriture, donc
 - on ne peut pas faire sans l'unicité (double dépense, création de “monnaie”, etc.),
 - (ces problèmes ne se posent pas avec de la monnaie physique).
- On sait garantir l'unicité d'un registre quand :
 - soit il est centralisé (une seule source d'autorité),
 - soit il est distribué mais les participants coopèrent pour tenir leur version à jour.
- Dans le cas d'une blockchain, on ne fait confiance à personne par hypothèse.
 - Seule solution : simuler la centralisation du registre.
 - Pour ça il est nécessaire de résoudre le problème du *consensus distribué*.

Unicité du registre

- Le registre sert à consigner des transactions en “cryptomonnaie” :
 - les transactions ne sont qu'un jeu d'écriture, donc
 - on ne peut pas faire sans l'unicité (double dépense, création de “monnaie”, etc.),
 - (ces problèmes ne se posent pas avec de la monnaie physique).
- On sait garantir l'unicité d'un registre quand :
 - soit il est centralisé (une seule source d'autorité),
 - soit il est distribué mais les participants ~~coopèrent~~ pour tenir leur version à jour.
- Dans le cas d'une blockchain, on ne fait confiance à personne par hypothèse.
 - Seule solution : simuler la centralisation du registre.
 - Pour ça il est nécessaire de résoudre le problème du *consensus distribué*.

Consensus distribué

- Principe du *consensus distribué* :
 - mettre l'ensemble des participants "d'accord" sur une valeur (n'importe laquelle)
 - qui dans le cas d'une blockchain sera le prochain bloc à ajouter au registre.
- Dans le contexte de défiance généralisée des blockchains :
 - on ne fait confiance à personne et on ne connaît pas l'ensemble des participants,
 - la taille des blocs est limitée et les participants ont des intérêts divergents,
 - impossible de satisfaire tout le monde, organiser un vote, ou faire "chacun son tour".
- Seule solution : recourir à un *tirage au sort non contestable*.

Consensus distribué

- Principe du *consensus distribué* :
 - mettre l'ensemble des participants "d'accord" sur une valeur (n'importe laquelle)
 - qui dans le cas d'une blockchain sera le prochain bloc à ajouter au registre.
- Dans le contexte de défiance généralisée des blockchains :
 - on ne fait confiance à personne et on ne connaît pas l'ensemble des participants,
 - la taille des blocs est limitée et les participants ont des intérêts divergents,
 - impossible de satisfaire tout le monde, organiser un vote, ou faire "chacun son tour".
- Seule solution : recourir à un *tirage au sort non contestable*.

Consensus distribué

- Principe du *consensus distribué* :
 - mettre l'ensemble des participants "d'accord" sur une valeur (n'importe laquelle)
 - qui dans le cas d'une blockchain sera le prochain bloc à ajouter au registre.
- Dans le contexte de défiance généralisée des blockchains :
 - on ne fait confiance à personne et on ne connaît pas l'ensemble des participants,
 - la taille des blocs est limitée et les participants ont des intérêts divergents,
 - impossible de satisfaire tout le monde, organiser un vote, ou faire "chacun son tour".
- Seule solution : recourir à un *tirage au sort non contestable*.

Illustration du consensus distribué

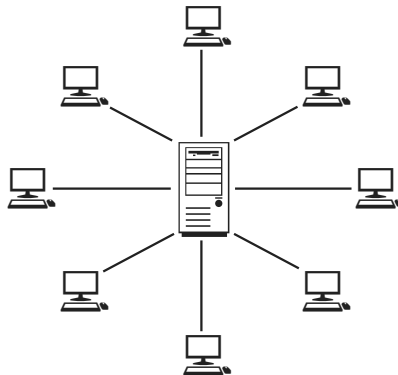


Illustration du consensus distribué

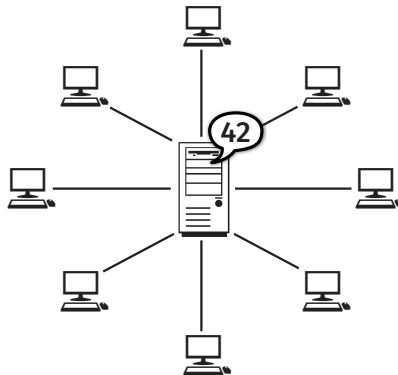


Illustration du consensus distribué

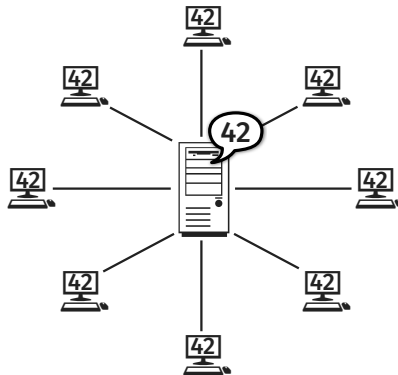


Illustration du consensus distribué

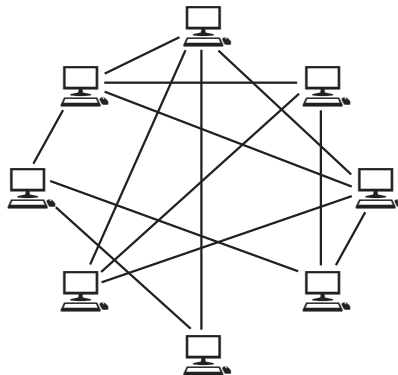


Illustration du consensus distribué

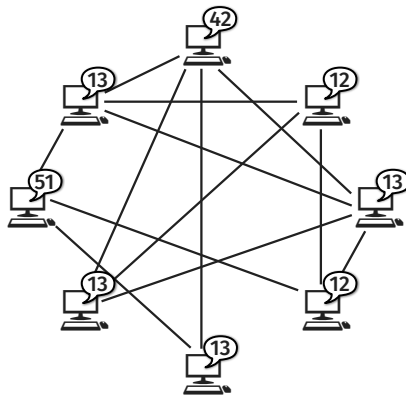
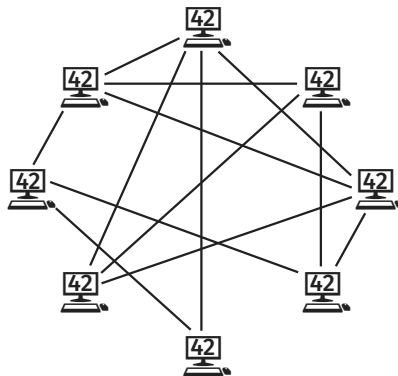


Illustration du consensus distribué



Tirage au sort distribué non contestable ("minage")

- On connaît essentiellement deux moyens de faire ça :
 - le "*minage*" par preuve de travail, et le "*minage*" par preuve d'enjeu.
- La *preuve de travail* consiste à faire un paquet de calculs inutiles, jusqu'à trouver *par hasard* la solution d'une inéquation de la forme $\mathcal{H}(\text{bloc}, x) < \text{seuil}$.
 - Impossible de prédire qui trouvera une solution en premier : chaque participant ayant un *bloc* différent, le plus petit *x* satisfaisant l'équation est différent pour chacun.
 - Étant donné un *bloc*, une solution *x* sert de *preuve* du travail en vérifiant l'inéquation.
- La *preuve d'enjeu* consiste en un tirage au sort pondéré par une *mise*.
 - La mise est un montant de "cryptomonnaie" bloquée/pariée pour le tirage.
 - Les mises sont publiques, donc le résultat du tirage ne doit pas surprendre.
 - Idée que plus on peut miser gros, plus on a intérêt à être réglo.
 - Moins solide que la preuve de travail en terme de sécurité, rarement auto-suffisante.

Tirage au sort distribué non contestable ("minage")

- On connaît essentiellement deux moyens de faire ça :
 - le "*minage*" par preuve de travail, et le "*minage*" par preuve d'enjeu.
- La *preuve de travail* consiste à faire un paquet de calculs inutiles, jusqu'à trouver *par hasard* la solution d'une inéquation de la forme $\mathcal{H}(\text{bloc}, x) < \text{seuil}$.
 - Impossible de prédire qui trouvera une solution en premier : chaque participant ayant un *bloc* différent, le plus petit x satisfaisant l'équation est différent pour chacun.
 - Étant donné un *bloc*, une solution x sert de *preuve* du travail en vérifiant l'inéquation.
- La *preuve d'enjeu* consiste en un tirage au sort pondéré par une *mise*.
 - La mise est un montant de "cryptomonnaie" bloquée/pariée pour le tirage.
 - Les mises sont publiques, donc le résultat du tirage ne doit pas surprendre.
 - Idée que plus on peut miser gros, plus on a intérêt à être réglo.
 - Moins solide que la preuve de travail en terme de sécurité, rarement auto-suffisante.

Tirage au sort distribué non contestable ("minage")

- On connaît essentiellement deux moyens de faire ça :
 - le "*minage*" par preuve de travail, et le "*minage*" par preuve d'enjeu.
- La *preuve de travail* consiste à faire un paquet de calculs inutiles, jusqu'à trouver *par hasard* la solution d'une inéquation de la forme $\mathcal{H}(\text{bloc}, x) < \text{seuil}$.
 - Impossible de prédire qui trouvera une solution en premier : chaque participant ayant un *bloc* différent, le plus petit *x* satisfaisant l'équation est différent pour chacun.
 - Étant donné un *bloc*, une solution *x* sert de *preuve* du travail en vérifiant l'inéquation.
- La *preuve d'enjeu* consiste en un tirage au sort pondéré par une *mise*.
 - La mise est un montant de "cryptomonnaie" bloquée/pariée pour le tirage.
 - Les mises sont publiques, donc le résultat du tirage ne doit pas surprendre.
 - Idée que plus on peut miser gros, plus on a intérêt à être réglo.
 - Moins solide que la preuve de travail en terme de sécurité, rarement auto-suffisante.

Tirage au sort par la preuve de travail

- L'idée de l'équation $\mathcal{H}(\text{bloc}, x) < \text{seuil}$ est ce qu'on appelle une *collision partielle*.
 - On s'attend à trouver une collision partielle avec le condensat égal à zéro.
 - La difficulté *moyenne* du "minage" est adaptable avec la taille de la collision attendue.
 - Initialement inventée contre le spam [DN92], première mise en œuvre [Bac97].
- Pour un cas précis, impossible de prédire le nombre de calculs nécessaires.
 - Exemple pour une collision sur 16 bits :
 - De "Unblockchained0" il faut aller jusqu'à "Unblockchained72572" :
0000f2a03f7389093acd17574254fae3dd2e8ec30a855e75ed4444a02eb2d828.
 - De "unblockchained0" il faut aller jusqu'à "unblockchained108232" :
0000422af4278ba7c3a1b2c5c036ba25e2e41e0a8d44ab8ecc54bcfda46c205a.
 - ~50% de calculs en plus.
- Même si cela reste fortement corrélé à la puissance de calcul disponible, il est techniquement impossible de prédire qui trouvera une collision partielle en premier.

Nécessité fonctionnelle de la "cryptomonnaie"

- Dans le cas de la preuve de travail :
 - le coût des calculs rend nécessaire une récompense comme incitation à participer,
 - cette récompense doit provenir intrinsèquement de la blockchain.
- Dans le cas de la preuve d'enjeu :
 - une récompense est nécessaire comme incitation à miser,
 - les mises doivent exister intrinsèquement sur la blockchain,
 - les montants misés doivent provenir intrinsèquement de la blockchain.

 Une blockchain ne peut pas fonctionner sans sa "cryptomonnaie".

Nécessité de donner de la valeur à la "cryptomonnaie"

- Pour que la "cryptomonnaie" joue son rôle, il faut qu'elle ait une valeur...
 - mais, par définition, il ne s'agit que d'une écriture, liée à rien dans la vraie vie.
- Aucun souci dans un mode de pensée ultra-libéral / libertarien...
 - La "valeur" d'une chose ne peut être que marchande,
 - elle est purement le fruit de la loi de l'offre et de la demande pour cette chose,
 - elle est donc directement corrélée à la rareté de la chose.
- ... il suffit d'en limiter la quantité et de mettre en place un marché ! 🐼
 - La valeur d'un crypto-actif correspond à ce que le prochain pigeon est prêt à payer pour.
 - À rapprocher des *pyramides de Ponzi* ou de la pratique du *pump and dump*.



Les "cryptomonnaies" sont des actifs purement spéculatifs.

Aller plus loin : https://fr.wikipedia.org/wiki/Théorie_du_plus_grand_fou

Nécessité de donner de la valeur à la "cryptomonnaie"

- Pour que la "cryptomonnaie" joue son rôle, il faut qu'elle ait une valeur...
 - mais, par définition, il ne s'agit que d'une écriture, liée à rien dans la vraie vie.
- Aucun souci dans un mode de pensée ultra-libéral / libertarien...
 - La "valeur" d'une chose ne peut être que marchande,
 - elle est purement le fruit de la loi de l'offre et de la demande pour cette chose,
 - elle est donc directement corrélée à la rareté de la chose.
- ... il suffit d'en limiter la quantité et de mettre en place un marché ! 🐼
 - La valeur d'un crypto-actif correspond à ce que le prochain pigeon est prêt à payer pour.
 - À rapprocher des *pyramides de Ponzi* ou de la pratique du *pump and dump*.

📄 Les "cryptomonnaies" sont des actifs purement spéculatifs.

Aller plus loin : https://fr.wikipedia.org/wiki/Théorie_du_plus_grand_fou

Nécessité de donner de la valeur à la "cryptomonnaie"

- Pour que la "cryptomonnaie" joue son rôle, il faut qu'elle ait une valeur...
 - mais, par définition, il ne s'agit que d'une écriture, liée à rien dans la vraie vie.
- Aucun souci dans un mode de pensée ultra-libéral / libertarien...
 - La "valeur" d'une chose ne peut être que marchande,
 - elle est purement le fruit de la loi de l'offre et de la demande pour cette chose,
 - elle est donc directement corrélée à la rareté de la chose.
- ... il suffit d'en limiter la quantité et de mettre en place un marché ! 🤖
 - La valeur d'un crypto-actif correspond à ce que le prochain pigeon est prêt à payer pour.
 - À rapprocher des *pyramides de Ponzi* ou de la pratique du *pump and dump*.

 Les "cryptomonnaies" sont des actifs purement spéculatifs.

Aller plus loin : https://fr.wikipedia.org/wiki/Théorie_du_plus_grand_fou

Fonctionnement d'une "cryptomonnaie"

- Une "*cryptomonnaie*" est un actif numérique échangeable sans autorité centrale.
 - Le préfixe "crypto" fait référence à la cryptographie utilisée par cette technologie.
 - Le suffixe "monnaie" fait référence à la nature autoproclamée de ces actifs.
- Le concept existe depuis le début des années 80 [Cha82].
- La version actuelle voit le jour avec Bitcoin, il y a un peu plus de 15 ans [Nak08].
 - La technologie de la blockchain a été inventé pour Bitcoin.
 - Beaucoup d'autres "cryptomonnaies" ont vu le jour depuis, avec quelques variations.
 - Bitcoin reste la plus importante.

Fonctionnement d'une "cryptomonnaie"

- Une "*cryptomonnaie*" est un actif numérique échangeable sans autorité centrale.
 - Le préfixe "crypto" fait référence à la cryptographie utilisée par cette technologie.
 - Le suffixe "monnaie" fait référence à la nature autoproclamée de ces actifs.
- Le concept existe depuis le début des années 80 [Cha82].
- La version actuelle voit le jour avec Bitcoin, il y a un peu plus de 15 ans [Nak08].
 - La technologie de la blockchain a été inventé pour Bitcoin.
 - Beaucoup d'autres "cryptomonnaies" ont vu le jour depuis, avec quelques variations.
 - Bitcoin reste la plus importante.

Fonctionnement d'une "cryptomonnaie"

- Une "*cryptomonnaie*" est un actif numérique échangeable sans autorité centrale.
 - Le préfixe "crypto" fait référence à la cryptographie utilisée par cette technologie.
 - Le suffixe "monnaie" fait référence à la **nature autoproclamée** de ces actifs.
- Le concept existe depuis le début des années 80 [Cha82].
- La version actuelle voit le jour avec Bitcoin, il y a un peu plus de 15 ans [Nak08].
 - La technologie de la blockchain a été inventé pour Bitcoin.
 - Beaucoup d'autres "cryptomonnaies" ont vu le jour depuis, avec quelques variations.
 - Bitcoin reste la plus importante.

Anatomie d'une transaction

- Une *transaction* a un *identifiant* : le condensat des données qui la compose.
- Elle est composée d'*entrées* (sources) et de *sorties* (cibles).
 - Les sorties sont numérotées et chacune d'elle est une paire associant :
 - un destinataire, et
 - un montant.
 - Chaque entrée est une paire associant :
 - une sortie de transaction passée (identifiant et numéro de sortie), et
 - une signature cryptographique prouvant l'autorisation de la dépense.

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie **7 pour Bob**, et 3 pour elle (son “rendu monnaie”).
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son “rendu monnaie”).
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son “rendu monnaie”).
 - est **signée par Alice**.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Exemple de transactions

- Alice veut transférer 7 à Bob.
- Admettons l'existence d'une transaction passée Tx42 dans laquelle Alice a reçu 10.
- Alice fait une transaction qui :
 - a en entrée la sortie de la Tx42 où elle a reçu 10,
 - a en sortie 7 pour Bob, et 3 pour elle (son “rendu monnaie”).
 - est signée par Alice.

Transaction passée

Entrées	Sorties
(Tx13, 4)	1. (Charlotte, 0.5)
(Tx21, 1)	2. (Alice, 10)
	3. (David, 1.3)
Identifiant : Tx42  (13,4)  (21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7)
	2. (Alice, 3)
Identifiant : Tx51  Alice	

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque signature cryptographique doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque signature cryptographique doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque signature cryptographique doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque signature cryptographique doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque signature cryptographique doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Validité d'une transaction

- Plusieurs conditions pour qu'une transaction soit valide.
 - La somme des montants des sorties doit être $\leq$ à celle des montants des entrées.
 - Tous les montants doivent être positifs.
 - Si la somme des sorties $<$ à la somme des entrées, le "mineur" qui a la transaction dans son bloc s'attribue la différence (frais de transaction).
 - Aucune des sorties de transactions listées en entrée ne doit avoir été dépensée.
 - Ni dans un bloc précédent, ni dans une autre transaction du même bloc.
 - On parle d'*UTXO* (*unspent transaction outputs*).
 - Chaque **signature cryptographique** doit être valide.
- ⇒ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

Cryptographie asymétrique et signature

- Avec la cryptographie symétrique, une même *clef secrète* sert à chiffrer et déchiffrer.
 - Problème de poule et d'œuf pour établir un canal de communication sécurisé.
- La *cryptographie asymétrique* [DH76, RSA78] résout ce problème.
 - Chaque participant a une paire de clefs : une *clef publique* et une *clef privée*.
 - La clef publique peut être distribuée à tout le monde.
 - La clef privée doit être gardée absolument secrète (y compris de ses interlocuteurs).
 - Ce qui est chiffré avec une clef n'est déchiffrable qu'avec l'autre clef de la même paire.
 - Pour envoyer un message secret à Bob, Alice chiffre le message avec la clef publique de Bob.
 - Elle est sûre que seul Bob pourra le lire : sa clef privée, dont lui seul dispose, est nécessaire.
- Dans les sorties d'une transaction, les cibles sont identifiées par leur clef publique.

Cryptographie asymétrique et signature

- Avec la cryptographie symétrique, une même *clef secrète* sert à chiffrer et déchiffrer.
 - Problème de poule et d'œuf pour établir un canal de communication sécurisé.
- La *cryptographie asymétrique* [DH76, RSA78] résout ce problème.
 - Chaque participant a une paire de clefs : une *clef publique* et une *clef privée*.
 - La clef publique peut être distribuée à tout le monde.
 - La clef privée doit être gardée absolument secrète (y compris de ses interlocuteurs).
 - Ce qui est chiffré avec une clef n'est déchiffrable qu'avec l'autre clef de la même paire.
 - Pour envoyer un message secret à Bob, Alice chiffre le message avec la clef publique de Bob.
 - Elle est sûre que seul Bob pourra le lire : sa clef privée, dont lui seul dispose, est nécessaire.
- Dans les sorties d'une transaction, les cibles sont identifiées par leur clef publique.

Cryptographie asymétrique et signature

- Avec la cryptographie symétrique, une même *clef secrète* sert à chiffrer et déchiffrer.
 - Problème de poule et d'œuf pour établir un canal de communication sécurisé.
- La *cryptographie asymétrique* [DH76, RSA78] résout ce problème.
 - Chaque participant a une paire de clefs : une *clef publique* et une *clef privée*.
 - La clef publique peut être distribuée à tout le monde.
 - La clef privée doit être gardée absolument secrète (y compris de ses interlocuteurs).
 - Ce qui est chiffré avec une clef n'est déchiffrable qu'avec l'autre clef de la même paire.
 - Pour envoyer un message secret à Bob, Alice chiffre le message avec la clef publique de Bob.
 - Elle est sûre que seul Bob pourra le lire : sa clef privée, dont lui seul dispose, est nécessaire.
- Dans les sorties d'une transaction, les cibles sont identifiées par leur clef publique.

Cryptographie asymétrique et signature

- Avec la cryptographie symétrique, une même *clef secrète* sert à chiffrer et déchiffrer.
 - Problème de poule et d'œuf pour établir un canal de communication sécurisé.
- La *cryptographie asymétrique* [DH76, RSA78] résout ce problème.
 - Chaque participant a une paire de clefs : une *clef publique* et une *clef privée*.
 - La clef publique peut être distribuée à tout le monde.
 - La clef privée doit être gardée absolument secrète (y compris de ses interlocuteurs).
 - Ce qui est chiffré avec une clef n'est déchiffrable qu'avec l'autre clef de la même paire.
 - Pour envoyer un message secret à Bob, Alice chiffre le message avec la clef publique de Bob.
 - Elle est sûre que seul Bob pourra le lire : sa clef privée, dont lui seul dispose, est nécessaire.
- Dans les sorties d'une transaction, les cibles sont identifiées par leur clef publique.

Cryptographie asymétrique et signature

- Avec la cryptographie symétrique, une même *clef secrète* sert à chiffrer et déchiffrer.
 - Problème de poule et d'œuf pour établir un canal de communication sécurisé.
- La *cryptographie asymétrique* [DH76, RSA78] résout ce problème.
 - Chaque participant a une paire de clefs : une *clef publique* et une *clef privée*.
 - La clef publique peut être distribuée à tout le monde.
 - La clef privée doit être gardée absolument secrète (y compris de ses interlocuteurs).
 - Ce qui est chiffré avec une clef n'est déchiffrable qu'avec l'autre clef de la même paire.
 - Pour envoyer un message secret à Bob, Alice chiffre le message avec la clef publique de Bob.
 - Elle est sûre que seul Bob pourra le lire : sa clef privée, dont lui seul dispose, est nécessaire.
- Dans les sorties d'une transaction, les cibles sont identifiées par leur clef publique.

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Signature

- Avec la cryptographie asymétrique, on peut réaliser des *signature cryptographique*.
 - Signature d'un message :
 - Pour générer la signature d'un message, Alice le chiffre avec sa clef privée.
 - Elle ajoute cette signature au message avant de l'envoyer à Bob.
 - Vérification de la signature :
 - Bob reçoit un message qui dit venir d'Alice, accompagné d'une signature.
 - Il déchiffre la signature avec la clef publique d'Alice pour voir s'il obtient bien le message.
 - Si oui, c'est bien Alice qui l'a signée : sa clef privée, dont elle seule dispose, était nécessaire.
- Les transactions doivent être signées par les cibles des UTXO qu'elles ont en entrées.

Transaction passée

Entrées	Sorties
(Tx13, 4) (Tx21, 1)	1. (Charlotte, 0.5) 2. (Alice, 10) 3. (David, 1.3)
Identifiant : Tx42  _(13,4)  _(21,1)	

Transaction d'Alice

Entrées	Sorties
(Tx42, 2)	1. (Bob, 7) 2. (Alice, 3)
Identifiant : Tx51  _{Alice}	

Incitation et récompense

- Il reste deux questions : la création de "cryptomonnaie", et l'incitation au "minage".
 - La première transaction de chaque bloc, qui sert de *récompense*, est particulière.
 - Elle n'a pas d'entrée.
 - Sa sortie sert à récompenser le(s) "mineur(s)" qui ont trouvé le bloc.
 - Le montant en sortie ne doit pas excéder la récompense + les frais des transactions.
 - La récompense est créée ex nihilo depuis la *coinbase*.
 - Les frais correspondent aux différences entre sorties et entrées des autres transactions.
 - Dans Bitcoin :
 - la récompense a démarré à 50 BTC le 9 janvier 2009 (premier bloc "miné") ;
 - elle est divisée par deux tous les 210 000 blocs (~4 ans), 3.125 BTC depuis le 19 avril 2024 ;
 - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction...
-  Les frais de transaction vont nécessairement devenir colossaux.

Incitation et récompense

- Il reste deux questions : la création de "cryptomonnaie", et l'incitation au "minage".
 - La première transaction de chaque bloc, qui sert de *récompense*, est particulière.
 - Elle n'a pas d'entrée.
 - Sa sortie sert à récompenser le(s) "mineur(s)" qui ont trouvé le bloc.
 - Le montant en sortie ne doit pas excéder la récompense + les frais des transactions.
 - La récompense est créée ex nihilo depuis la *coinbase*.
 - Les frais correspondent aux différences entre sorties et entrées des autres transactions.
 - Dans Bitcoin :
 - la récompense a démarré à 50 BTC le 9 janvier 2009 (premier bloc "miné") ;
 - elle est divisée par deux tous les 210 000 blocs (~4 ans), 3.125 BTC depuis le 19 avril 2024 ;
 - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction...
-  Les frais de transaction vont nécessairement devenir colossaux.

Incitation et récompense

- Il reste deux questions : la création de "cryptomonnaie", et l'incitation au "minage".
 - La première transaction de chaque bloc, qui sert de *récompense*, est particulière.
 - Elle n'a pas d'entrée.
 - Sa sortie sert à récompenser le(s) "mineur(s)" qui ont trouvé le bloc.
 - Le montant en sortie ne doit pas excéder la récompense + les frais des transactions.
 - La récompense est créée ex nihilo depuis la *coinbase*.
 - Les frais correspondent aux différences entre sorties et entrées des autres transactions.
 - Dans Bitcoin :
 - la récompense a démarré à 50 BTC le 9 janvier 2009 (premier bloc "miné") ;
 - elle est divisée par deux tous les 210 000 blocs (~4 ans), 3.125 BTC depuis le 19 avril 2024 ;
 - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction...
-  Les frais de transaction vont nécessairement devenir colossaux.

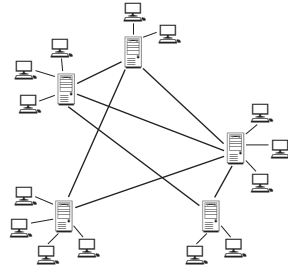
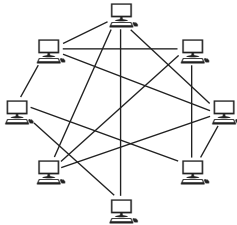
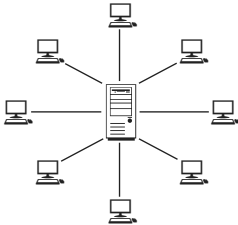
Incitation et récompense

- Il reste deux questions : la création de "cryptomonnaie", et l'incitation au "minage".
 - La première transaction de chaque bloc, qui sert de *récompense*, est particulière.
 - Elle n'a pas d'entrée.
 - Sa sortie sert à récompenser le(s) "mineur(s)" qui ont trouvé le bloc.
 - Le montant en sortie ne doit pas excéder la récompense + les frais des transactions.
 - La récompense est créée ex nihilo depuis la *coinbase*.
 - Les frais correspondent aux différences entre sorties et entrées des autres transactions.
 - Dans Bitcoin :
 - la récompense a démarré à 50 BTC le 9 janvier 2009 (premier bloc "miné") ;
 - elle est divisée par deux tous les 210 000 blocs (~4 ans), 3.125 BTC depuis le 19 avril 2024 ;
 - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction...
-  Les frais de transaction vont nécessairement devenir colossaux.

Points vocabulaire

Décentralisation ?

- Le réseau sous-jacent à une blockchain est *décentralisé*, le registre est *distribué*, et les transactions y sont *centralisées*.



Consensus ?

- Comme incitation à participer au “minage”, une récompense dans chaque bloc :
 - chaque “mineur” tente de se l’auto-attribuer,
 - donc aucun ne propose le même bloc.
- Le “consensus” se fait sur une valeur proposée par un contre *tous* les autres :
 - chaque “consensus” est en fait le tirage au sort d’un dictateur temporaire,
 - il s’agit en vérité d’une mise en concurrence généralisée.



La notion de “consensus” des blockchains est à l’extrême opposé du sens courant.

Aller plus loin : <https://p4bl0.net/post/2022/01/Vocabulaire--consensus>

Consensus ?

- Comme incitation à participer au “minage”, une récompense dans chaque bloc :
 - chaque “mineur” tente de se l’auto-attribuer,
 - donc aucun ne propose le même bloc.
- Le “consensus” se fait sur une valeur proposée par un contre *tous* les autres :
 - chaque “consensus” est en fait le tirage au sort d’un dictateur temporaire,
 - il s’agit en vérité d’une mise en concurrence généralisée.



La notion de “consensus” des blockchains est à l’extrême opposé du sens courant.

Aller plus loin : <https://p4bl0.net/post/2022/01/Vocabulaire--consensus>

Minage ?

- Le choix de l'analogie avec l'extraction minière n'est pas un accident :
 - l'analogie est déjà présente dans l'article d'introduction de Bitcoin,
 - elle n'est pas le fruit d'une tentative de vulgarisation, mais un choix de conception.
- Un choix politique et économique fortement marqué (et marquant) :
 - volonté explicite de créer un équivalent de l'or en version numérique (rareté, difficulté),
 - vision de la monnaie comme monnaie-marchandise (*métallisme*),
 - confusion entretenue avec le terme “mineurs” entre travailleurs et capitalistes.



L'idéologie libertarienne est inscrite dans les fondements mêmes de la technologie.

Aller plus loin : <https://p4bl0.net/post/2023/05/Vocabulaire-miner-minage-blockchain>

Cash ? Pair-à-pair ? Portefeuille ?

- Les transactions sont *centralisées* sur un registre *distribué*.
 - Seul le fonctionnement du réseaux sous-jacent est décentralisé / pair-à-pair.
 - La “cryptomonnaie” elle-même n'existe et ne s'échange que par un pur jeu d'écriture.
- “*Bitcoin: A Peer-to-Peer Electronic Cash System*” : deux mensonges dès le titre.
 - *Quand bien même* ce serait de la monnaie, ce n'est ni du liquide, ni pair-à-pair.
 - Même souci avec la notion de “*wallet*” (portefeuille), qui est en fait un “compte”.



Une transaction en “cryptomonnaie” est en fait l'équivalent d'un *virement interne*.

Aller plus loin : <https://p4bl0.net/post/2022/01/Vocabulaire--portefeuille>

"Cryptomonnaie" ?

- Pour être ainsi qualifiée, une monnaie doit pouvoir servir :
 - de réserve de valeur,
 - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
 - d'unité de compte, et
 - idem, même les taux de changes entre "cryptomonnaies" sont mesurés en vraie monnaie ;
 - d'intermédiaire d'échange.
 - n'arrive quasiment pas en pratique mais techniquement possible,
 - propriété triviale (points de fidélités, tickets de kermesse, ...).
- Par ailleurs, la monnaie *est* de la dette, par nature, depuis ses origines.
 - Cela implique, notamment, de la confiance *sociale*, niée par les blockchains.
 - D'où le refus de la monnaie-dette, et la volonté d'une forme de monnaie-marchandise.



La monnaie n'est pas un instrument neutre, ni politiquement ni économiquement.

Aller plus loin : <https://www.cairn.info/revue-economique-2008-4-page-813.htm>

"Cryptomonnaie" ?

- Pour être ainsi qualifiée, une monnaie doit pouvoir servir :
 - de réserve de valeur :
 - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
 - d'unité de compte :
 - idem, même les taux de changes entre "cryptomonnaies" sont mesurés en vraie monnaie ;
 - d'intermédiaire d'échange :
 - n'arrive quasiment pas en pratique mais techniquement possible,
 - propriété triviale (points de fidélités, tickets de kermesse, ...).
- Par ailleurs, la monnaie *est* de la dette, par nature, depuis ses origines.
 - Cela implique, notamment, de la confiance *sociale*, niée par les blockchains.
 - D'où le refus de la monnaie-dette, et la volonté d'une forme de monnaie-marchandise.



La monnaie n'est pas un instrument neutre, ni politiquement ni économiquement.

Aller plus loin : <https://www.cairn.info/revue-economique-2008-4-page-813.htm>

"Cryptomonnaie" ?

- Pour être ainsi qualifiée, une monnaie doit pouvoir servir :
 - de réserve de valeur :
 - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
 - d'unité de compte :
 - idem, même les taux de changes entre "cryptomonnaies" sont mesurés en vraie monnaie ;
 - d'intermédiaire d'échange :
 - n'arrive quasiment pas en pratique mais techniquement possible,
 - propriété triviale (points de fidélités, tickets de kermesse, ...).
- Par ailleurs, la monnaie *est* de la dette, par nature, depuis ses origines.
 - Cela implique, notamment, de la confiance *sociale*, niée par les blockchains.
 - D'où le refus de la monnaie-dette, et la volonté d'une forme de monnaie-marchandise.



La monnaie n'est pas un instrument neutre, ni politiquement ni économiquement.

Aller plus loin : <https://www.cairn.info/revue-economique-2008-4-page-813.htm>

"Cryptomonnaie" ?

- Pour être ainsi qualifiée, une monnaie doit pouvoir servir :
 - de réserve de valeur :
 - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
 - d'unité de compte :
 - idem, même les taux de changes entre "cryptomonnaies" sont mesurés en vraie monnaie ;
 - d'intermédiaire d'échange :
 - n'arrive quasiment pas en pratique mais techniquement possible,
 - propriété triviale (points de fidélités, tickets de kermesse, ...).
- Par ailleurs, la monnaie *est* de la dette, par nature, depuis ses origines.
 - Cela implique, notamment, de la confiance *sociale*, niée par les blockchains.
 - D'où le refus de la monnaie-dette, et la volonté d'une forme de monnaie-marchandise.



La monnaie n'est pas un instrument neutre, ni politiquement ni économiquement.

Aller plus loin : <https://www.cairn.info/revue-economique-2008-4-page-813.htm>

"Cryptomonnaie" ?

- Pour être ainsi qualifiée, une monnaie doit pouvoir servir :
 - de réserve de valeur :
 - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
 - d'unité de compte :
 - idem, même les taux de changes entre "cryptomonnaies" sont mesurés en vraie monnaie ;
 - d'intermédiaire d'échange :
 - n'arrive quasiment pas en pratique mais techniquement possible,
 - propriété triviale (points de fidélités, tickets de kermesse, ...).
- Par ailleurs, la monnaie *est* de la dette, par nature, depuis ses origines.
 - Cela implique, notamment, de la confiance *sociale*, niée par les blockchains.
 - D'où le refus de la monnaie-dette, et la volonté d'une forme de monnaie-marchandise.



La monnaie n'est pas un instrument neutre, ni politiquement ni économiquement.

Aller plus loin : <https://www.cairn.info/revue-economique-2008-4-page-813.htm>

Blockchain : quelles limites ?

La blockchain comme solution à son propre problème

- S'il existe une autorité extérieure ou un tiers de confiance :
 - on a pas besoin de résoudre le problème du consensus distribué,
 - donc pas besoin de blockchain (on sait faire mieux et moins coûteux).
- Dans une situation de défiance généralisée :
 - aucun tiers ne peut assurer la correspondance entre écriture et vérité,
 - blockchains intéressantes seulement quand c'est l'écriture qui *définit* la vérité,
 - seule application qui fait sens : les "cryptomonnaies".
- *La solution du problème de la solution* :
 - une blockchain a besoin d'avoir sa "cryptomonnaie" pour fonctionner,
 - pour exister, une "cryptomonnaie" a besoin de sa blockchain (unicité du registre).



Blockchains et "cryptomonnaies" sont des solutions qui sont leur propre problème.

Aller plus loin : <https://p4bl0.net/post/2022/02/Cryptomonnaie-blockchain-un-serpent-qui-se-mord-la-queue>

La blockchain comme solution à son propre problème

- S'il existe une autorité extérieure ou un tiers de confiance :
 - on a pas besoin de résoudre le problème du consensus distribué,
 - donc pas besoin de blockchain (on sait faire mieux et moins coûteux).
- Dans une situation de défiance généralisée :
 - aucun tiers ne peut assurer la correspondance entre écriture et vérité,
 - blockchains intéressantes seulement quand c'est l'écriture qui *définit* la vérité,
 - seule application qui fait sens : les “cryptomonnaies”.
- *La solution du problème de la solution :*
 - une blockchain a besoin d'avoir sa “cryptomonnaie” pour fonctionner,
 - pour exister, une “cryptomonnaie” a besoin de sa blockchain (unicité du registre).



Blockchains et “cryptomonnaies” sont des solutions qui sont leur propre problème.

Aller plus loin : <https://p4bl0.net/post/2022/02/Cryptomonnaie-blockchain-un-serpent-qui-se-mord-la-queue>

La blockchain comme solution à son propre problème

- S'il existe une autorité extérieure ou un tiers de confiance :
 - on a pas besoin de résoudre le problème du consensus distribué,
 - donc pas besoin de blockchain (on sait faire mieux et moins coûteux).
- Dans une situation de défiance généralisée :
 - aucun tiers ne peut assurer la correspondance entre écriture et vérité,
 - blockchains intéressantes seulement quand c'est l'écriture qui *définit* la vérité,
 - seule application qui fait sens : les "cryptomonnaies".
- *La solution du problème de la solution* :
 - une blockchain a besoin d'avoir sa "cryptomonnaie" pour fonctionner,
 - pour exister, une "cryptomonnaie" a besoin de sa blockchain (unicité du registre).

 Blockchains et "cryptomonnaies" sont des solutions qui sont leur propre problème.

Aller plus loin : <https://p4bl0.net/post/2022/02/Cryptomonnaie-blockchain-un-serpent-qui-se-mord-la-queue>

Le surcoût systématique du recours à une blockchain

- Pour vérifier la validité d'une transaction :
 - il faut entre autre vérifier que l'émetteur dispose bien de la “cryptomonnaie” dépensée.
- Une blockchain est une structure de données particulièrement inefficace :
 - elle liste seulement les *modifications* du système (les transactions),
 - le montant de “cryptomonnaie” disponible sur un compte correspond à ce qui y a déjà été encaissé mais qui n'a pas encore été dépensé.
 - si on ne dispose que du registre, il faut le relire en entier pour chaque vérification.
- En pratique, on est obligé de maintenir une vraie base de données :
 - on a besoin d'accéder à l'état du système directement,
 - on construit l'état consolidé qu'on tient à jour à chaque nouveau bloc.



Le coût d'une blockchain est toujours *en plus* et non *à la place* de l'alternative.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-cout-d-une-blockchain>

Le surcoût systématique du recours à une blockchain

- Pour vérifier la validité d'une transaction :
 - il faut entre autre vérifier que l'émetteur dispose bien de la "cryptomonnaie" dépensée.
- Une blockchain est une structure de données particulièrement inefficace :
 - elle liste seulement les *modifications* du système (les transactions),
 - le montant de "cryptomonnaie" disponible sur un compte correspond à ce qui y a déjà été encaissé mais qui n'a pas encore été dépensé.
 - si on ne dispose que du registre, il faut le relire en entier pour chaque vérification.
- En pratique, on est obligé de maintenir une vraie base de données :
 - on a besoin d'accéder à l'état du système directement,
 - on construit l'état consolidé qu'on tient à jour à chaque nouveau bloc.



Le coût d'une blockchain est toujours *en plus* et non *à la place* de l'alternative.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-cout-d-une-blockchain>

Le surcoût systématique du recours à une blockchain

- Pour vérifier la validité d'une transaction :
 - il faut entre autre vérifier que l'émetteur dispose bien de la "cryptomonnaie" dépensée.
- Une blockchain est une structure de données particulièrement inefficace :
 - elle liste seulement les *modifications* du système (les transactions),
 - le montant de "cryptomonnaie" disponible sur un compte correspond à ce qui y a déjà été encaissé mais qui n'a pas encore été dépensé.
 - si on ne dispose que du registre, il faut le relire en entier pour chaque vérification.
- En pratique, on est obligé de maintenir une vraie base de données :
 - on a besoin d'accéder à l'état du système directement,
 - on construit l'état consolidé qu'on tient à jour à chaque nouveau bloc.



Le coût d'une blockchain est toujours *en plus* et non *à la place* de l'alternative.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-cout-d-une-blockchain>

Une catastrophe écologique : l'exemple de Bitcoin

- Les blockchains à preuve de travail consomme énormément d'énergie.
 - Le travail doit nécessairement être inutile pour que le modèle de sécurité fonctionne.
 - On parle de PoW (*proof of waste*). Traduction possible : PDG (*preuve de gaspillage*).
- Bitcoin a lui seul consomme annuellement :
 - ~80 Mt de CO₂ ;
 - ~140 TWh d'énergie électrique ;
 - ~30 kt de déchets électroniques ;
 - ~2000 GL d'eau.
- En normalisant ces chiffres pour avoir des données par transaction, on peut comparer avec les transactions VISA par exemple :
 - 1 transaction Bitcoin consomme autant d'énergie que ~560 000 transactions VISA ;
 - 1 transaction Bitcoin émet autant de CO₂ que ~1 million de transaction VISA.

Aller plus loin : <https://digiconomist.net/bitcoin-energy-consumption>

La “tragédie des communs” s'applique aux blockchains

- Il faut permettre à tout moment l'arrivée de nouveaux participants.
 - Sinon, on connaît l'ensemble des participants : plus besoin de blockchain.
 - Les nouveaux participants doivent récupérer tout l'historique du registre.
 - Le registre ne fait que grossir :
 - avec le temps, son stockage et son partage sont de plus en plus coûteux.
 - Soit on est vraiment dans une situation de défiance généralisée et de concurrence :
 - on se retrouve face à une forme de *dilemme du prisonnier*,
 - on est en plein dans un cas de “tragédie des communs”.
 - Soit il faut admettre que toute organisation nécessite altruisme et confiance :
 - cela remet en cause les hypothèses justifiant le recours à une blockchain.
-  Les hypothèses rendant une blockchain nécessaire empêchent sa pérennité.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-probleme-resolu-par-la-blockchain-n-existe-pas>

La “tragédie des communs” s’applique aux blockchains

- Il faut permettre à tout moment l’arrivée de nouveaux participants.
 - Sinon, on connaît l’ensemble des participants : plus besoin de blockchain.
 - Les nouveaux participants doivent récupérer tout l’historique du registre.
 - Le registre ne fait que grossir :
 - avec le temps, son stockage et son partage sont de plus en plus coûteux.
 - Soit on est vraiment dans une situation de défiance généralisée et de concurrence :
 - on se retrouve face à une forme de *dilemme du prisonnier*,
 - on est en plein dans un cas de “*tragédie des communs*”.
 - Soit il faut admettre que toute organisation nécessite altruisme et confiance :
 - cela remet en cause les hypothèses justifiant le recours à une blockchain.
-  Les hypothèses rendant une blockchain nécessaire empêchent sa pérennité.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-probleme-resolu-par-la-blockchain-n-existe-pas>

La "tragédie des communs" s'applique aux blockchains

- Il faut permettre à tout moment l'arrivée de nouveaux participants.
 - Sinon, on connaît l'ensemble des participants : plus besoin de blockchain.
 - Les nouveaux participants doivent récupérer tout l'historique du registre.
- Le registre ne fait que grossir :
 - avec le temps, son stockage et son partage sont de plus en plus coûteux.
- Soit on est vraiment dans une situation de défiance généralisée et de concurrence :
 - on se retrouve face à une forme de *dilemme du prisonnier*,
 - on est en plein dans un cas de "*tragédie des communs*".
- Soit il faut admettre que toute organisation nécessite altruisme et confiance :
 - cela remet en cause les hypothèses justifiant le recours à une blockchain.



Les hypothèses rendant une blockchain nécessaire empêchent sa pérennité.

Aller plus loin : <https://p4bl0.net/post/2022/02/Le-probleme-resolu-par-la-blockchain-n-existe-pas>

L'individualisme forcené de la décentralisation absolue

- La *cryptographie asymétrique* vraiment décentralisée ne peut pas être conviviale :
 - les notions en jeu ne sont pas simples à appréhender,
 - il est notoirement difficile de conserver ses clefs privées de façon sécurisée et fiable.
- Cela pose également des problèmes techniques et politiques :
 - les erreurs (fuites, fausses manipulations, pertes, etc.) sont définitives et irréparables,
 - impossibilité d'imposer des décisions collectives.
- L'alternative à la décentralisation absolue n'est pas forcément la centralisation :
 - en terme d'architecture réseau, la *fédération* est une approche pragmatique,
 - même pour les blockchains en pratique : *exchanges* centralisés, *hosted wallets*,
 - mais implique la possibilité de *confiance* : sortie du cadre justifiant les blockchains.



En plus d'être politiquement indésirable, la décentralisation totale est un mythe.

Aller plus loin : <https://p4bl0.net/post/2023/06/ultra-individualisme-decentralisation-totale>

L'individualisme forcené de la décentralisation absolue

- La *cryptographie asymétrique* vraiment décentralisée ne peut pas être conviviale :
 - les notions en jeu ne sont pas simples à appréhender,
 - il est notoirement difficile de conserver ses clefs privées de façon sécurisée et fiable.
 - Cela pose également des problèmes techniques et politiques :
 - les erreurs (fuites, fausses manipulations, pertes, etc.) sont définitives et irréparables,
 - impossibilité d'imposer des décisions collectives.
 - L'alternative à la décentralisation absolue n'est pas forcément la centralisation :
 - en terme d'architecture réseau, la *fédération* est une approche pragmatique,
 - même pour les blockchains en pratique : *exchanges* centralisés, *hosted wallets*,
 - mais implique la possibilité de *confiance* : sortie du cadre justifiant les blockchains.
-  En plus d'être politiquement indésirable, la décentralisation totale est un mythe.

Aller plus loin : <https://p4bl0.net/post/2023/06/ultra-individualisme-decentralisation-totale>

L'individualisme forcené de la décentralisation absolue

- La *cryptographie asymétrique* vraiment décentralisée ne peut pas être conviviale :
 - les notions en jeu ne sont pas simples à appréhender,
 - il est notoirement difficile de conserver ses clefs privées de façon sécurisée et fiable.
- Cela pose également des problèmes techniques et politiques :
 - les erreurs (fuites, fausses manipulations, pertes, etc.) sont définitives et irréparables,
 - impossibilité d'imposer des décisions collectives.
- L'alternative à la décentralisation absolue n'est pas forcément la centralisation :
 - en terme d'architecture réseau, la *fédération* est une approche pragmatique,
 - même pour les blockchains en pratique : *exchanges* centralisés, *hosted wallets*,
 - mais implique la possibilité de *confiance* : sortie du cadre justifiant les blockchains.



En plus d'être politiquement indésirable, la décentralisation totale est un mythe.

Aller plus loin : <https://p4bl0.net/post/2023/06/ultra-individualisme-decentralisation-totale>

La performativité de l'écriture, source de confusion

- Dans le cas des “cryptomonnaies”, on a affaire à de l'*écriture performative* :
 - les transactions ne sont qu'un jeu d'écriture, sans aucun lien avec le monde réel,
 - les choses sont vraies *parce qu'elles* sont écrites, du fait même d'être écrites.
- Ça ne fonctionne pour rien d'autre :
 - certification : il faut une autorité de certification,
 - contrat : il faut une tierce partie pour le faire appliquer,
 - traçabilité : on doit connaître les acteurs,
 - vote électronique : je... hein ? 😐
- Bref...

La seule vérité garantie par l'écriture d'une information sur une blockchain est que cette information est écrite sur cette blockchain.

Aller plus loin : <https://p4bl0.net/post/2021/06/La-vérité-sur-la-blockchain>

La performativité de l'écriture, source de confusion

- Dans le cas des “cryptomonnaies”, on a affaire à de l'*écriture performative* :
 - les transactions ne sont qu'un jeu d'écriture, sans aucun lien avec le monde réel,
 - les choses sont vraies *parce qu'elles* sont écrites, du fait même d'être écrites.
- Ça ne fonctionne pour rien d'autre :
 - certification : il faut une autorité de certification,
 - contrat : il faut une tierce partie pour le faire appliquer,
 - traçabilité : on doit connaître les acteurs,
 - vote électronique : je... hein ? 😐
- Bref...

La seule vérité garantie par l'écriture d'une information sur une blockchain est que cette information est écrite sur cette blockchain.

Aller plus loin : <https://p4bl0.net/post/2021/06/La-vérité-sur-la-blockchain>

La performativité de l'écriture, source de confusion

- Dans le cas des “cryptomonnaies”, on a affaire à de l'*écriture performative* :
 - les transactions ne sont qu'un jeu d'écriture, sans aucun lien avec le monde réel,
 - les choses sont vraies *parce qu'elles* sont écrites, du fait même d'être écrites.
- Ça ne fonctionne pour rien d'autre :
 - certification : il faut une autorité de certification,
 - contrat : il faut une tierce partie pour le faire appliquer,
 - traçabilité : on doit connaître les acteurs,
 - vote électronique : je... hein ? 😐
- Bref...

La seule vérité garantie par l'écriture d'une information sur une blockchain est que cette information est écrite sur cette blockchain.

Aller plus loin : <https://p4bl0.net/post/2021/06/La-vérité-sur-la-blockchain>

Conclusions



Des blockchains partout par pur effet de mode : technosolutionnisme + confusion.

- “J’ai un problème à résoudre.” vs “J’ai une blockchain, quel problème vais-je résoudre ?”.
- Incompréhension de la nature *performative* de l’écriture pour les “cryptomonnaies”.



Une blockchain n’est la solution qu’à son propre problème.

- Seul usage valide (écriture performative) d’une blockchain : sa “cryptomonnaie”.
- Sa “cryptomonnaie” est nécessaire à son fonctionnement.



Une blockchain est une technologie *non neutre*, d’idéologie libertarienne.

- Présuppose une défiance généralisée et des comportements ultra-individualistes.
- Usage concret principal : véhicule de propagation et normalisation de cette idéologie.
 - Individualisme, métallisme, non régulation, refus de contribuer au bien commun, etc.
 - Déplacement de la confiance des personnes et organisations vers la technologie (sans en voir ni admettre les limites).



Des blockchains partout par pur effet de mode : technosolutionnisme + confusion.

- “J’ai un problème à résoudre.” vs “J’ai une blockchain, quel problème vais-je résoudre ?”.
- Incompréhension de la nature *performative* de l’écriture pour les “cryptomonnaies”.



Une blockchain n’est la solution qu’à son propre problème.

- Seul usage valide (écriture performative) d’une blockchain : sa “cryptomonnaie”.
- Sa “cryptomonnaie” est nécessaire à son fonctionnement.



Une blockchain est une technologie *non neutre*, d’idéologie libertarienne.

- Présuppose une défiance généralisée et des comportements ultra-individualistes.
- Usage concret principal : véhicule de propagation et normalisation de cette idéologie.
 - Individualisme, métallisme, non régulation, refus de contribuer au bien commun, etc.
 - Déplacement de la confiance des personnes et organisations vers la technologie (sans en voir ni admettre les limites).



Des blockchains partout par pur effet de mode : technosolutionnisme + confusion.

- “J’ai un problème à résoudre.” vs “J’ai une blockchain, quel problème vais-je résoudre ?”.
- Incompréhension de la nature *performative* de l’écriture pour les “cryptomonnaies”.



Une blockchain n’est la solution qu’à son propre problème.

- Seul usage valide (écriture performative) d’une blockchain : sa “cryptomonnaie”.
- Sa “cryptomonnaie” est nécessaire à son fonctionnement.



Une blockchain est une technologie *non neutre*, d’idéologie libertarienne.

- Présuppose une défiance généralisée et des comportements ultra-individualistes.
- Usage concret principal : véhicule de propagation et normalisation de cette idéologie.
 - Individualisme, métallisme, non régulation, refus de contribuer au bien commun, etc.
 - Déplacement de la confiance des personnes et organisations vers la technologie (sans en voir ni admettre les limites).

Perspectives (de lutte)

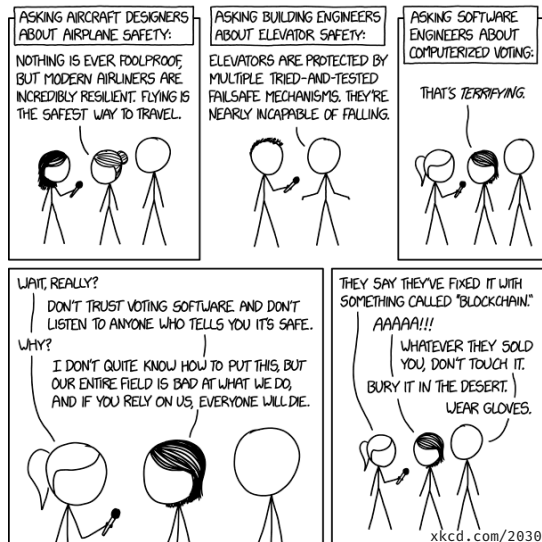
🔍 Expliquer le fonctionnement technique et les limites des blockchains.

⚠️ Alerter sur leur nature lourdement idéologique et propagandiste.

👊 Attaquer sans relâche les projets absurdes voire dangereux.

💡 Informer sur les alternatives.

→ pablockchain.fr



Blockchain : quèsaco ?

Définition

- Usage

Fonctionnement d'une blockchain

Immuabilité du registre

- Condensat cryptographique
- L'intégrité d'une blockchain
- Collision

Distribution du registre

Unicité du registre

- Consensus distribué
- Tirage au sort distribué non contestable ("minage")

Nécessité fonctionnelle de la "cryptomonnaie"

- Nécessité de donner de la valeur à la "cryptomonnaie"

Fonctionnement d'une "cryptomonnaie"

- Anatomie d'une transaction
- Validité d'une transaction
- Cryptographie asymétrique et signature
- Incitation et récompense

Points vocabulaire

Propagande par les mots

- Décentralisation ?
- Consensus ?
- Minage ?
- Cash ? Pair-à-pair ? Portefeuille ?
- "Cryptomonnaie" ?

Blockchain : quelles limites ?

Un serpent qui se mord la queue

- La blockchain comme solution à son propre problème

Une structure de données inefficace

- Le surcoût systématique du recours à une blockchain

Un problème qui n'existe pas

- La "tragédie des communs" s'applique aux blockchains

Un élitisme aveuglé

- L'individualisme forcené de la décentralisation absolue

La vérité sur la blockchain

- La performativité de l'écriture, source de confusion

Conclusions

Perspectives (de lutte)



A. Back.

Hash cash postage implementation.

Cypherpunks mailing-list, 1997.

<http://hashcash.org/>.



D. Chaum.

Blind Signatures for Untraceable Payments.

Advances in Cryptology: Proceedings of Crypto '82, 1982.

https://sci-hub.se/10.1007/978-1-4757-0602-4_18.



W. Diffie and M. Hellman.

New directions in cryptography.

IEEE Transactions on Information Theory 22-6, 1976.

<https://cr.yp.to/bib/1976/diffie.pdf>.



C. Dwork and M. Naor.

Pricing via processing or combatting junk mail.

Advances in Cryptology: Proceedings of Crypto '92, 1992.

https://link.springer.com/content/pdf/10.1007/3-540-48071-4_10.pdf.



R. C. Merkle.

A digital signature based on a conventional encryption function.

Advances in Cryptology: Proceedings of Crypto '87, 1987.

https://link.springer.com/content/pdf/10.1007/3-540-48184-2_32.pdf.



S. Nakamoto.

Bitcoin: A Peer-to-Peer Electronic Cash System.

Rapport, 2008.

<https://bitcoin.org/bitcoin.pdf>.



NIST.

Secure Hash Standard.

Federal Information Processing Standards Publication 180-2, 2002.

<https://csrc.nist.gov/csrc/media/publications/fips/180/2/archive/2002-08-01/documents/fips180-2.pdf>.



R. L. Rivest, A. Shamir, and L. Adleman.

A method for obtaining digital signatures and public-key cryptosystems.

Communications of the ACM 21-2, 1978.

<https://dl.acm.org/doi/pdf/10.1145/359340.359342>.