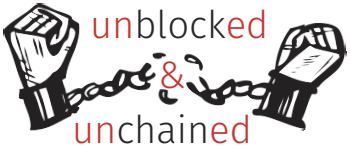
Université Paris 8 / LIASD & GÉODE
équipe de recherche PASTIS

univ-paris8.fr
informatique.up8.edu/pastis

Promesses et (dés)illusions :
une introduction technocritique aux blockchains



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

Article en libre accès : Terminal vol. 136
Pour allez plus loin : pablockchain.fr

Contexte

- Constat : les blockchains sont partout, et sont supposées changer le monde.
  - ⇒ Quels enjeux de sécurité, juridiques, économiques, écologiques, démocratiques, … ?
- Commençons par le commencement :
  - Qu'est-ce que c'est concrètement une blockchain ?
  - Comment ça fonctionne ?
  - Quelles sont les limites de cette technologie ?
  - Ses nombreux usages sont-ils seulement justifiés ?
  - Quid des promesses de sécurité, de décentralisation, et de désintermédiation systématiquement mises en avant ?

- Constat : les blockchains sont partout, et sont supposées changer le monde.
  - ⇒ Quels enjeux de sécurité, juridiques, économiques, écologiques, démocratiques, … ?
- Commençons par le commencement :
  - Qu'est-ce que c'est concrètement une blockchain ?
  - Comment ça fonctionne ?
  - Quelles sont les limites de cette technologie ?
  - Ses nombreux usages sont-ils seulement justifiés ?
  - Quid des promesses de sécurité, de décentralisation, et de désintermédiation systématiquement mises en avant ?

Qu’est-ce qu’une blockchain ?

— Une blockchain est un registre *distribué* et *immuable* dans lequel sont écrites des informations qui font *consensus*.

— Sachant que :

  – On ne connaît pas l’ensemble des participants potentiels.

  – On n’accorde aucune confiance aux participants (ni à personne d’autre).

— Une blockchain est un registre *distribué* et *immuable* dans lequel sont écrites des informations qui font *consensus*.

— Sachant que :

  – On ne connaît pas l'ensemble des participants potentiels.

  – On n'accorde aucune confiance aux participants (ni à personne d'autre).

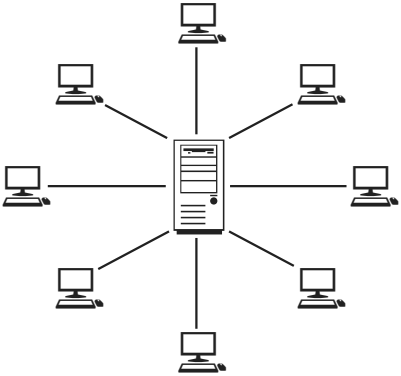
Un registre *distribué*

— Le registre est *distribué* au sens où chaque pair en possède une copie complète.
  – Cela permet à chacun de consulter et partager le registre sans dépendre de quiconque.
  – Mais attention à ne pas confondre “distribué” et “décentralisé” (ou “pair-à-pair”).

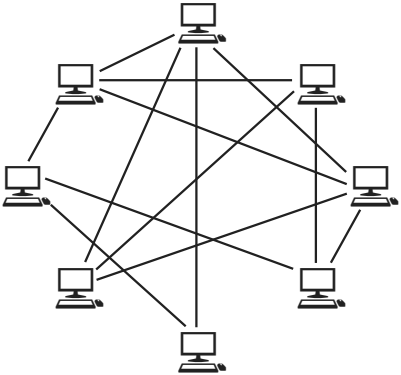
Un registre *distribué*

— Le registre est *distribué* au sens où chaque pair en possède une copie complète.
  - Cela permet à chacun de consulter et partager le registre sans dépendre de quiconque.
  - Mais attention à ne pas confondre “distribué” et “décentralisé” (ou “pair-à-pair”).

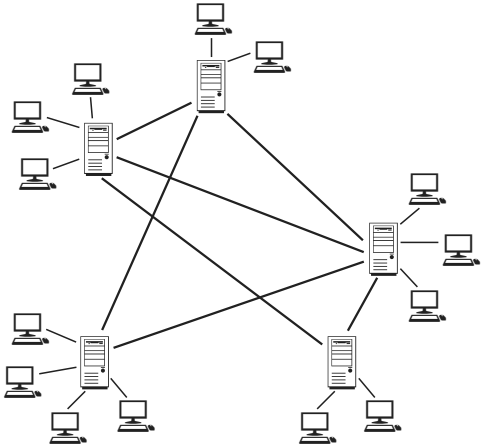
Réseaux centralisés



Réseaux décentralisés



Réseaux fédérés



- Le réseau sous-jacent, où se passent les échanges entre pairs, est décentralisé.
- Le registre est distribué : il se veut identique partout, comme s'il n'y en avait qu'un.
- *Les transactions sont centralisées* sur ce registre.
  - Les transactions sont ajoutées par *bloc* à la suite de la chaîne.

- Le réseau sous-jacent, où se passent les échanges entre pairs, est décentralisé.
- Le registre est distribué : il se veut identique partout, comme s'il n'y en avait qu'un.
- *Les transactions sont centralisées* sur ce registre.
  - Les transactions sont ajoutées par *bloc* à la suite de la chaîne.

- Le réseau sous-jacent, où se passent les échanges entre pairs, est décentralisé.
- Le registre est distribué : il se veut identique partout, comme s'il n'y en avait qu'un.
- *Les transactions sont centralisées* sur ce registre.
  - Les transactions sont ajoutées par *bloc* à la suite de la chaîne.

- Le réseau sous-jacent, où se passent les échanges entre pairs, est décentralisé.
- Le registre est distribué : il se veut identique partout, comme s'il n'y en avait qu'un.

Les transactions sont centralisées sur ce registre.

  - Les transactions sont ajoutées par *bloc* à la suite de la chaîne.

Un registre *immuable*

- Le registre se veut *immuable* : ce qu'on y ajoute ne peut être modifié ni supprimé.
  - L'idée est de pouvoir faire confiance "pour toujours" à ce qu'on y lit.
  - Une fois un bloc de transactions ajouté à la chaîne, il ne doit plus être modifiable.
- Le registre étant un fichier, son immuabilité n'existe que virtuellement.
  - En permettant de *vérifier son intégrité*, on rend impossible sa modification *discrète*.
  - La vérification d'intégrité est permise par l'utilisation de *condensat cryptographique*.

Un registre *immuable*

— Le registre se veut *immuable* : ce qu'on y ajoute ne peut être modifié ni supprimé.
  – L'idée est de pouvoir faire confiance "pour toujours" à ce qu'on y lit.
  – Une fois un bloc de transactions ajouté à la chaîne, il ne doit plus être modifiable.

— Le registre étant un fichier, son immuabilité n'existe que virtuellement.
  – En permettant de *vérifier son intégrité*, on rend impossible sa modification *discrète*.
  – La vérification d'intégrité est permise par l'utilisation de *condensat cryptographique*.

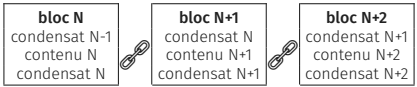
Condensat cryptographique

— 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] :
    - "Terminal" : `e0926fdac700b09497b5f0218ea3dd54fa13c0bdeaee6caa7b85e50b852aa05f`,
    - "terminal" : `4e686af7bdcc5ae005a247624fd8c7283257c2514f6b3ad2ff5d4cb6d95196e6`,
    - "T" est encodé par `01`**`1`**`10100` tandis que "t" l'est par `01`**`0`**`10100`.

— Le principe est de pouvoir *identifier* une donnée.

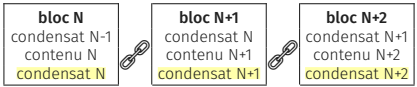
L'intégrité d'une blockchain

- 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.



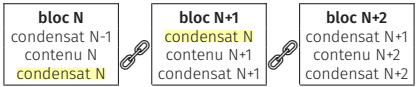
L'intégrité d'une blockchain

- 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.



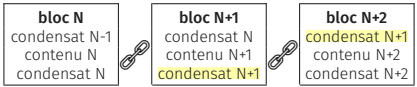
L'intégrité d'une blockchain

- 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.

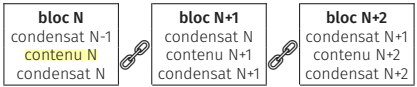


L'intégrité d'une blockchain

- 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.

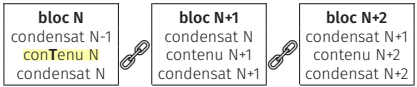


- Dans une blockchain, chaque bloc est identifié par son condensat.
  - On peut facilement vérifier son intégrité en recalculant son condensat.
  - Un ordinateur personnel peut calculer plusieurs millions de condensats par seconde.
- L'intégrité de la blockchain est vérifiable grâce à la structure de 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.



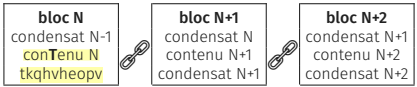
L'intégrité d'une blockchain

- 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.



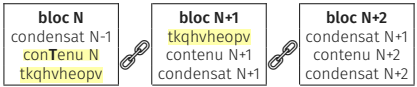
L'intégrité d'une blockchain

- 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.



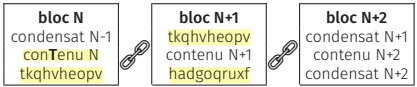
L'intégrité d'une blockchain

- 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.



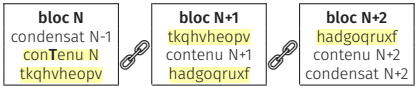
L'intégrité d'une blockchain

- 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.



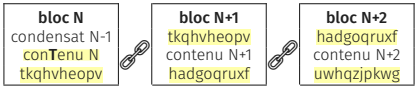
L'intégrité d'une blockchain

- 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.



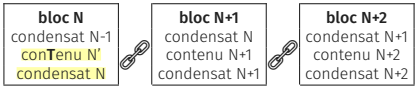
L'intégrité d'une blockchain

- 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.

Lors de l'arrivée d'un nouveau participant :
  - Nécessité de récupérer l'historique de la blockchain.
  - Quelle confiance en la ou les sources de partage ?
  - Pour ces sources, quid de la constante augmentation du coût de ce partage nécessaire ?

L'immuabilité n'est *pas* produite par la preuve de travail (ou d'enjeu).

L'immuabilité d'une blockchain

— L'immuabilité d'une blockchain est donc produite par :
  - la possibilité de vérifier son intégrité ;
  - sa distribution, pour avoir des points de comparaison.

Lors de l'arrivée d'un nouveau participant :
  - Nécessité de récupérer l'historique de la blockchain.
  - Quelle confiance en la ou les sources de partage ?
  - Pour ces sources, quid de la constante augmentation du coût de ce partage nécessaire ?

L'immuabilité n'est *pas* produite par la preuve de travail (ou d'enjeu).

L'immuabilité d'une blockchain

— L'immuabilité d'une blockchain est donc produite par :
  - la possibilité de vérifier son intégrité ;
  - sa distribution, pour avoir des points de comparaison.

Lors de l'arrivée d'un nouveau participant :
  - Nécessité de récupérer l'historique de la blockchain.
  - Quelle confiance en la ou les sources de partage ?
  - Pour ces sources, quid de la constante augmentation du coût de ce partage nécessaire ?

L'immuabilité n'est *pas* produite par la preuve de travail (ou d'enjeu).

— L'immuabilité d'une blockchain est donc produite par :
  - la possibilité de vérifier son intégrité ;
  - sa distribution, pour avoir des points de comparaison.

Lors de l'arrivée d'un nouveau participant :
  - Nécessité de récupérer l'historique de la blockchain.
  - Quelle confiance en la ou les sources de partage ?
  - Pour ces sources, quid de la constante augmentation du coût de ce partage nécessaire ?

L'immuabilité n'est *pas* produite par la preuve de travail (ou d'enjeu).

Des informations qui font *consensus*

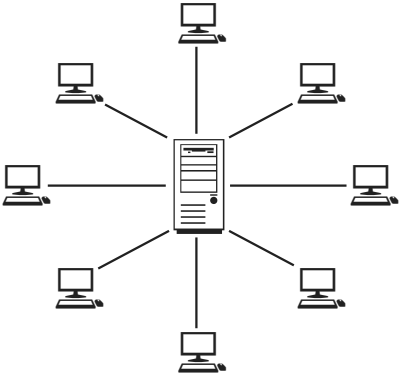
- Dans le langage courant, le *consensus* est :
  - « l'accord et le consentement du plus grand nombre, de l'opinion publique » (source : Larousse en ligne).
- La notion technique est très différente de celle du langage courant.

Des informations qui font *consensus*

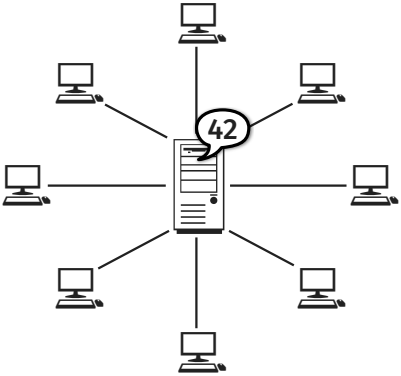
- Dans le langage courant, le *consensus* est :
  - « l'accord et le consentement du plus grand nombre, de l'opinion publique » (source : Larousse en ligne).
- La notion technique est très différente de celle du langage courant.

— En informatique, le *problème du consensus* consiste à mettre d'accord un ensemble de machines sur une valeur unique.
  – Il n'est plus question de politique / moral / consentement.
— Le consensus est atteint quand toutes les machines sont d'accord sur une valeur.
  – Cette valeur n'est pas forcément la proposition majoritaire.

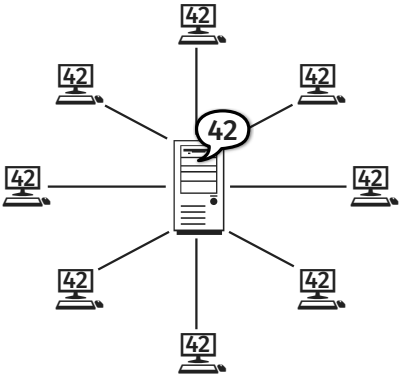
Le problème du consensus distribué



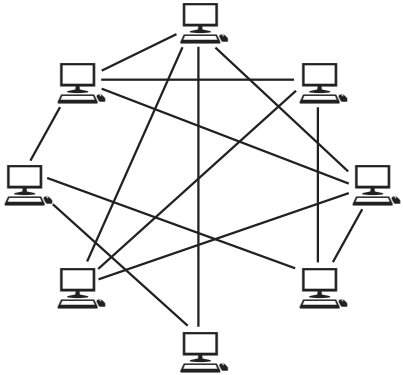
Le problème du consensus distribué



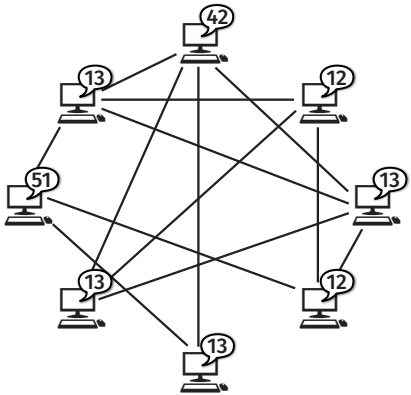
Le problème du consensus distribué



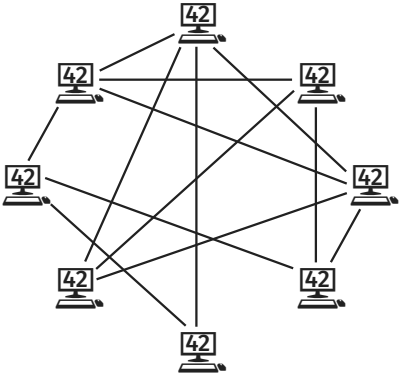
Le problème du consensus distribué



Le problème du consensus distribué



Le problème du consensus distribué



Le cas d'une blockchain

- Dans le cas d'une blockchain, on veut se mettre d'accord sur le prochain bloc.
- Sans faire confiance à personne.
  - Les participants ont de toutes façons des intérêts divergents.
  - Chacun veut prioriser ses transactions alors que la taille des blocs est limitée.
  - ⇒ Impossible de satisfaire tout le monde.
- Sans connaître l'ensemble des participants potentiels.
  - ⇒ Impossible de faire "chacun son tour".

⇒ Il est nécessaire de recourir à un *tirage au sort non contestable*.

- Donc sans que celui-ci soit fait par un participant ou une entité extérieure (cela casserait le modèle de défiance généralisée et d'absence d'autorité centrale).

Le cas d'une blockchain

- Dans le cas d'une blockchain, on veut se mettre d'accord sur le prochain bloc.
- Sans faire confiance à personne.
  - Les participants ont de toutes façons des intérêts divergents.
  - Chacun veut prioriser ses transactions alors que la taille des blocs est limitée.

  ⇒ Impossible de satisfaire tout le monde.
- Sans connaître l'ensemble des participants potentiels.

  ⇒ Impossible de faire "chacun son tour".

⇒ Il est nécessaire de recourir à un *tirage au sort non contestable*.

- Donc sans que celui-ci soit fait par un participant ou une entité extérieure (cela casserait le modèle de défiance généralisée et d'absence d'autorité centrale).

Le cas d'une blockchain

- Dans le cas d'une blockchain, on veut se mettre d'accord sur le prochain bloc.
- Sans faire confiance à personne.
  - Les participants ont de toutes façons des intérêts divergents.
  - Chacun veut prioriser ses transactions alors que la taille des blocs est limitée.

  ⇒ Impossible de satisfaire tout le monde.
- Sans connaître l'ensemble des participants potentiels.

  ⇒ Impossible de faire "chacun son tour".

⇒ Il est nécessaire de recourir à un *tirage au sort non contestable*.

- Donc sans que celui-ci soit fait par un participant ou une entité extérieure (cela casserait le modèle de défiance généralisée et d'absence d'autorité centrale).

Le cas d'une blockchain

- Dans le cas d'une blockchain, on veut se mettre d'accord sur le prochain bloc.
- Sans faire confiance à personne.
  - Les participants ont de toutes façons des intérêts divergents.
  - Chacun veut prioriser ses transactions alors que la taille des blocs est limitée.

  ⇒ Impossible de satisfaire tout le monde.
- Sans connaître l'ensemble des participants potentiels.

  ⇒ Impossible de faire "chacun son tour".

⇒ Il est nécessaire de recourir à un *tirage au sort non contestable*.

- Donc sans que celui-ci soit fait par un participant ou une entité extérieure (cela casserait le modèle de défiance généralisée et d'absence d'autorité centrale).

- La *preuve de travail* est une invention du début des années 90 [DN92].
- Il s'agissait originalement de lutter contre le spam :
  - demander pour chaque mail de faire un minimum de calculs inutiles ;
  - raisonnable pour un usage normal du mail, trop coûteux pour les spammeurs.
- Première mise en œuvre avec Hashcash [Bac97].
  - Calcul de condensats cryptographiques en boucle jusqu'à trouver une *collision partielle*.
  - Les 20 premiers bits du condensat devaient être à zéro.

— Pour trouver une *collision partielle*, on utilise un *nonce*.
  - Le mot "*nonce*" est la contraction de "*number*" et "*once*".
  - Ce nombre est modifié à chaque nouveau calcul (par exemple, incrémenté).
  - En boucle jusqu'à trouver la collision partielle qui nous intéresse.
  - Le *nonce* obtenu à ce moment là est notre preuve de travail.
    - Il suffit de calculer un seul condensat, avec ce *nonce*, pour vérifier le travail.

Minage

- Dans les blockchains, la recherche d'une collision partielle s'appelle le *minage*.
- La difficulté *moyenne* du minage est adaptable avec la taille de la collision attendue.
- Pour un cas précis, impossible de prédire le nombre de calculs nécessaires.
  - Exemple pour une collision sur 20 bits :
    - De "Terminal0" il faut aller jusqu'à "Terminal1277191" : `0000061c3401962f21905bec7299328d495d1b10846ac6cf699be03e96497a8f`.
    - De "terminal0" il faut aller jusqu'à "terminal296762" (~1 million de calculs en moins) : `000007085de78efeec7e3e9f120ca34fadfb782f5b386fa63989d5ddbd86269d`.

Minage

- Dans les blockchains, la recherche d'une collision partielle s'appelle le *minage*.
- La difficulté *moyenne* du minage est adaptable avec la taille de la collision attendue.
- Pour un cas précis, impossible de prédire le nombre de calculs nécessaires.
  - Exemple pour une collision sur 20 bits :
    - De "Terminal0" il faut aller jusqu'à "Terminal1277191" :
      `0000061c3401962f21905bec7299328d495d1b10846ac6cf699be03e96497a8f`.
    - De "terminal0" il faut aller jusqu'à "terminal296762" (~1 million de calculs en moins) :
      `000007085de78efeec7e3e9f120ca34fadfb782f5b386fa63989d5ddbd86269d`.

Minage

- Dans les blockchains, la recherche d'une collision partielle s'appelle le *minage*.
- La difficulté *moyenne* du minage est adaptable avec la taille de la collision attendue.
- Pour un cas précis, impossible de prédire le nombre de calculs nécessaires.
  - Exemple pour une collision sur 20 bits :
    - De "Terminal0" il faut aller jusqu'à "Terminal1277191" : `0000061c3401962f21905bec7299328d495d1b10846ac6cf699be03e96497a8f`.
    - De "terminal0" il faut aller jusqu'à "terminal296762" (~1 million de calculs en moins) : `000007085de78efeec7e3e9f120ca34fadfb782f5b386fa63989d5ddbd86269d`.

Tirage au sort non contestable

- Dans le cas du minage d'un nouveau bloc d'une blockchain :
  - l'identifiant du dernier bloc est le même pour tout le monde,
  - mais pas forcément les données que l'on souhaite ajouter sur la chaîne ;
  - ⇒ impossible (en théorie) de prédire qui trouvera en premier un nouveau bloc valide.
- Le premier qui trouve partage son bloc sur le réseau :
  - le bloc sera vérifié par les autres mineurs, puis ajouté à la chaîne,
  - les autres mineurs abandonnent leur calcul en cours et repartent du nouveau bloc.
  - Si plusieurs blocs valides sont trouvés ~ en même temps :
    - chaque mineur choisit de quel bloc il repart,
    - à terme la chaîne qui fait foi est la plus longue.
  - Un bloc est valide quand toutes les transactions qu'il contient le sont.
    - On reviendra sur la validité des transactions en parlant des « cryptomonnaies ».

Tirage au sort non contestable

- Dans le cas du minage d'un nouveau bloc d'une blockchain :
  - l'identifiant du dernier bloc est le même pour tout le monde,
  - mais pas forcément les données que l'on souhaite ajouter sur la chaîne ;
  - ⇒ impossible (en théorie) de prédire qui trouvera en premier un nouveau bloc valide.
- Le premier qui trouve partage son bloc sur le réseau :
  - le bloc sera vérifié par les autres mineurs, puis ajouté à la chaîne,
  - les autres mineurs abandonnent leur calcul en cours et repartent du nouveau bloc.
  - Si plusieurs blocs valides sont trouvés ~ en même temps :
    - chaque mineur choisit de quel bloc il repart,
    - à terme la chaîne qui fait foi est la plus longue.
  - Un bloc est valide quand toutes les transactions qu'il contient le sont.
    - On reviendra sur la validité des transactions en parlant des « cryptomonnaies ».

Tirage au sort non contestable

- Dans le cas du minage d'un nouveau bloc d'une blockchain :
  - l'identifiant du dernier bloc est le même pour tout le monde,
  - mais pas forcément les données que l'on souhaite ajouter sur la chaîne ;
  - ⇒ impossible (en théorie) de prédire qui trouvera en premier un nouveau bloc valide.
- Le premier qui trouve partage son bloc sur le réseau :
  - le bloc sera vérifié par les autres mineurs, puis ajouté à la chaîne,
  - les autres mineurs abandonnent leur calcul en cours et repartent du nouveau bloc.
  - Si plusieurs blocs valides sont trouvés ~ en même temps :
    - chaque mineur choisit de quel bloc il repart,
    - à terme la chaîne qui fait foi est la plus longue.
  - Un bloc est valide quand toutes les transactions qu'il contient le sont.
    - On reviendra sur la validité des transactions en parlant des « cryptomonnaies ».

Tirage au sort non contestable

— Dans le cas du minage d'un nouveau bloc d'une blockchain :
  – l'identifiant du dernier bloc est le même pour tout le monde,
  – mais pas forcément les données que l'on souhaite ajouter sur la chaîne ;
  ⇒ impossible (en théorie) de prédire qui trouvera en premier un nouveau bloc valide.

— Le premier qui trouve partage son bloc sur le réseau :
  – le bloc sera vérifié par les autres mineurs, puis ajouté à la chaîne,
  – les autres mineurs abandonnent leur calcul en cours et repartent du nouveau bloc.
  – Si plusieurs blocs valides sont trouvés ~ en même temps :
    - chaque mineur choisit de quel bloc il repart,
    - à terme la chaîne qui fait foi est la plus longue.
  – Un bloc est valide quand toutes les transactions qu'il contient le sont.
    - On reviendra sur la validité des transactions en parlant des « cryptomonnaies ».

Nécessité d'une incitation

- Sans récompense à la clef, le coût de ces calculs serait prohibitif.
  - La taille de la collision partielle attendue pour Bitcoin en ce moment est de 76 bits.
  - Sur un ordinateur personnel ça se compterait en milliards d'années de calcul.
  - ⇒ Coût énergétique élevé, matériel spécifique nécessaire.
- Chacun tente de s'auto-attribuer la récompense.
  - Les mineurs sont en compétition les uns contre les autres.
  - ⇒ Le "consensus" se fait sur une valeur voulue *par un seul contre tous les autres*.
- La récompense doit provenir intrinsèquement de la blockchain elle-même.
  - Même raison que pour le tirage au sors : aucun tiers de confiance, même extérieur.
  - ⇒ Une blockchain ne peut pas fonctionner sans sa « cryptomonnaie ».

Nécessité d'une incitation

- Sans récompense à la clef, le coût de ces calculs serait prohibitif.
  - La taille de la collision partielle attendue pour Bitcoin en ce moment est de 76 bits.
  - Sur un ordinateur personnel ça se compterait en milliards d'années de calcul.
  - ⇒ Coût énergétique élevé, matériel spécifique nécessaire.
- Chacun tente de s'auto-attribuer la récompense.
  - Les mineurs sont en compétition les uns contre les autres.
  - ⇒ Le "consensus" se fait sur une valeur voulue *par un seul contre tous les autres*.
- La récompense doit provenir intrinsèquement de la blockchain elle-même.
  - Même raison que pour le tirage au sors : aucun tiers de confiance, même extérieur.
  - ⇒ Une blockchain ne peut pas fonctionner sans sa « cryptomonnaie ».

Nécessité d'une incitation

- Sans récompense à la clef, le coût de ces calculs serait prohibitif.
  - La taille de la collision partielle attendue pour Bitcoin en ce moment est de 76 bits.
  - Sur un ordinateur personnel ça se compterait en milliards d'années de calcul.
  - ⇒ Coût énergétique élevé, matériel spécifique nécessaire.
- Chacun tente de s'auto-attribuer la récompense.
  - Les mineurs sont en compétition les uns contre les autres.
  - ⇒ Le "consensus" se fait sur une valeur voulue *par un seul contre tous les autres*.
- La récompense doit provenir intrinsèquement de la blockchain elle-même.
  - Même raison que pour le tirage au sors : aucun tiers de confiance, même extérieur.
  - ⇒ Une blockchain ne peut pas fonctionner sans sa « cryptomonnaie ».

- Il peut exister d'autres mécanismes de consensus, typiquement, la *preuve d'enjeu*.
- Au lieu de calculs inutiles, le tirage au sort de base sur un enjeu.
  - Enjeu : immobiliser un montant de « cryptomonnaie ».
  - Plus son enjeu est grand et long dans le temps, moins on aurait intérêt à tricher.
  - Ces informations étant publiques, le résultat du tirage ne doit pas surprendre.
- Moins sûre que la preuve de travail, en particulier contre la *double-dépense*.
  - Facile de poursuivre plusieurs versions alternatives de la chaîne à la fois.
- Tout de même un inconvénient de devoir mettre en jeu sa « cryptomonnaie ».
  - Le montant mis en jeu est immobilisé.
  - Une récompense est donc nécessaire également.

- Il peut exister d'autres mécanismes de consensus, typiquement, la *preuve d'enjeu*.
- Au lieu de calculs inutiles, le tirage au sort de base sur un enjeu.
  - Enjeu : immobiliser un montant de « cryptomonnaie ».
  - Plus son enjeu est grand et long dans le temps, moins on aurait intérêt à tricher.
  - Ces informations étant publiques, le résultat du tirage ne doit pas surprendre.
- Moins sûre que la preuve de travail, en particulier contre la *double-dépense*.
  - Facile de poursuivre plusieurs versions alternatives de la chaîne à la fois.
- Tout de même un inconvénient de devoir mettre en jeu sa « cryptomonnaie ».
  - Le montant mis en jeu est immobilisé.
  - Une récompense est donc nécessaire également.

- Il peut exister d'autres mécanismes de consensus, typiquement, la *preuve d'enjeu*.
- Au lieu de calculs inutiles, le tirage au sort de base sur un enjeu.
  - Enjeu : immobiliser un montant de « cryptomonnaie ».
  - Plus son enjeu est grand et long dans le temps, moins on aurait intérêt à tricher.
  - Ces informations étant publiques, le résultat du tirage ne doit pas surprendre.
- Moins sûre que la preuve de travail, en particulier contre la *double-dépense*.
  - Facile de poursuivre plusieurs versions alternatives de la chaîne à la fois.
- Tout de même un inconvénient de devoir mettre en jeu sa « cryptomonnaie ».
  - Le montant mis en jeu est immobilisé.
  - Une récompense est donc nécessaire également.

- Il peut exister d'autres mécanismes de consensus, typiquement, la *preuve d'enjeu*.
- Au lieu de calculs inutiles, le tirage au sort de base sur un enjeu.
  - Enjeu : immobiliser un montant de « cryptomonnaie ».
  - Plus son enjeu est grand et long dans le temps, moins on aurait intérêt à tricher.
  - Ces informations étant publiques, le résultat du tirage ne doit pas surprendre.
- Moins sûre que la preuve de travail, en particulier contre la *double-dépense*.
  - Facile de poursuivre plusieurs versions alternatives de la chaîne à la fois.
- Tout de même un inconvénient de devoir mettre en jeu sa « cryptomonnaie ».
  - Le montant mis en jeu est immobilisé.
  - Une récompense est donc nécessaire également.

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

À retenir

- Réseau sous-jacent décentralisé, transactions centralisées par le registre distribué.
  - On a omit les aspects réseaux techniques (fonctionnement du protocole pair-à-pair) :
    - les usagers peuvent envoyer des transactions sur le réseau, et
    - espérer qu'elles seront ajoutées par des mineurs dans un bloc ;
    - les blocs valides sont transmis aux mineurs pour vérification, puis
    - les blocs vérifiés sont transmis à l'ensemble des participants pour ajout à la chaîne.
  - Remise en question de la pérennité d'une blockchain :
    - leur modèle de sécurité (défiance généralisée) favorise la "*tragédie des communs*".
- Immuabilité et distribution indépendants du mécanisme de consensus.
  - Ce ne sont pas des propriétés uniques des blockchains.
  - On savait déjà faire depuis longtemps.
- Défiance généralisée, nécessité *fonctionnelle* de la « cryptomonnaie ».
  - Il nous reste donc à comprendre le fonctionnement d'une « cryptomonnaie ».

Étude de quelques cas d'usage typiques

Les « cryptomonnaies »

- 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 presque 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.

Les « cryptomonnaies »

- 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 presque 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.

Les « cryptomonnaies »

- 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 presque 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.

- Une monnaie doit remplir plusieurs conditions pour être qualifiée de telle.
- En particulier, elle 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
    - pour la même raison cette utilisation est impossible,
    - 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 [Thé08].
  - Cela implique, notamment, de la confiance *sociale*, pourtant niée par les blockchains.

- Une monnaie doit remplir plusieurs conditions pour être qualifiée de telle.
- En particulier, elle doit pouvoir servir :
  - de réserve de valeur :
    - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
  - d'unité de compte :
    - pour la même raison cette utilisation est impossible,
    - 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 [Thé08].
  - Cela implique, notamment, de la confiance *sociale*, pourtant niée par les blockchains.

- Une monnaie doit remplir plusieurs conditions pour être qualifiée de telle.
- En particulier, elle doit pouvoir servir :
  - de réserve de valeur :
    - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
  - d'unité de compte :
    - pour la même raison cette utilisation est impossible,
    - 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 [Thé08].
  - Cela implique, notamment, de la confiance *sociale*, pourtant niée par les blockchains.

- Une monnaie doit remplir plusieurs conditions pour être qualifiée de telle.
- En particulier, elle doit pouvoir servir :
  - de réserve de valeur :
    - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
  - d'unité de compte :
    - pour la même raison cette utilisation est impossible,
    - 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 [Thé08].
  - Cela implique, notamment, de la confiance *sociale*, pourtant niée par les blockchains.

- Une monnaie doit remplir plusieurs conditions pour être qualifiée de telle.
- En particulier, elle doit pouvoir servir :
  - de réserve de valeur :
    - la volatilité extrême due à l'aspect purement et uniquement spéculatif l'empêche.
  - d'unité de compte :
    - pour la même raison cette utilisation est impossible,
    - 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 [Thé08].
  - Cela implique, notamment, de la confiance *sociale*, pourtant niée par les blockchains.

- 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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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, signée par elle,
  - en sortie 7 pour Bob, et 3 pour elle (son "rendu monnaie").

Transaction passée

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

Transaction d'Alice

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

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.

$\Rightarrow$ 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.

$\Rightarrow$ 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.

$\Rightarrow$ 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.

$\Rightarrow$ 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.

$\Rightarrow$ 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.

$\Rightarrow$ On doit disposer de la liste des UTXO pour vérifier la validité d'une transaction.

- 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.

- 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.

- 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.

- 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.

- 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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

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.

— Dans les entrées d'une transaction, les UTXO doivent être signées par leur cible.

Transaction passée

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

Transaction d'Alice

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

- Ce recours à la cryptographie asymétrique pose de nombreuses questions.
  - Des questions de convivialité :
    - Les concepts en jeu ne sont pas simples à appréhender.
    - Conserver ses clefs fiablement et sécuritairement est complexe.
  - Des questions techniques :
    - La perte de sa clef privée est irrémédiable et définitive.
    - La moindre erreur sur une clef publique peut rendre des fonds irrécupérables.
  - Des questions politiques :
    - Seul le détenteur d'une clef privée contrôle les fonds qui y sont associés.
    - Impossibilité d'imposer des décisions collectives (taxation par exemple).
- En pratique, la gestion des clefs est le plus souvent déléguée à un tiers…
  - Principe des "portefeuilles" hébergés sur les plateformes de trading.

Usabilité

- Ce recours à la cryptographie asymétrique pose de nombreuses questions.
  - Des questions de convivialité :
    - Les concepts en jeu ne sont pas simples à appréhender.
    - Conserver ses clefs fiablement et sécuritairement est complexe.
  - Des questions techniques :
    - La perte de sa clef privée est irrémédiable et définitive.
    - La moindre erreur sur une clef publique peut rendre des fonds irrécupérables.
  - Des questions politiques :
    - Seul le détenteur d'une clef privée contrôle les fonds qui y sont associés.
    - Impossibilité d'imposer des décisions collectives (taxation par exemple).
- En pratique, la gestion des clefs est le plus souvent déléguée à un tiers…
  - Principe des "portefeuilles" hébergés sur les plateformes de trading.

Usabilité

- Ce recours à la cryptographie asymétrique pose de nombreuses questions.
  - Des questions de convivialité :
    - Les concepts en jeu ne sont pas simples à appréhender.
    - Conserver ses clefs fiablement et sécuritairement est complexe.
  - Des questions techniques :
    - La perte de sa clef privée est irrémédiable et définitive.
    - La moindre erreur sur une clef publique peut rendre des fonds irrécupérables.
  - Des questions politiques :
    - Seul le détenteur d'une clef privée contrôle les fonds qui y sont associés.
    - Impossibilité d'imposer des décisions collectives (taxation par exemple).
- En pratique, la gestion des clefs est le plus souvent déléguée à un tiers…
  - Principe des "portefeuilles" hébergés sur les plateformes de trading.

- Ce recours à la cryptographie asymétrique pose de nombreuses questions.
  - Des questions de convivialité :
    - Les concepts en jeu ne sont pas simples à appréhender.
    - Conserver ses clefs fiablement et sécuritairement est complexe.
  - Des questions techniques :
    - La perte de sa clef privée est irrémédiable et définitive.
    - La moindre erreur sur une clef publique peut rendre des fonds irrécupérables.
  - Des questions politiques :
    - Seul le détenteur d'une clef privée contrôle les fonds qui y sont associés.
    - Impossibilité d'imposer des décisions collectives (taxation par exemple).
- En pratique, la gestion des clefs est le plus souvent déléguée à un tiers…
  - Principe des "portefeuilles" hébergés sur les plateformes de trading.

Usabilité

- Ce recours à la cryptographie asymétrique pose de nombreuses questions.
  - Des questions de convivialité :
    - Les concepts en jeu ne sont pas simples à appréhender.
    - Conserver ses clefs fiablement et sécuritairement est complexe.
  - Des questions techniques :
    - La perte de sa clef privée est irrémédiable et définitive.
    - La moindre erreur sur une clef publique peut rendre des fonds irrécupérables.
  - Des questions politiques :
    - Seul le détenteur d'une clef privée contrôle les fonds qui y sont associés.
    - Impossibilité d'imposer des décisions collectives (taxation par exemple).
- En pratique, la gestion des clefs est le plus souvent déléguée à un tiers…
  - Principe des "portefeuilles" hébergés sur les plateformes de trading.

- 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), 6.25 BTC depuis le 11 mai 2020 ;
    - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction…

    ⇒ Les frais de transaction vont 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), 6.25 BTC depuis le 11 mai 2020 ;
    - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction…

    ⇒ Les frais de transaction vont devenir colossaux.

— 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), 6.25 BTC depuis le 11 mai 2020 ;
    - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction…

    ⇒ Les frais de transaction vont devenir colossaux.

— 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), 6.25 BTC depuis le 11 mai 2020 ;
    - quand 21 millions de BTC auront été distribués, il ne restera que les frais de transaction…

    ⇒ Les frais de transaction vont devenir colossaux.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.
  - $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.
  - ⇒ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.
  - $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

Remarques

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.

  $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.

  $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- **Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.**
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.

  $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

- En terme d'usabilité et de démocratie, les blockchains ne tiennent pas la route.
  - Gestion des clefs privées : pertes irrémédiables, "portefeuilles" hébergés, etc.
- Le terme "portefeuille" n'est pas bon, il s'agit plutôt de comptes.
  - Son utilisation participe à l'*illusion de décentralisation*.
  - Rappel que *les transactions en « cryptomonnaies » sont centralisées*.
  - Les transactions sur une blockchain sont des virements internes, pas du liquide.
- **Les « cryptomonnaies » sont le seul cas d'*écriture performative* sur blockchain.**
  - Dans tous les autres cas, écrire sur une blockchain n'apporte aucune garantie.
  - Rappel qu'une « cryptomonnaie » est nécessaire au fonctionnement d'une blockchain.

  $\Rightarrow$ Blockchain et « cryptomonnaie » sont les solutions de leur propre problème.

— *GNU Taler* [BDGS16] est un *système de transaction* numérique.
  – C'est un logiciel libre.
  – Il utilise une monnaie externe, et n'est pas un moyen de stockage.
  – Il protège la vie privée des acheteurs.
  – Il permet l'auditabilité (taxation, etc.) des marchands.
  – Les *coins* et les transactions sont *bearer-based*.
  ⇒ Il permet les transactions effectivement décentralisées, et même hors-ligne.

— Il utilise de la cryptographie avancée… mais pas de blockchain .

→ `https://taler.net/fr/`

Certification

- En dehors des « cryptomonnaies », l'usage le plus souvent mis en avant.
  - Vendu avec “sécurité”, “confiance”, “vérifiabilité”, “fiabilité”, “pérennité”, etc.
- Le principe est de mettre des certificats (documents signés) dans une blockchain.
  - Avantages : distribution et immuabilité.
  - Problèmes : coûts en surplus du maintien de la base de données qui reste nécessaire.
  - Écriture non performative $\Rightarrow$ vérité assurée par une autorité tierce.
- On peut avoir les avantages sans les problèmes… en évitant les blockchains.

Certification

- En dehors des « cryptomonnaies », l'usage le plus souvent mis en avant.
  - Vendu avec "sécurité", "confiance", "vérifiabilité", "fiabilité", "pérennité", etc.
- Le principe est de mettre des certificats (documents signés) dans une blockchain.
  - Avantages : distribution et immuabilité.
  - Problèmes : coûts en surplus du maintien de la base de données qui reste nécessaire.
  - Écriture non performative $\Rightarrow$ vérité assurée par une autorité tierce.
- On peut avoir les avantages sans les problèmes… en évitant les blockchains.

Certification

- En dehors des « cryptomonnaies », l'usage le plus souvent mis en avant.
  - Vendu avec "sécurité", "confiance", "vérifiabilité", "fiabilité", "pérennité", etc.
- Le principe est de mettre des certificats (documents signés) dans une blockchain.
  - Avantages : distribution et immuabilité.
  - Problèmes : coûts en surplus du maintien de la base de données qui reste nécessaire.
  - Écriture non performative $\Rightarrow$ vérité assurée par une autorité tierce.
- On peut avoir les avantages sans les problèmes… en évitant les blockchains.

- Un exemple typique : vouloir délivrer des diplômes dans une blockchain [BL22].
- Alors que le recours à une blockchain n'est pas justifié.
  - La valeur d'un diplôme est directement liée à la confiance en l'institution qui le délivre.
  - Les acteurs ayant la possibilité de délivrer des diplômes sont identifiés.
  - ⇒ On est pas dans une situation décentralisée ni de défiance généralisée.
- Alors que le recours à une blockchain est inefficace.
  - Inenvisageable de parcourir toute la blockchain pour chaque requête de vérification :
    - pour trouver le certificat de délivrance, puis
    - pour s'assurer qu'il n'y a pas de certificat de révocation depuis.
  - ⇒ Chaque participant doit maintenir sa version utilisable de la base de données...

- Un exemple typique : vouloir délivrer des diplômes dans une blockchain [BL22].
- Alors que le recours à une blockchain n'est pas justifié.
  - La valeur d'un diplôme est directement liée à la confiance en l'institution qui le délivre.
  - Les acteurs ayant la possibilité de délivrer des diplômes sont identifiés.

  $\Rightarrow$ On est pas dans une situation décentralisée ni de défiance généralisée.
- Alors que le recours à une blockchain est inefficace.
  - Inenvisageable de parcourir toute la blockchain pour chaque requête de vérification :
    - pour trouver le certificat de délivrance, puis
    - pour s'assurer qu'il n'y a pas de certificat de révocation depuis.

  $\Rightarrow$ Chaque participant doit maintenir sa version utilisable de la base de données...

- Un exemple typique : vouloir délivrer des diplômes dans une blockchain [BL22].
- Alors que le recours à une blockchain n'est pas justifié.
  - La valeur d'un diplôme est directement liée à la confiance en l'institution qui le délivre.
  - Les acteurs ayant la possibilité de délivrer des diplômes sont identifiés.

  $\Rightarrow$ On est pas dans une situation décentralisée ni de défiance généralisée.
- Alors que le recours à une blockchain est inefficace.
  - Inenvisageable de parcourir toute la blockchain pour chaque requête de vérification :
    - pour trouver le certificat de délivrance, puis
    - pour s'assurer qu'il n'y a pas de certificat de révocation depuis.

  $\Rightarrow$ Chaque participant doit maintenir sa version utilisable de la base de données…

Alternative : Certificats type X.509 + PKI

- Des institutions naturellement présentes font autorité (États, universités).
- On sait depuis longtemps distribuer des certificats dans ce type de contexte…
- Les *certificats X.509* [UC88] sont utilisés par exemple pour les noms de domaines.
- Le sigle "PKI" signifie "*public key infrastructure*" :
  - hiérarchie d'autorités de certification (qui démarre de *certificats racines*) ;
  - infrastructure de distribution des clefs publiques permettant les vérifications (en remontant la chaîne de signatures jusqu'à un certificat racine connu).

Alternative : Certificats type X.509 + PKI

- Des institutions naturellement présentes font autorité (États, universités).
- On sait depuis longtemps distribuer des certificats dans ce type de contexte…
- Les *certificats X.509* [UC88] sont utilisés par exemple pour les noms de domaines.
- Le sigle "PKI" signifie "*public key infrastructure*" :
  - hiérarchie d'autorités de certification (qui démarre de *certificats racines*) ;
  - infrastructure de distribution des clefs publiques permettant les vérifications (en remontant la chaîne de signatures jusqu'à un certificat racine connu).

- Dans le cas de la traçabilité, il est nécessaire que les acteurs se fassent confiance.
  - Rien, pas plus une blockchain qu'autre chose, ne permet de prévenir du mensonge.
  - ⇒ On est pas dans une situation décentralisée ni de défiance généralisée.
- Dans ce cas ce qui est intéressant est l'immuabilité.
  - L'objectif est bien de garder trace des informations ajoutées au registres.

- *Git* [Tor05] est un logiciel de gestion de version décentralisé.
- La structure qui créée l'immuabilité des blockchains existe aussi dans Git.
  - Chaque *commit* (enregistrement) est identifié par un condensat de son contenu.
  - Son contenu inclut l'identifiant de son parent.
  - Il est trivial de distribuer un dépôt Git, et de garder sa copie à jour.
  - Il est également simple de permettre des ajouts par des acteurs identifiés.
- Avec quelques conventions d'utilisation entre participants, Git fait le travail.

- Les blockchains de consortium et privées disposent d'un système de permission.
  - ⇒ On est pas dans une situation décentralisée ni de défiance généralisée…
- Dans les meilleurs cas, ce ne sont en fait pas du tout des blockchains.
  - Exemple : la blockchain notariale, sans aucun mécanisme de consensus [CDH21].

— Les blockchains de consortium et privées disposent d'un système de permission.
  ⇒ On est pas dans une situation décentralisée ni de défiance généralisée…
— Dans les meilleurs cas, ce ne sont en fait pas du tout des blockchains.
  – Exemple : la blockchain notariale, sans aucun mécanisme de consensus [CDH21].

— Un *NFT* est un *jeton non fongible* enregistrée sur une blockchain.
  – Fondamentalement, une « cryptomonnaie » avec un jeton unique et indivisible.

— C'est censé servir de preuve de propriété sur un objet numérique ou physique.
  – Mais nécessite une autorité extérieure pour "rendre vraie" la propriété…

⇒ Un NFT n'est qu'un objet numérique dont la rareté artificielle permet la spéculation.
  – Ce n'est et ça ne peut être rien d'autre.
  – En particulier ce n'est pas l'objet dont ça prétend représenter la propriété.
  – La seule différence avec les « cryptomonnaies » est le fait que la valeur de chaque NFT suit son propre cours de spéculation.
  ⇒ L'étude éthique des NFT aboutie à la recommandation de ne pas en utiliser [Fli22].

Les NFT 💩

- Un *NFT* est un *jeton non fongible* enregistrée sur une blockchain.
  - Fondamentalement, une « cryptomonnaie » avec un jeton unique et indivisible.
- C'est censé servir de preuve de propriété sur un objet numérique ou physique.
  - Mais nécessite une autorité extérieure pour "rendre vraie" la propriété…

⇒ Un NFT n'est qu'un objet numérique dont la rareté artificielle permet la spéculation.

- Ce n'est et ça ne peut être rien d'autre.
- En particulier ce n'est pas l'objet dont ça prétend représenter la propriété.
- La seule différence avec les « cryptomonnaies » est le fait que la valeur de chaque NFT suit son propre cours de spéculation.

⇒ L'étude éthique des NFT aboutie à la recommandation de ne pas en utiliser [Fli22].

Les NFT 💩

— Un *NFT* est un *jeton non fongible* enregistrée sur une blockchain.
  – Fondamentalement, une « cryptomonnaie » avec un jeton unique et indivisible.

— C'est censé servir de preuve de propriété sur un objet numérique ou physique.
  – Mais nécessite une autorité extérieure pour "rendre vraie" la propriété…

⇒ Un NFT n'est qu'un objet numérique dont la rareté artificielle permet la spéculation.
  – Ce n'est et ça ne peut être rien d'autre.
  – En particulier ce n'est pas l'objet dont ça prétend représenter la propriété.
  – La seule différence avec les « cryptomonnaies » est le fait que la valeur de chaque NFT suit son propre cours de spéculation.
  ⇒ L'étude éthique des NFT aboutie à la recommandation de ne pas en utiliser [Fli22].

Le vote électronique

- Un autre usage régulièrement proposé des blockchains est le vote électronique.
  - L'usage d'une blockchain permettrait de sécuriser le vote.
  - Sécuriser contre quoi ? Ce n'est jamais clair.
  - Le fait est qu'une blockchain ne sert techniquement à rien pour un système de vote.
- Cette idée n'est pas seulement stupide, elle est aussi dangereuse.
  - Illusion de sécurité, incompréhensibilité du processus démocratique, etc.
- Ses défenseurs ne comprennent généralement pas grand chose au sujet [BLVR22].

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 (incitation).
- Une blockchain est une technologie *non neutre*, d’idéologie libertarienne [Gol16].
  - 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/organisations vers la technologie (sans en voir/admettre les limites).
    - “web3” = généralisation de la concurrence et de l’économie de marché à toute interaction.

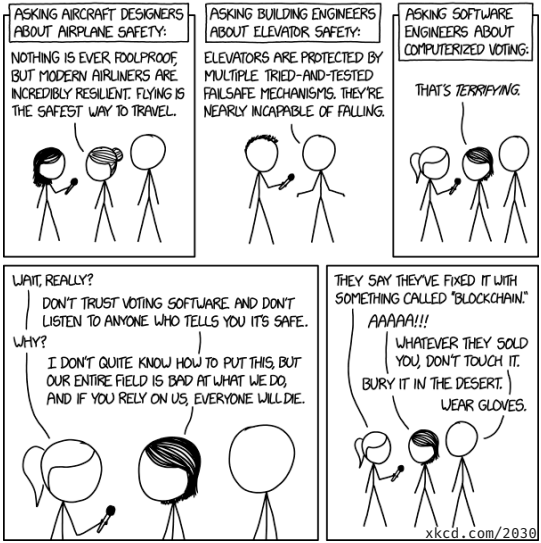
- Des blockchains partout par pur effet de mode : technosolutionnisme + confusion.
  - “J’ai un problème à résoudre.” vs “J’ai une blockchain, quel problème vais-je résoudre ?”.
  - Incompréhension de la nature *performative* de l’écriture pour les « cryptomonnaies ».
- Une blockchain n’est la solution qu’à son propre problème.
  - Seul usage valide (écriture performative) d’une blockchain : sa « cryptomonnaie ».
  - Sa « cryptomonnaie » est nécessaire à son fonctionnement (incitation).
- Une blockchain est une technologie *non neutre*, d’idéologie libertarienne [Gol16].
  - 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/organisations vers la technologie (sans en voir/admettre les limites).
    - “web3” = généralisation de la concurrence et de l’économie de marché à toute interaction.

- 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 (incitation).
- Une blockchain est une technologie *non neutre*, d’idéologie libertarienne [Gol16].
  - 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/organisations vers la technologie (sans en voir/admettre les limites).
    - “web3” = généralisation de la concurrence et de l’économie de marché à toute interaction.

- 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 (incitation).
- Une blockchain est une technologie *non neutre*, d’idéologie libertarienne [Gol16].
  - 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/organisations vers la technologie (sans en voir/admettre les limites).
    - “web3” = généralisation de la concurrence et de l’économie de marché à toute interaction.

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.



Plan

Contexte

Qu'est-ce qu'une blockchain ?

Un registre *distribué*

— Topologie de réseaux
  – Réseaux centralisés
  – Réseaux décentralisés
  – Réseaux fédérés
— Quid d'une blockchain ?

Un registre *immuable*

— Condensat cryptographique
  – L'intégrité d'une blockchain
  – Collision
— L'immuabilité d'une blockchain

Des informations qui font *consensus*

— Le problème du consensus
  – Le problème du consensus distribué
  – Le cas d'une blockchain
— La preuve de travail
  – Collision partielle
  – Minage
  – Tirage au sort non contestable
  – Nécessité d'une incitation
— La preuve d'enjeu

À retenir

Étude de quelques cas d'usage typiques

Les « cryptomonnaies »

— Détour : une monnaie ?
— Anatomie d'une transaction
  – Exemple de transactions
  – Validité d'une transaction
— Cryptographie asymétrique
  – Signature
  – Usabilité
— Incitation et récompense
— Remarques
— Alternative : GNU Taler

Certification

— Exemple des diplômes
  – Alternative : Certificats type X.509 + PKI
— Traçabilité
  – Alternative : Git
— Blockchains de consortium et blockchains privées

Les NFT

Le vote électronique

Conclusions

Perspectives (de lutte)

A. Back.
Hash cash postage implementation.
Cypherpunks mailing-list, 1997.
http://hashcash.org/.

J. Burdges, F. Dold, C. Grothoff, and M. Stanisci.
Enabling Secure Web Payments with GNU Taler.
6th International Conference on Security, Privacy and Applied Cryptographic Engineering, SPACE, 2016.
https://taler.net/papers/taler2016space.pdf.

BCDiploma and Université de Lille.
Attestations numériques blockchain de réussite au diplôme de l'Université de Lille.
Livre blanc, 2022.
https://www.univ-lille.fr/fileadmin/user_upload/presse/2022/20220114_Livre_blanc_Dem-Attest-ULille_FR.pdf.

E. Blanchard, F. Li Vigni, and P. Rauzy.
Auteur·ices, relecteur·ices : redoublons de prudence face aux effets de modes technologiques.
Rapport, 2022.
https://hal.archives-ouvertes.fr/hal-03741811.

C. Chaserant, C. Dauchez, and S. Harnay.

Du notaire à la blockchain notariale : les tribulations d'un tiers de confiance entre confiance interindividuelle, confiance institutionnelle et méfiance généralisée.
Revue juridique de la Sorbonne, n°3, 2021.
https://irjs.pantheonsorbonne.fr/sites/default/files/inline-files/Du_notaire_a_la_blockchain_notariale_C_CHASERANT_C_DAUCHEZ_S_HARNAY.pdf.

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.

C. Flick.
A Critical Professional Ethical Analysis of Non-Fungible Tokens (NFTs).
Journal of Responsible Technology (in press), 2022.
https://doi.org/10.1016/j.jrt.2022.100054.

D. Golumbia.
The Politics of Bitcoin: Software as Right-Wing Extremism.
University of Minnesota Press, 2016.
https://www.upress.umn.edu/book-division/books/the-politics-of-bitcoin.

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.

B. Théret.
Les trois états de la monnaie.
Revue économique vol. 59 n^o 4, 2008.
https://www.cairn.info/revue-economique-2008-4-page-813.htm.

L. Torvalds.
Initial revision of "git", the information manager from hell.
First git commit, 2005.
https://git-scm.com/.

Union internationale des télécommunications and Comité consultatif international télégraphique et téléphonique.
Recommandation X.509 : Annuaire – cadre d'authentification.
Série X : réseaux de communications de données : annuaire, 1988.
https://www.itu.int/rec/T-REC-X.509/recommendation.asp?lang=fr&parent=T-REC-X.509-198811-S.