# Objectifs

Découvrir le contexte historique et l’utilisation actuelle des blockchains

# Réalisation

Créer une ligne du temps avec les éléments jugés importants de l’histoire de la blockchain

Commenter succinctement les éléments particuliers trouvés

1. 1991 Première description d’une chaîne de blocs par Stuart Haber et Scott Stornetta
2. 1998 l’informaticien Nick Szabo travaille sur ‘big gold’, une monnaie décentralisée
3. 2000 Stefan Konst publie sa théorie des chaînes cryptographiques sécurisées
4. 2008 Des développeurs travaillant sous le pseudonyme Satoshi Nakamoto publient un « white paper » décrivant le model pour une block-chain
5. 2009 Nakamoto établi la première plateforme block-chain pour des transactions faites grâce au bitcoin
6. 2014 La technologie de la block-chain est séparée de la cryptomonaie et son potentiel est exploité pour d’autres sortes de transactions ( smart-contracts)

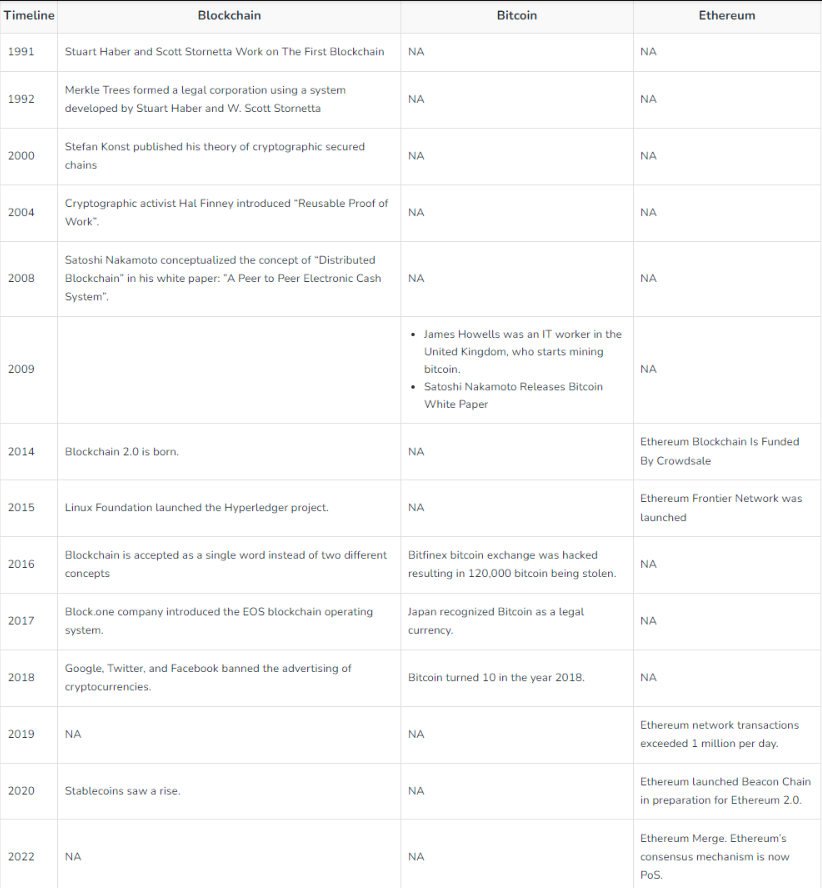
[Article source](https://www.icaew.com/technical/technology/blockchain-and-cryptoassets/blockchain-articles/what-is-blockchain/history)

1. 2015 Ethereum et le boom des altcoins
2. 2016 Explosion des ICO (crowdfunding pour la crypto) **Bitconnnnect** !!!!
3. 2017 Le bitcoin atteint $ 20000.-
4. 2018 Eclatement de la bulle spéculative et baisse des prix
5. 2019 Résolution de problème réels

[Article source](https://kriptomat.io/fr/crypto-monnaies/une-breve-histoire-de-la-crypto-monnaie/)

1. 2020 Boom des stablecoins qui promettent plus de stabilité que les cryptomonnaies classiques
2. 2022 Ethereum change de processus de consensus de « Proof of work » à « Proof of stake » ce qui a réduit la consommation d’énergie de 99.95%

[Article source](https://www.geeksforgeeks.org/history-of-blockchain/)



Lister les utilisations actuelles de la blockchain et proposez une classification

1. Accès à des services financiers pour les utilisateurs du monde entier
2. Création d’une « cryptoshère » basée sur la technologie ethereum
3. Intégration dans l’internet des objets (IOT)
4. Tokenisation des actifs classiques comme l’immobilier ou les métaux précieux
5. Stablecoins permettant l’échange à parité avec les monnaies « fiat » ($, €, ...)

Enfin, choisissez un des exemples d’utilisation et expliquez-le sur 3 chapitres : résumé, avantages et inconvénients.

**2. La crypto sphère basée sur la technologie Ethereum – les smart-contracts**

« Bitcoin a peut-être entamé la révolution des cryptomonnaies, mais c’est [Ethereum](https://www.bitpanda.com/academy/fr/lecons/quest-ce-quethereum) qui a transformé les cryptomonnaies en une véritable industrie. Grâce à son titre officieux « d’ordinateur mondial » pour [les applications décentralisées (DApps)](https://www.bitpanda.com/academy/fr/lecons/quest-ce-quune-dapp), la popularité des [smart contracts](https://www.bitpanda.com/academy/fr/lecons/tout-ce-quil-faut-savoir-sur-les-smart-contracts) et l’introduction du [standard ERC20](https://www.bitpanda.com/academy/fr/lecons/quest-ce-que-le-standard-erc20) pour les tokens, le réseau Ethereum est actuellement la première plateforme mondiale pour l’informatique distribuée. »

[Article source](https://www.bitpanda.com/academy/fr/lecons/cinq-cas-dutilisation-de-cryptomonnaies/)

**ERC20**

L'une des plus importantes [normes de contrat intelligent](https://ethereum.org/fr/developers/docs/standards/) sur Ethereum est connue sous le nom de [ERC-20](https://ethereum.org/fr/developers/docs/standards/tokens/ERC-20/), elle est apparue comme le standard technique de référence pour tous les contrats intelligents sur la blockchain Ethereum pour les implémentations de jetons fongibles.

**Getters**

function totalSupply() external view returns (uint256);

Renvoie le nombre total de jetons existants. Cette fonction est un getter et ne modifie pas l'état du contrat. Gardez à l'esprit qu'il n'y a pas de nombres décimaux dans Solidity. C'est pourquoi la plupart des jetons adoptent 18 décimales et retournent le total des jetons ainsi que d'autres résultats sous la forme de 1000000000000000000 pour 1 jeton. Tous les jetons n'ont pas nécessairement 18 décimales et c'est donc un élément à surveiller lorsqu'on manipule des jetons.

function balanceOf(address account) external view returns (uint256);

Renvoie le nombre de jetons détenus par une adresse (compte). Cette fonction est un getter et ne modifie pas l'état du contrat.

function allowance(address owner, address spender) external view returns (uint256);

La norme ERC-20 permet à une adresse de donner une allocation (« allowance ») à une autre adresse pour pouvoir récupérer des jetons à partir de celle-ci. Ce getter retourne le nombre restant de jetons que le dépenseur sera autorisé à dépenser au nom du propriétaire. Cette fonction est un getter et ne modifie pas l'état du contrat et doit retourner 0 par défaut.

**Fonctions**

function transfer(address recipient, uint256 amount) external returns (bool);

Déplace le montant amount de jetons de l'adresse de l'appelant de la fonction (msg.sender) à l'adresse du destinataire. Cette fonction émet l'événement Transfert que nous expliquerons plus tard. Il renvoie « vrai » si le transfert a été possible.

function approve(address spender, uint256 amount) external returns (bool);

function transferFrom(address sender, address recipient, uint256 amount) external returns (bool);

**Evénements**

event Transfer(address indexed from, address indexed to, uint256 value);

Cet événement est émis lorsque le nombre de jetons (valeur) est envoyé depuis l'adresse de à l'adresse à.

Dans le cas du minage de nouveaux jetons, le transfert s'opère généralement depuis l'adresse 0x00..000 alors que dans le cas d'une destruction de jetons, le transfert s'opère vers l'adresse 0x00..0000.

event Approval(address indexed owner, address indexed spender, uint256 value);

Cet événement est émis lorsque le nombre de jetons (value) est approuvé par le propriétaire owner pour etre utilisé par le dépenseur spender.

[Article source](https://ethereum.org/fr/developers/tutorials/understand-the-erc-20-token-smart-contract/)

**Avantages**

#### Fongible

Les tokens ERC-20 sont [fongibles](https://academy.binance.com/glossary/fungibility), chaque unité est interchangeable avec une autre. Si vous détenez un BinanceAcademyToken, le token spécifique que vous possédez n'a pas d'importance. Vous pourriez l'échanger contre celui de quelqu'un d'autre, et ils seraient toujours identiques, comme du liquide ou de l'or.

C'est l'idéal si votre token vise à devenir une sorte de monnaie. Vous ne voudriez pas des unités individuelles avec des traits distinctifs, ce qui les rendrait non fongibles. Ainsi, certains tokens pourraient avoir plus ou moins de valeur que d'autres, ce qui nuirait à leur utilité.

#### Flexibilité

Comme nous l'avons vu dans la section précédente, les tokens ERC-20 sont hautement personnalisables et peuvent être adaptés à de nombreuses applications différentes. Par exemple, ils peuvent être utilisés comme monnaie en jeu, dans des programmes de fidélité, comme [objets de collection numériques](https://academy.binance.com/fr/articles/a-guide-to-crypto-collectibles-and-non-fungible-tokens-nfts), ou même pour représenter des droits de propriété et des oeuvres d'art.

#### Populaire

La popularité de l'ERC-20 dans le secteur des cryptomonnaies est une raison très convaincante de l'utiliser comme modèle. Il existe une pléthore d'exchanges, de portefeuilles et de smart contracts déjà compatibles avec les tokens récemment lancés. De plus, le support des développeurs et la documentation sont nombreux.

**Inconvéniants**

#### Scalabilité

Comme pour de nombreux réseaux de cryptomonnaies, Ethereum n'est pas à l'abri des difficultés croissantes. Dans sa forme actuelle, il n'est pas très scalable : essayer d'envoyer une transaction aux heures de pointe entraîne des frais élevés et des délais. Si vous lancez un token ERC-20 et que le réseau est encombré, son utilisation peut en être affectée.

Ce n'est pas un problème exclusif à Ethereum. Il s'agit plutôt d'un compromis nécessaire pour tous les systèmes sécurisés et distribués. La communauté prévoit de résoudre ces problèmes lors de la migration vers [Ethereum 2.0](https://academy.binance.com/fr/articles/what-is-ethereum" \l "scalability-eth-2-0-and-the-future-of-ethereum" \t "_blank), qui mettra en œuvre des mises à niveau comme [Ethereum Plasma](https://academy.binance.com/fr/articles/what-is-ethereum-plasma" \t "_blank) et [Ethereum Casper](https://academy.binance.com/fr/articles/ethereum-casper-explained" \t "_blank).

Pour en savoir plus sur les problèmes de scalabilité, consultez [*Évolutivité de la blockchain : Sidechains et passerelles de paiement*](https://academy.binance.com/fr/articles/blockchain-scalability-sidechains-and-payment-channels).

#### Les escroqueries

Bien qu'il ne s'agisse pas d'un problème avec la technologie elle-même, la facilité avec laquelle un token peut être lancé peut-être considérée comme un inconvénient à certains égards. Il suffit d'un effort très faible pour créer un token ERC-20 simple, ce qui signifie que n'importe qui pourrait le faire, que ce soit à bon ou à mauvais escient.

Vous devez donc faire attention à ce dans quoi vous investissez. Il existe un certain nombre de [systèmes pyramidaux et de Ponzi](https://academy.binance.com/fr/articles/pyramid-and-ponzi-schemes) déguisés en projets blockchain. [Faites vos propres recherches](https://academy.binance.com/glossary/do-your-own-research) avant d'investir afin de vous faire vos propres idées sur la légitimité d'une opportunité.

[Article source](https://academy.binance.com/fr/articles/an-introduction-to-erc-20-tokens#pros-of-erc-20-tokens)

Timing : 60’ (+20’ mise en commun)

Document c\_Inf-M107\_P\_0001\_useOfBlockchain\_VISA.docx à rendre sous c:\temp