CORPORATE

Les smart contracts : code is law ou law is code ?

Face à la prolifération des technologies de smart contract, un constat s'impose pour tout avocat ou legal counsel : il ne s'agit pas seulement de code, c'est aussi un contrat. Il est capital de comprendre les implications de cette affirmation et de les anticiper lorsqu'on construit un protocole, une plateforme, un service mais également lorsqu'on est investisseur, client ou utilisateur de smart contracts.

Par Arthur Millerand, associé, Parallel Avocats

Dans son texte fondateur « Code is law – On liberty in cyberespace » (2000), Lawrence Lessig considère que le régulateur du cyberespace est le code (i.e. l'ensemble des protocoles TCP/IP) puisqu'il permet l'échange de données entre réseaux interconnectés de manière neutre, c'est-à-dire sans que le code ne « sache » qui sont les utilisateurs et quel est le contenu des données. Cependant, on a vu se développer des architectures permettant l'identification des utilisateurs et la qualification des contenus. Ce qui compte alors, c'est la manière dont elles sont conçues et ce qu'elles permettent de faire (ou de ne pas faire). Le code est donc porteur de valeurs et induit une régulation, dont le degré et la nature dépendent de sa conception.

Depuis ces premières réflexions, le cyberespace a connu un développement inouï et nos sociétés reposent désormais, pour une partie croissante, sur une gouvernance algorithmique (entendue comme les procédures ou prises de décisions reposant sur des traitements de données par des algorithmes). L'architecture de ces technologies est porteuse de règles qui s'imposent aux utilisateurs. Au lieu de s'appuyer ex post sur des tiers (ex. : les juridictions), les règles sont appliquées ex ante.

Dans leur article « Blockchain technology as a regulatory technology » (2018), Primavera De Filippi et Samer Hassan constatent que la technologie de la blockchain (qui est un registre) et des smart contracts (qui sont des scripts) permettent au code de jouer un rôle encore plus important dans la régulation des interactions des personnes. Ils décrivent alors le passage de « code is law » (le code a force de loi) à « law is code » (la loi est transcrite dans le code). Cette approche mérite d'être confrontée à la pratique juridique et judiciaire.

Smart contracts et contrats traditionnels : une proximité sous-estimée

En droit des contrats, l'identité et la volonté des parties sont centrales, de sorte qu'est pris en compte ce qui est écrit mais également un champ plus large, qui va de l'intention des parties, à leurs attentes et le contexte. Les contrats sont librement convenus, sous réserve des règles d'ordre public, mais ils ne peuvent être exécutés de manière forcée ou interprétés que par une juridiction.

Les smart contracts sont différents car, grâce à l'efficacité tirée de leur automatisation, ils s'exécutent, automatiquement et de manière uniforme, lorsque les conditions codées sont remplies. Cette logique assure une prévisibilité mais ne laisse pas de place à l'interprétation et aux ajustements. De plus, l'identité des parties n'est pas une condition à la formation du smart contract, ce qui, dans certaines situations, n'est pas sans poser des difficultés juridiques et procédurales.

Le code des smart contracts, quoique standardisé n'en demeure pas moins une opération relevant d'une qualification contractuelle avec la combinaison des principes d'autonomie de la volonté (les parties peuvent s'accorder sur les éléments essentiels) et du consensualisme (pas de formalisme ad validitatem). En cela les notions qui semblaient éloignées se rapprochent et le juriste, qu'il soit avocat ou legal counsel, a un rôle clé à jouer.

Intégrer la dimension juridique dans les smart contracts : une exigence décisive

Au-delà de ses caractéristiques techniques, la force normative du smart contract pourra être augmentée grâce à son incorporation dans une structure juridique « traditionnelle ». Cette approche hybride permet de régir des points juridiques clés, comme le droit applicable, les juridictions compétentes, la hiérarchie des éléments contractuels ou la limitation de responsabilité. A titre d'exemple, pour un airdrop (i.e. la distribution gratuite de tokens), il est courant que les smart contracts exécutent la distribution selon les critères convenus dans le code et que d'autres documents complètent cette dimension informatique (ex. : les Terms of Services, le Whitepaper, les communications du protocole ou l'interface utilisateur).

Si l'immuabilité des smart contracts est une caractéristique souhaitable pour des raisons de sécurité et de preuve (il y a une traçabilité des flux et le code

peut être revu), cela peut devenir une difficulté en cas de bug ou de situation imprévue. Il est ainsi pertinent de mettre en place des mécanismes permettant de faire des adaptations, sans pour autant nuire à la décentralisation. C'est notamment le cas des fonctionnalités permettant à la communauté ou à la DAO (Decentralised Autonomous Organisation) d'intervenir pour suspendre des opérations ou apporter des modifications. On peut faire référence à la mise à jour des smarts contracts du protocole Uniswap permettant d'ajuster les paramètres sur le long terme ou la correction apportée au smart contract du protocole Compound en 2021 à la suite d'un bug.

Par ailleurs, les contrats traditionnels peuvent faire l'objet de renégociations ou d'adaptations s'ils s'avèrent inadaptés à l'évolution des circonstances. Or, pour prévoir une obligation de renégociation (ex. : clause de hardship), encore faut-il être en mesure de détailler, en amont, les conditions dans lesquelles elle pourrait être actionnée. Pour les smart contracts, la difficulté est encore plus grande car il faut prévoir, dans le code, les conditions permettant (ou non) de conduire à une renégociation.

Cette absence de plasticité peut avoir des conséquences significatives, comme cela fut le cas pour le protocole MakerDAO suite au krach crypto de mars 2020. A la suite de ce retournement de marché, des ajustements aux smart contracts et une évolution de la gouvernance ont été décidés pour protéger l'avenir du protocole mais des dommages significatifs avaient déjà eu lieu.

Il faut aussi souligner l'importance du design des protocoles et des interfaces utilisateurs car cela permet de renforcer la portée juridique des accords. On peut faire référence au cas AAVE/CoW (mars 2026) où un trader a utilisé l'interface mobile officielle d'AVVE pour échanger (swap) des cryptoactifs et a subi d'importantes pertes. L'application avait pourtant averti l'utilisateur que sa transaction aurait un impact significatif sur les prix et qu'il pouvait reconsidérer sa décision. Le legal design peut être déterminant.

Le contrôle juridictionnel sur les smart contracts : un levier dont l'effectivité est variable

Si les juristes sont habitués à penser aux juridictions comme recours en cas de litige contractuel, l'utilisation d'un smart contract, lequel est constitué de conditions objectives, claires et précises, semble remettre en cause l'intervention du juge. Pourtant, ce n'est pas parce que le code autorise une action que celle-ci est acceptable sur le plan juridique. Le code n'est donc pas absolu ou infaillible, de sorte que la légalité des transactions permises par les smart contracts, ou le résultat de leur exécution, peuvent être questionnés (ex. : en cas de faille dans le code).

Le cas Mango Markets (2022) est le meilleur exemple : Avraham Eisenberg avait soutenu que son trading était « légal » puisqu'il exploitait une faille des smart contracts. L'argument selon lequel « code is law » n'a pas été suivi par les juridictions américaines (si sa condamnation a été annulée en mai 2025, la cour n'a pas retenu qu'il était légal de faire ce que le code informatique permettait). Mais alors qui est responsable ? Le protocole, ses concepteurs, ses développeurs, les entreprises ayant édité un des modules concourant au fonctionnement du smart contract ou bien l'oracle qui a fourni les données permettant au smart contract de s'exécuter ?

On peut mentionner le jugement du tribunal judiciaire de Créteil (mai 2025) qui a rejeté les demandes d'investisseurs dirigées contre trois personnes identifiées comme ayant développé le projet et soutenant avoir été lésées à la suite d'un piratage et d'un dysfonctionnement allégué d'une fonction du smart contract. Ainsi, la seule participation à l'élaboration d'un smart contract n'induit pas une responsabilité, sauf à démontrer les conditions de la responsabilité civile (une faute, un préjudice et un lien de causalité), ce qui n'était pas le cas en l'espèce. Si les décisions sur les smart contracts sont rares, les principes de droit demeurent applicables et la sécurité informatique des smart contracts est essentielle pour la création d'un écosystème viable d'applications décentralisées (DApps) et de services reposant sur cette technologie.

Maîtriser les smart contracts : un avantage de long terme

La question n'est pas de savoir si une organisation sera concernée par la conclusion de smart contracts mais quand cela interviendra. Cette interrogation n'est pas limitée à une niche de marché mais se pose déjà quotidiennement dans les industries financières, assurantielles et créatives. Beaucoup d'autres suivront. Aussi, chaque dirigeant(e) ou directeur (trice) juridique doit identifier les situations dans lesquelles leurs opérations impliquent des smart contracts afin de conduire des vérifications nécessaires. ■