



Collection lausannoise
CEDIDAC

Alexandre Richa / Damiano Canapa
(éditeurs)

Droit et économie numérique

Unil



Stämpfli Editions



Collection lausannoise
CEDIDAC

Alexandre Richa / Damiano Canapa
(éditeurs)

Droit et économie numérique



Collection lausannoise

Volume 73

Comité éditorial

Hansjörg Peter; Damiano Canapa, Robert J. Danon,
Anne-Christine Favre, Andrew M. Garbarski, Eva Lein

Volumes 1 à 72 publiés dans la collection Recherches juridiques
lausannoises

Sous-collection CEDIDAC (volume 107) dirigée par Damiano Canapa,
fondée par François Dessemontet sous le titre Publication CEDIDAC et
continué par Jean-Marc Rapp et Edgar Philippin



Stämpfli Editions

© Stämpfli Editions SA Berne



Collection lausannoise
CEDIDAC

Droit et économie numérique

Édité par

Alexandre RICHA

Professeur à l'Université de Lausanne, avocat

Damiano CANAPA

Professeur à l'Université de Lausanne, directeur du CEDIDAC



Stämpfli Editions

© Stämpfli Editions SA Berne

Information bibliographique de la Deutsche Nationalbibliothek
La Deutsche Nationalbibliothek a répertorié cette publication dans la Deutsche Nationalbibliografie; les données bibliographiques détaillées peuvent être consultées sur Internet à l'adresse <http://dnb.d-nb.de>.

Tous droits réservés, en particulier le droit de reproduction, de diffusion et de traduction. Sans autorisation écrite de l'éditeur, l'œuvre ou des parties de celle-ci ne peuvent pas être reproduites, sous quelque forme que ce soit (photocopies, par exemple), ni être stockées, transformées, reproduites ou diffusées électroniquement, excepté dans les cas prévus par la loi.

© Stämpfli Editions SA Berne • 2021
www.staempfliverlag.com

Print ISBN 978-3-7272-3467-5

Dans notre librairie en ligne www.staempflishop.com,
la version suivante est également disponible :

E-Book ISBN 978-3-7272-3468-2

printed in
switzerland



© Stämpfli Editions SA Berne

Smart contracts – de la théorie à la pratique

Michel JACCARD

Avocat

Mehmet TORAL

Avocat

“[...] if the owner fails to make payments, the smart contract invokes the lien protocol, which returns control of the car keys to the bank. This protocol might be much cheaper and more effective than a repo man. A further reification would provably remove the lien when the loan has been paid off, as well as account for hardship and operational exceptions. For example, it would be rude to revoke operation of the car while it's doing 75 down the freeway.”¹

I. Introduction

L'essor et la popularité des technologies *blockchain* ces dernières années reposent sur quelques caractéristiques fondamentales : (i) les *blockchains* permettent de stocker et donner accès à des informations de manière sûre, (ii) elles sont immuables, chaque transaction étant enregistrée et accessible en permanence (rien ne se perd), et (iii) les plus importantes en termes d'utilisation ne sont – du moins en théorie – sous le contrôle de personne en particulier, du fait de leur nature distribuée.

De par ces caractéristiques, les technologies *blockchain* peuvent être utilisées dans un nombre incalculable de domaines - vu qu'il s'agit essentiellement d'une nouvelle manière de stocker et mettre à jour des informations, il y a autant de domaines d'application qu'il y a de données à écrire et à déplacer. Mieux encore, la nature informatique de la technologie permet une automatisation importante des processus. Quoi de plus naturel, donc, que d'imaginer des contrats entiers dont la conclusion et l'exécution se ferait via *blockchain* au lieu de documents classiques ?

L'accord ainsi formé et reflété sur un registre distribué serait à l'abri des aléas de la volonté changeante des parties et des vicissitudes d'une justice trop humaine – et donc faillible. Une fois les ressources elles-mêmes mises à disposition via *blockchain* (sous forme de crypto-monnaies ?), le contrat

¹ SZABO, Formalizing and Securing Relationships on Public Networks.

n'aurait plus besoin des avocats, juges, comptables et banquiers pour s'exécuter, et deviendrait, en un sens, intelligent.

Un *smart contract*, en somme.

Notre contribution examine les effets sur la pratique contractuelle d'une utilisation de *smart contracts* selon le cycle contractuel suivant :



Nous exposerons que le *smart contract* initialement conçu comme pur outil d'exécution d'une volonté exprimable sous forme numérique est susceptible d'avoir des effets sur la formation de la volonté contractuelle par les moyens qu'il met en œuvre (*cf. infra* II). Nous examinerons ensuite une question d'ordinaire laissée de côté, soit comment, concrètement, on peut envisager un passage d'un contrat tel que connu des juristes, à un *smart contract* tel que conçu par des ingénieurs (*cf. infra* III). Enfin, nous terminerons avec quelques observations sur les aspects liés à l'exécution de la volonté (*cf. infra* IV).

A. Définition et description du contrat « classique »

Le terme « contrat » désigne deux notions : d'une part une relation juridique établie (volontairement) entre deux ou plusieurs parties, d'autre part la manifestation de l'établissement de cette relation juridique, le plus souvent sous forme d'un document écrit.²

En droit suisse, la relation juridique est établie par une manifestation concordante de volontés.³ La formalisation écrite du contrat est la preuve de

² *Cf.* TERCIER/BIERI/CARRON, N 7-13 qui distingue encore l'acte juridique (soit l'échange des manifestations de volonté) de la relation juridique (soit la relation contractuelle en tant que tel).

³ Art. 1 al. 1 CO : « *Le contrat est parfait lorsque les parties ont, réciproquement et d'une manière concordante, manifesté leur volonté* ». A noter que le Code fédéral des obligations du 14 juin 1881 était encore plus direct : « *Il n'y a contrat que si les parties ont manifesté d'une manière concordante leur volonté réciproque* » (nous soulignons) *cf.* ZK-JAEGGI *ad* art. 1-17 CO, mettant bien l'accent sur la relation fondamentale entre l'existence même du contrat et la manifestation concordante des volontés, alors que l'itération plus moderne (bien que maintenant centenaire) est focalisée sur la perfection du contrat plutôt que son existence... laissant la place aux contrats imparfaits. Nous

cette manifestation lorsque le contrat n'est soumis à aucune exigence formelle spécifique. Lorsque la loi réserve une forme particulière, le respect de l'exigence formelle conditionne la validité de la manifestation de volontés.⁴

La manifestation de volonté elle-même doit porter sur un contenu matériel minimum, dont le respect est nécessaire et suffisant à la création du contrat : le contrat est ainsi réputé conclu lorsque les parties sont d'accord sur ses points « essentiels ». Le juge est compétent pour régler le contenu des points secondaires sur lesquels aucun accord n'aurait été trouvé.⁵

Le « contrat » tel que compris par le juriste est une notion qui englobe donc à la fois un mode de création d'obligations et un mode de cristallisation d'obligations, mais qui s'arrête là où commence l'exécution. Exception faite des contrats qui comprennent un acte formateur, soit les contrats qui ne se bornent pas à créer une obligation mais qui modifient une situation juridique (par exemple : un acte de cession), le contrat est en effet un acte générateur d'obligations mais non un acte d'exécution en tant que tel.

B. Définition et description du *smart contract*

Une définition du *smart contract* communément admise est celle fournie par Nick SZABO en 1997 : « *Smart contracts combine protocols, user interfaces,*

laissons de côté ici le contrat « normatif » créé par application du principe de confiance lorsque les volontés internes des parties sont en fait non-concordantes (cf. p.ex. ATF 144 III 93 ; ATF 130 III 417). En effet, l'impossibilité pour le juge d'établir la volonté concordante des parties est due à un dysfonctionnement qui n'a pas de place dans l'environnement du *smart contract* : notre contribution vise à montrer, entre autres, que le *smart contract* doit décrire complètement les droits et obligations qu'il incorpore afin d'être efficace.

⁴ Sur la question du respect du formalisme juridique dans le cadre d'échanges de données informatisées, voir M. JACCARD, La conclusion de contrats par ordinateur, p. 187 ss.

⁵ Pour un examen détaillé du système établi par l'art. 2 al. 1 et 2 CO, cf. TERCIER/PICHONNAZ, N 609 ss. On notera en particulier la distinction entre les clauses nécessaires qui « *individualisent le contrat* » et sans lesquelles « il ne peut y avoir accord » (N 610) et les clauses secondaires qui « *aménagent le contrat* ». A noter également que le principe posé à l'art. 2 al. 1 CO que le contrat est réputé conclu si « *les parties se sont mises d'accord sur tous les points essentiels [...] lors même que des points secondaires ont été réservés* » est lui-même sujet à interprétations multiples, dès lors qu'on considère que cette disposition pose une présomption réfragable (approche dynamique) ou irréfragable (approche statique). Au vu du déterminisme du *smart contract* (cf. *infra* III.C.1 et III.C.2.f)), la question devrait en principe rester d'intérêt purement académique dans le contexte qui nous intéresse.

and promises expressed via those interfaces, to formalize and secure relationships over public networks. »⁶

Autrement dit, le *smart contract* serait la résultante de l'expression d'une promesse à travers une interface utilisateur communiquant selon des règles prédéterminées (un protocole). Cette expression répondrait au double objectif de la formalisation et de la cristallisation d'une relation.

Dans cette perspective, le *smart contract* se recoupe avec le contrat « classique » dans la composante de la formalisation, mais s'en détache pour le surplus. Alors que le contrat « classique » s'attache à la formation de la volonté et sa formalisation comme moyen d'ancrer cette volonté, le *smart contract* est plus centré sur la formalisation d'un accord de volontés de manière à en assurer l'exécution.

Le *smart contract* vit ainsi principalement dans les deux dernières étapes du cycle contractuel tel que décrit plus haut : la formalisation de la volonté et son exécution. La formation de la volonté n'est pas nécessairement étrangère au *smart contract*, mais peut être considérée comme neutre, du moins dans l'acception communément admise du *smart contract*, qui ne nécessite aucune volonté concordante : il lui est suffisant d'avoir une volonté quelconque à cristalliser et à exécuter.

Cette neutralité par rapport à la formation de la volonté mène d'ordinaire à considérer que le *smart contract* n'est en fait pas un contrat au sens juridique du terme,⁷ ou du moins que sa conclusion ne correspond pas nécessairement à la conclusion d'un contrat au sens classique.⁸

⁶ SZABO, *Formalizing and Securing Relationships on Public Networks*. Pour un tour d'horizon des diverses définitions possibles qui sont en concurrence, cf. FAVROD-COUNE/BELET, p. 1106. Il est remarquable que les seules définitions s'approchant d'une neutralité technologique soient fournies par les informaticiens (Nick Szabo et Vitalik Buterin), alors que les définitions produites par les juristes sont systématiquement complétées par une référence au support technologique. Ainsi, G. JACCARD, N 9 : « [...] , and that is stored on a distributed ledger » ; ESSEBIER/WYSS, N 30 : « [...] die in einer Blockchain gespeichert und repliziert werden können » ; MEYER/SCHUPPLI, p. 208 : « [...] gestützt auf die Blockchain-Architektur, beim Eintritt gewisser Bedingungen selbst ausführen und aufgrund der dezentralen und kryptografischen Ausgestaltung der Blockchain selbstdurchsetzend und manipulationssicher sind » ; CARRON/BOTTERON, p. 106 : « [...] and stored in the blockchain [...] » ; MÜLLER, N 6 : « [...] basé sur la technologie de la blockchain [...] » (tout en admettant, id. N 8, que la blockchain n'est en fait pas nécessaire au smart contract).

⁷ Dans ce sens, cf. MÜLLER, N 7, pour qui : « le terme « Smart Contract » est particulièrement mal choisi, étant donné qu'un « Smart Contract » n'est ni un contrat au sens juridique du terme, ni smart. Il n'est pas un contrat au sens juridique du terme, mais un programme informatique servant à la conclusion et l'exécution de véritables contrats » (N 7). Voir aussi M. JACCARD, *La conclusion de contrats par ordinateur*, p. 392 ss.

⁸ CARRON/BOTTERON, p. 108.

II. Formation de la volonté

A. Formation de la volonté dans le CO : de la systématique à la logique

En droit suisse, le contrat est établi par une manifestation concordante de volontés. Cette manifestation suit un modèle d'offre et d'acceptation décrit aux articles 3 et suivants CO.

L'offre et l'acceptation sont traitées comme des sujets liés mais néanmoins distincts, suivant la séquence suivante :

- durée de validité d'une offre avec délai pour l'acceptation (art. 3 CO) ;
- durée de validité d'une offre sans délai pour l'acceptation si faite entre présents (art. 4 CO) ;
- durée de validité d'une offre sans délai pour l'acceptation si faite entre absents (art. 5 CO) ;
- acceptation tacite d'une offre dans les cas particuliers ne nécessitant pas d'acceptation expresse (art. 6 CO) ;
- cas particuliers dans lesquels une manifestation de volonté de l'offrant n'est pas considérée comme une offre (art. 6a et 7 CO) ;
- cas particulier de l'offre publique d'une partie intéressée à obtenir une prestation en échange d'un prix (art. 8 CO) ; et
- règles applicables au retrait d'une offre ou d'une acceptation (art. 9 CO).⁹

Si la présentation dans le code des obligations a l'avantage d'une certaine simplicité, on remarquera néanmoins d'emblée que l'ordre dans lequel les sujets sont traités ne répond pas nécessairement à une logique algorithmique claire. Sur ce point, le code des obligations ne suit en effet pas un arbre décisionnel linéaire, mais est construit autour d'une approche consistant à établir des grandes catégories par systématisation, combinée avec un mouvement du général vers le particulier (soit, traitement des offres avec délai

⁹ L'art. 10 CO qui détermine le moment auquel les effets du contrat remontent est en général inclus dans l'exposé du système de la formation de la volonté en droit suisse (cf. p.ex. TERCIER/PICHONNAZ, N 644), et se trouve rangé dans la même partie du CO, faisant partie du chapitre « A. Conclusion du Contrat » au même titre que les art. 1 à 9 CO. Cela étant, il s'agit d'une disposition qui n'a aucun effet lorsqu'il s'agit de déterminer si un contrat a été conclu, ne réglant que la question du moment à partir duquel le contrat déploierait ses effets. Nous n'examinons donc pas cette disposition dans cette partie qui est focalisée exclusivement sur la formation du contrat, la question du moment de sa prise d'effet pouvant rester ouverte.

vs offres sans délai, puis des offres entre présents vs offres entre absents comme sous-catégorie des offres sans délai).

Pour adopter une approche de logique séquentielle visant à optimiser le processus décisionnel, il faudrait plutôt commencer par se poser la question de la validité de base de l'offre (art. 6a, 7 et 9 al. 1 CO). En effet, sans offre valable, tout le reste de l'analyse devient sans objet. On devrait ensuite se poser la question de savoir si une acceptation formelle est nécessaire à la conclusion du contrat (art. 6 et 8 CO), puisque dans le cas contraire on pourra de nouveau arrêter l'analyse. Enfin, on peut se demander pendant combien de temps l'offre reste valable (art. 3, 4 et 5 CO). La logique séquentielle rejoint l'ordre du code des obligations sur le point du retrait de l'acceptation (art. 9 al. 2 CO) : dernière sortie possible de l'arbre décisionnel une fois toutes les autres questions traitées.

En tout état, cette brève dissection du processus d'offre et d'acceptation montre déjà que la systématisation de la logique juridique dans le code des obligations n'est pas incompatible avec une logique « informatisable », mais qu'une réduction des dispositions légales à un arbre décisionnel nécessite néanmoins que les juristes impliqués prennent un recul important par rapport aux textes législatifs.

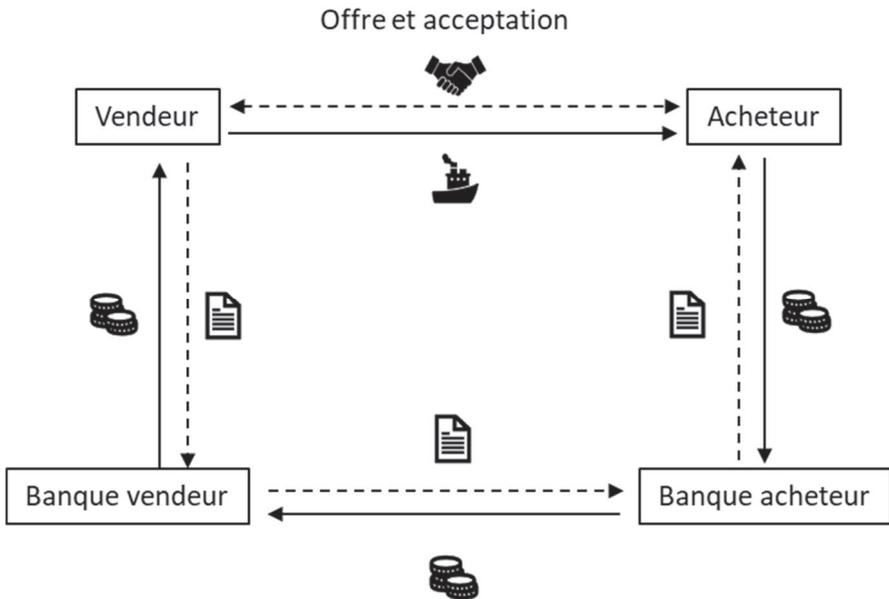
Cela étant, il est fort probable qu'en pratique l'usage de *smart contracts* finisse par priver l'analyse du processus de l'offre et de l'acceptation de sa pertinence, comme nous le démontrerons par un exemple dans le chapitre suivant.

B. Formation de la volonté dans un environnement de *smart contracts* : la volonté a-t-elle encore un sens ?

Certaines activités nécessitent la mise en place d'un nombre relativement important de contrats pour une opération en apparence assez simple. Il en va notamment ainsi du cycle de crédit documentaire. Dans cette partie, nous utiliserons l'exemple du cycle de crédit documentaire pour illustrer comment un passage à des *smart contracts* peut neutraliser la question de la formation de la volonté.

Dans son expression la plus simple, le cycle de crédit documentaire implique, en plus de l'importateur et l'exportateur des marchandises, deux banques intermédiaires dans un système visant à pallier l'absence de confiance réciproque des parties avec (i) des relations contractuelles et de confiance entre chaque partie et sa banque et (ii) des relations contractuelles et de confiance entre les banques elles-mêmes. Le schéma se présente comme suit, avec les

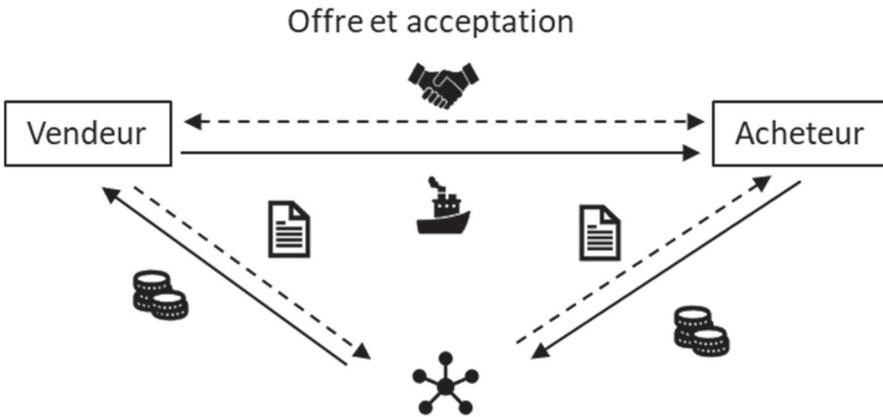
flux d'argent et de marchandises en trait plein et les flux d'informations/accords contractuels en pointillés :



Après conclusion du contrat, l'acheteur/importateur ouvre une lettre de crédit via sa banque. Le vendeur/exportateur remet les documents relatifs à la marchandise à sa banque, qui les fait parvenir à l'acheteur via la banque de cette dernière. Une fois les documents arrivés, la lettre de crédit est « débloquée » par un flux de fonds dans le sens inverse, de l'acheteur au vendeur en passant de nouveau par la banque de chacun.

Dans ce schéma, la relation entre vendeur et acheteur se noue de manière classique, avec offre et acceptation reconnaissables sous l'angle du code des obligations. L'ensemble des possibilités offertes par le code des obligations est potentiellement pertinent (offres avec ou sans délais d'acceptation, entre présents ou absents, par offres publiques ou de gré à gré, *etc.*).

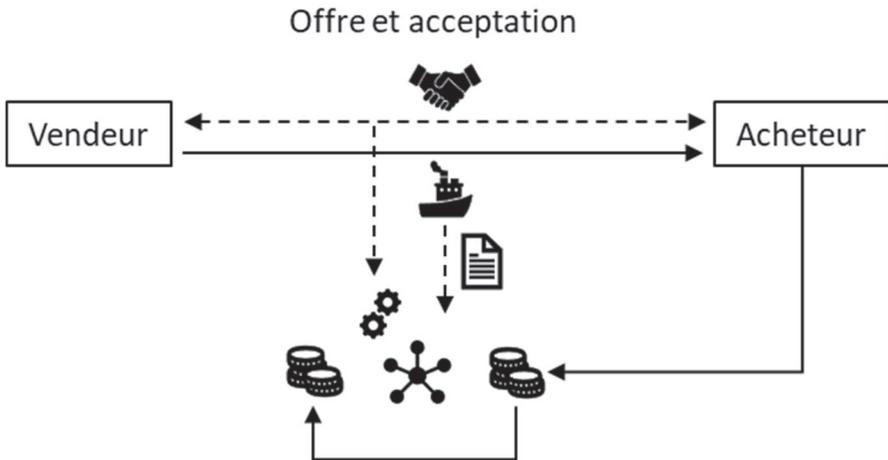
Le passage par une *blockchain* permet (du moins en théorie) une simplification de ce processus par désintermédiation sous la forme suivante :



Dans ce schéma, on garde encore une fois le processus de formation de la volonté, le recours à une *blockchain* permettant uniquement de simplifier l'opération sans remise en question de la situation juridique.¹⁰ Les parties arrivent en effet à échanger de l'information en ayant la confiance nécessaire sur le fait que l'intégrité de l'information est protégée, et arrivent à échanger des valeurs par les mêmes moyens (en imaginant que le paiement se fasse à travers une cryptomonnaie en la valeur de laquelle les deux parties ont confiance) sans passer par leurs banques respectives. A noter qu'il n'y a pas encore de *smart contract* dans ce système (sauf dans le sens le plus étroit du code informatique permettant l'échange des informations et valeurs par les parties).

L'intégration d'un *smart contract* comme pur outil d'exécution donne ce qui suit :

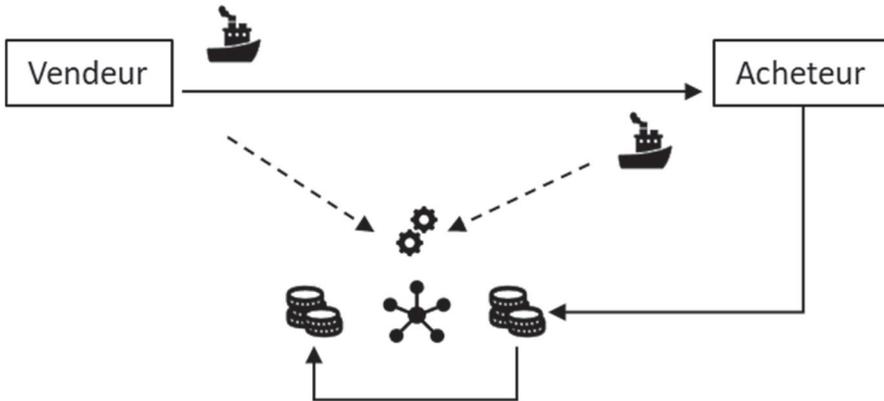
¹⁰ Nous faisons abstraction ici de la question de la relation juridique établie avec les contrôleurs ou développeurs de la *blockchain* utilisée. Cf. MEYER/SCHUPPLI, p. 210 ss pour des développements approfondis à ce sujet. La désintermédiation est une des premières utilités de la *blockchain*, pour ne pas dire la principale d'un point de vue juridique – cf. GILLIERON, Transformation numérique, p. 137 ; IFFLAND, p. 504. Certains voient dans l'utilisation des *blockchains* et des *smart contracts*, en particulier dans l'utilisation des ICO pour lever des fonds, une « capacité à pousser la désintermédiation financière à son paroxysme », cf. DARBELLAY/REYMOND, p. 11. Le Conseil fédéral se montre plus circonspect, indiquant simplement que l'utilisation de blockchain « pourrait par exemple contribuer à l'amélioration de l'efficacité et de la résilience des processus dans le secteur financier, voire même conduire à une désintermédiation » (Rapport du Conseil fédéral du 14 décembre 2018, Bases juridiques pour la *distributed ledger technology* et la *blockchain* en Suisse - État des lieux avec un accent sur le secteur financier, p. 13).



En apparence, le contrat est toujours conclu de la même manière, par offre et acceptation. Par contre, il est ensuite virtualisé sur une *blockchain* sous forme de *smart contract*. Le vendeur ne fournit plus d'informations au système, celles-ci étant vérifiées de manière « indépendante » et automatique par le *smart contract*. Enfin, l'acheteur n'envoie pas de valeurs au vendeur via une *blockchain* en échange de l'information fournie par le vendeur, mais alimente le *smart contract* qui libère les fonds en faveur du vendeur suite à sa vérification indépendante et automatique.

Dans un tel système, le risque de non-paiement est quasi nul : le *smart contract* peut être conçu pour que rien ne se passe – il n'entre pas en vigueur – tant que l'acheteur n'a pas alimenté son compte. Le *smart contract* n'est en effet utile que s'il permet l'exécution automatique, qui est sa raison d'être. Sans les moyens de cette réalisation (soit, dans le cas qui nous occupe, un accès aux fonds nécessaires pour sécuriser le paiement), le *smart contract* représentera au mieux un moyen de preuve de l'intention des parties – rôle auquel il sera singulièrement mal adapté en raison de la forme qu'il prendra (code informatique).

Pour pousser la réflexion plus loin, on peut considérer que dans la mesure où la confiance est garantie par le système et où notre vendeur a la certitude d'obtenir le paiement de son prix, il lui est égal de savoir qui achète sa marchandise. En pratique, le *smart contract* trouvera donc son utilité première lorsqu'il permettra un affranchissement total du processus d'offre/acceptation qui implique un contact entre les parties, en établissant un schéma qui prendra la forme suivante :



Dans ce schéma, le vendeur crée un *smart contract* avec les conditions de livraison (prix, moment) pré-intégrées. L'acheteur alimente en espèces le *smart contract* (dont on imagine qu'il peut apprendre l'existence en consultant, par exemple, une plateforme qui listerait les types de contrats qui l'intéressent et qui sont à disposition). A l'arrivée de la marchandise, le système de transport envoie l'information au *smart contract*, qui libère les fonds en faveur du vendeur.

Première observation : il n'y a plus de communication, même indirecte, entre l'acheteur et le vendeur. Tous les flux d'information sont à destination du système. Seconde observation : nous sommes dans le cas d'une offre publique avec une acceptation par actes concluants. Il s'agirait dans ce cas du scénario de l'article 7 alinéa 3 CO, qui devrait s'appliquer par analogie. Le cas inverse peut également être envisagé, avec un acheteur qui pourrait créer un *smart contract* pré-alimenté en fonds pour recevoir un certain type de marchandises – auquel cas nous serions plutôt dans le cadre de l'article 8 alinéa 1 CO.

On peut toutefois également se demander si les notions d'offre et d'acceptation gardent une pertinence quelconque dans ce cadre, hormis d'un point de vue purement théorique ou dans des situations marginales. En effet, les dispositions idoines du code des obligations répondent aux questions de savoir quand une offre et une acceptation sont valables et susceptibles de donner naissance à un contrat. Or, si la relation juridique peut être entièrement réduite au *smart contract*, il n'y a plus de place pour l'incertitude quant au processus d'offre et d'acceptation : le *smart contract* est soit effectif (et efficace), soit n'existe tout simplement pas en tant qu'objet juridiquement pertinent. Sauf cas de fraude, les parties ne pourront invoquer l'absence d'offre et d'acceptation dès lors que le *smart contract* existe et a obtenu les moyens nécessaires à son exécution.

Bien plus épineuse sera la question des vices éventuels affectant le consentement. Afin que le *smart contract* puisse remplacer un contrat ordinaire sans risque, ou du moins sans créer plus de problèmes qu’il n’en résout, encore faut-il que son contenu soit aisément compréhensible par les parties – ce qui risque fort de réduire son utilité pratique aux domaines permettant une standardisation par adoption de *smart contracts* usuels pour des branches d’activité données.

Cela nous amène au sujet de la formalisation de la volonté, traité dans le chapitre suivant.

III. Formalisation de la volonté

Du point de vue du juriste, une grande partie des discussions autour de la relation entre contrat et *smart contract* est focalisée sur la notion de consentement et sur la relation entre le *smart contract* et l’ordre juridique (soit, quand un *smart contract* est-il un contrat, et quel effet juridique doit-on reconnaître au *smart contract* ?¹¹). Pour les informaticiens, les discussions ont lieu autour de l’exécution (soit, comment assurer l’efficacité d’un contrat dans un environnement informatique par recours à un *smart contract* ?).

Alors que quelques efforts ont été entrepris par des informaticiens pour formaliser des langages et protocoles informatiques appropriés à la retranscription d’obligations juridiques,¹² il manque encore un examen systématique et méthodique de la rédaction contractuelle pour permettre aux juristes de faire le pas vers une informatisation de leur outil de travail.

Dans cette partie, nous allons donc présenter un exemple de « numérisation » de la rédaction et de la conclusion d’un simple contrat bilatéral, un contrat de

¹¹ Cf. p.ex. CARRON/BOTTERON, dont l’optique est d’examiner « *the legal aspects of smart contracts and their integration into Swiss contractual law* » (p.103), ou encore G. JACCARD, (extrait du résumé de l’article : « *[t]he paper gives an overview on smart contracts and assesses their legal relevance [...] first explores how smart contracts can be relevant in the eyes of the law and then differentiates and assesses smart contract with regards to their types.* »... et plus généralement toutes les références d’articles de juristes se trouvant dans la bibliographie en fin d’article.

¹² Pour une approche théorique, cf. SZABO, *A Formal Language for Analyzing Contracts* ; pour un des premiers langages de programmation orientés *smart contract*, cf. <http://www.erights.org> pour une description du langage « E » développé en 1997 par Mark. S. Miller (consulté le 06.09.2020) ; pour un projet plus récent, cf. Accord Project, (<https://www.accordproject.org>, consulté le 28.09.2020), dans le cadre duquel un langage informatique *ad hoc* été développé pour l’exécution de clauses contractuelles (ergo, disponible en *open source*, <https://github.com/accordproject/ergo>, consulté le 28.09.2020) et un ensemble de modèles mettant en œuvre le langage (cicero, <https://github.com/accordproject/cicero>, consulté le 28.09.2020).

prêt, afin de mettre en lumière des éléments de réflexion auxquels les rédacteurs de contrats sont rarement confrontés.

A. Contrat exemple

Pour la construction de cet exemple, nous prenons comme base un contrat de prêt de consommation simple comme suit :

Loan Agreement

dated [date] between [name], [address] (the Lender) and [name], [address] (the Borrower)

Preamble

The Borrower is a company active in [business]. The Borrower has asked the Lender to grant a loan for an aggregate amount of [currency] [amount], which the Lender has accepted pursuant to the terms of this loan agreement (the Agreement).

1. Loan

1.1. Amount. *Subject to the terms of this Agreement, the Lender hereby grants to the Borrower a loan (the **Loan**) in the amount of CHF [amount] (the **Principal Amount**).*

1.2. Disbursement. *The Lender shall disburse the Principal Amount in full, free of any costs or charges, no later than 3 days from the date of this Agreement to the bank account notified by the Borrower to the Lender for such purposes.*

1.3. Interest. *The Loan, during its term, shall bear interest at the rate of 5% on a 30/360 basis.*

1.4. Term. *The Loan is granted until [date] (the **Maturity Date**).*

2. Reimbursement

2.1. Reimbursement. *The Loan, including any interest, shall be reimbursed on the Maturity Date. The Loan may not be prepaid without the consent of the Lender.*

2.2. Acceleration. *On and at any time after the occurrence of any of the events listed hereafter, the Lender may declare by written notice to the Borrower that the Principal Amount and all accrued interest under the Agreement shall be immediately due and payable:*

the Borrower has a resolution passed for its winding-up; or

the Borrower suspends all of its business operations.

3. **Governing Law and Jurisdiction**

3.1. Governing Law. *This Agreement shall be governed by and construed in accordance with the laws of Switzerland.*

3.2. Jurisdiction. *Any dispute or difference arising out of or in relation to this Agreement shall be subject to the exclusive jurisdiction of the courts of Switzerland.*

Lender

Borrower

Signature

Signature

B. Déconstruction juridique du contrat

1. Aspects de fond

Une analyse simple des aspects de fond du contrat par un juriste se ferait comme suit :

Droit applicable : application du droit suisse en vertu de l'article 116 alinéa 1 LDIP sur la base de la clause d'élection de droit.

Qualification du contrat : prêt de consommation soumis aux articles 312 à 318 CO.

Éléments principaux du contrat : parties et montant du prêt tels que définis dans le contrat.

Points secondaires du contrat :¹³ intérêts à 5% sur base 30/360, modalités de paiement par le prêteur, interdiction de remboursement anticipé par l'emprunteur, conditions particulières d'exigibilité avant le terme.

2. Aspects de forme

a) Éléments d'interprétation

On peut ranger dans cette catégorie le titre du contrat et son préambule, qui fournissent des indices quant à la qualification voulue par les parties. Dans la mesure où le contenu du contrat est clair, ils n'ont aucune autre fonction que

¹³ Il s'agit ici de relever les points objectivement secondaires, sans lesquels les autres éléments peuvent suffire à caractériser un contrat de prêt conforme au droit suisse. Il va de soi que certains de ces points peuvent par ailleurs être d'une importance subjective telle qu'ils pourraient constituer des éléments principaux de la relation contractuelle.

de fournir une étiquette facilement reconnaissable par le lecteur ainsi que du contexte utile à la compréhension de l'opération envisagée par le contrat.

b) Éléments structurants

On peut ranger dans cette catégorie le titre et la numérotation des dispositions. On remarquera également qu'il est usuel pour les clauses contractuelles de suivre un certain ordre selon le type de contrat. En principe, le titre des dispositions comme leur ordre n'ont cependant aucun effet contraignant, même s'ils peuvent servir d'outils d'interprétation si le contenu du contrat est ambigu.

c) Éléments de rédaction

On peut ranger dans cette catégorie le style de langage, mais également l'utilisation de termes définis par l'utilisation de majuscules.

Il est important de noter que les termes définis en majuscules dans le contrat répondent à une simple exigence d'économie de rédaction. Bien que la notion s'approche de la définition d'une « variable » au sens informatique du terme (soit, pris largement, l'association d'un identifiant spécifique à une valeur ou à un sens définis), nous verrons par la suite que les éléments du contrat qui seront définis comme des variables dans une approche informatique sont différents.

C. Déconstruction informatique du contrat

Dans cette partie, nous examinerons comment un contrat de prêt se déconstruit logiquement afin d'en tirer quelques algorithmes simplifiés permettant de schématiser une approche informatique du contrat (infra III.C.1), puis nous déconstruirons le contrat pour en examiner chaque aspect dans une perspective informatique (infra III.C.2). Enfin, nous donnerons un exemple de « traduction » de notre contrat en langage informatique simplifié afin de rendre compte de la transformation (infra III.C.3).

1. Analyse algorithmique du contrat de prêt

Selon l'Encyclopedia Universalis, « [l']objet de l'algorithmique est la conception, l'évaluation et l'optimisation des méthodes de calcul en mathématiques et en informatique. Un algorithme consiste en la spécification d'un schéma de

calcul, sous forme d'une suite d'opérations élémentaires obéissant à un enchaînement déterminé »¹⁴.

Cette définition relativement étroite doit aujourd'hui être élargie pour couvrir l'élaboration de tout ensemble de règles qu'une personne ou une machine (et tout particulièrement un ordinateur) peut suivre afin d'atteindre un résultat constant à partir de données déterminées.¹⁵

La particularité de l'algorithme - comme descripteur d'un processus - et son principal intérêt tiennent au fait que l'algorithme est conceptuellement indépendant de la machine qui l'exécute, et du langage dans lequel il est traduit. Une approche algorithmique d'une relation contractuelle implique ainsi d'exprimer sous forme aussi abstraite que possible les diverses étapes du contrat.

La formalisation d'une opération de prêt sous forme algorithmique simple donne ce qui suit :

- (1) *A et B se mettent d'accord pour que A prête une somme d'argent à B.*
- (2) *A verse de l'argent à B.*
- (3) *B verse la même somme à A, augmentée le cas échéant d'un montant d'intérêts.*

L'étape (1) correspond à la formation de l'acte générateur d'obligation. Pour que cet accord soit pertinent d'un point de vue informatique, son contenu doit correspondre à un contenu exécutable.

L'étape (2) correspond dans notre exemple à l'exécution par A d'une obligation de paiement exprimé de manière inconditionnelle.

L'étape (3) correspond dans notre exemple à l'exécution par B d'une obligation de paiement sujette à la réalisation des conditions alternatives suivantes :

- survenance d'un événement intrinsèquement lié à la nature du contrat par un effet juridique (fin du contrat, soit la liquidation de la relation) ; ou
- survenance d'un événement défini contractuellement mais dont l'occurrence est incertaine (événement déclencheur) ; ou

¹⁴ COLLARD/FLAJOLET.

¹⁵ Cf. <https://www.merriam-webster.com/dictionary/algorithm> pour une approche beaucoup plus large (consulté le 28.09.2020) : « *The current term of choice for a problem-solving procedure, algorithm, is commonly used nowadays for the set of rules a machine (and especially a computer) follows to achieve a particular goal. It does not always apply to computer-mediated activity, however. The term may as accurately be used of the steps followed in making a pizza or solving a Rubik's Cube as for computer-powered data analysis* ».

- survenance d'un évènement défini contractuellement mais dont l'occurrence est certaine (date).

Il est évident que chacune de ces descriptions d'étape souffre d'un niveau de généralité trop élevé pour être utile d'un point de vue informatique et réduite à une série d'instructions exécutables par une machine. Un premier affinage de chaque étape pourrait prendre la forme suivante :

- (1) *A et B se mettent d'accord pour que A prête une somme d'argent à B :*
 - (1.1) *A offre de prêter de l'argent à B.*
 - (1.2) *B accepte l'offre de A.*
- (2) *A verse de l'argent à B :*
 - (2.1) *A vérifie que son compte contient suffisamment d'argent.*
 - (2.2) *A verse l'argent sur le compte renseigné par B.*
- (3) *B rembourse la même somme à A, augmentée le cas échéant d'un montant d'intérêts :*
 - (3.1) *Le contrat arrive à sa fin :*
 - (3.1.1) *Une des parties met un terme au contrat.*
 - (3.1.2) *Une condition précisée dans le contrat déclenche un versement.*
 - (3.1.3) *Le contrat arrive à son échéance ordinaire.*
 - (3.2) *Les intérêts sont calculés sur la base de la date de versement (2.2) et de la date de remboursement (3.1).*
 - (3.3) *B vérifie que son compte contient suffisamment d'argent.*
 - (3.4) *B verse l'argent sur le compte renseigné par A.*

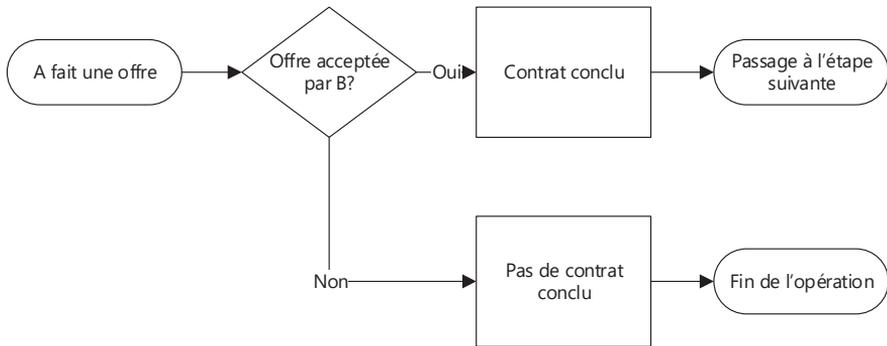
On voit déjà sur cette base que l'algorithme est incomplet : en particulier, les étapes 1.2 (B accepte l'offre de A), 2.1 (A vérifie que son compte contient suffisamment d'argent) et 3.3 (B vérifie que son compte contient suffisamment d'argent) peuvent mener à une impasse dans le monde réel dont il n'est pas rendu compte dans cette description.

En d'autres termes, si l'offre n'est pas acceptée, ou si l'une des parties n'a pas l'argent nécessaire au moment de procéder à un paiement, l'algorithme tel que décrit doit prendre fin – ou on doit passer à un autre algorithme.

Si une description linéaire de ce type est déjà utile pour donner une idée d'une approche purement logique du contrat, une autre approche consiste à séparer les décisions (vérifiées en général par une question logique) des processus à mettre en œuvre pour atteindre la finalité du contrat. Cette approche

appliquée aux grandes étapes décrites ci-dessus donnerait les organigrammes suivants¹⁶ :

- (1) A et B se mettent d'accord pour que A prête une somme d'argent à B :



L'intérêt de cette réduction du contrat à un processus est de mettre en exergue certains éléments fonctionnels qui ne sont pas nécessairement intuitifs pour le juriste.

Les points importants pour le juriste, et les éléments qui guident sa rédaction, sont les parties au contrat, la définition de l'objet du contrat, et la description des obligations qui le composent.

La réduction à l'algorithme met plutôt l'accent sur l'entrée d'information initiale, la sortie d'information finale, après passage par des conditions factuelles à vérifier et l'exécution des sous-processus qui en résultent.

D'un point de vue juridique, le schéma présenté reste incomplet car l'acceptation ou la non-acceptation d'une offre par B peut prendre plusieurs formes :

- Acceptation expresse : il s'agit de la forme décrite dans le schéma présenté.
- Acceptation tacite (art. 6 CO) : en principe, le fonctionnement du *smart contract* ne permet pas d'acceptation « tacite ».
- Non-acceptation expresse : il s'agit de la forme décrite dans le schéma présenté.
- Non-acceptation avec contre-offre : cette forme n'a pas de description formelle différente du schéma présenté. La contre-offre constituant

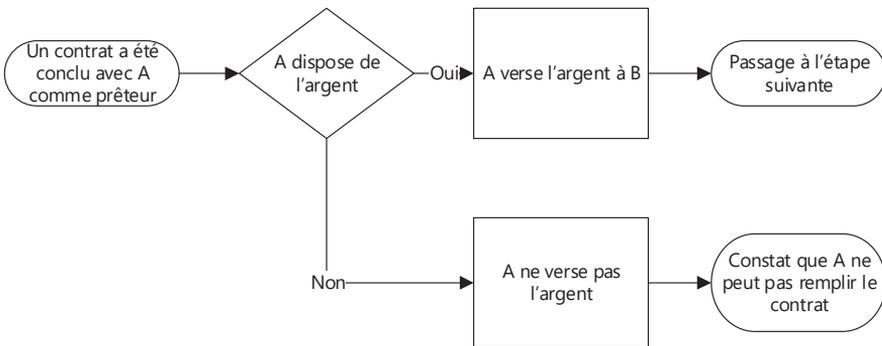
¹⁶ Nous adoptons ici le modèle de visualisation reconnu consistant à utiliser les symboles suivants : ovales pour début et fin de processus, losanges pour choix/questions à résoudre, et rectangles pour indiquer une lecture ou « écriture » d'information (soit un changement d'état). Il s'agit d'une version très simplifiée du modèle ISO 5807 : 1985, résultant lui-même de standards développés dans les années 60 par des ingénieurs, mais d'ordinaire peu utilisés (pour l'instant ?) par les juristes.

juridiquement une offre propre, il suffit d'échanger les noms des parties dans le schéma pour obtenir le résultat voulu.

- Absence d'acceptation, ou non-acceptation par silence : tant que l'offre n'est pas acceptée, il n'y a pas de contrat conclu. La question se pose donc de la continuité de la validité de l'offre. Cette question est réglée dans la loi aux articles 3 à 5 CO. En principe, des mécanismes similaires peuvent être mis en place dans le cadre de *smart contracts* (par exemple, délai pour acceptation faute de quoi l'offre devient caduque). Une description formelle de ces mécanismes est toutefois sans grand intérêt dans le contexte qui nous occupe.

D'un point de vue informatique, le passage à l'étape suivante comme dernier stade dans ce schéma ne rend pas compte du fait que la conclusion du contrat implique la formalisation sous forme informatique des algorithmes suivants dans le système (soit, à l'issue de l'étape 1, les étapes 2 et 3 sont entièrement absorbées dans le contrat conclu). Alors que l'accord « juridique » n'implique pas nécessairement la production d'un contrat au sens physique (sous réserve d'exigences de forme à respecter), l'accord « informatique », lui, nécessite clairement la production d'un programme déterministe, sans quoi le *smart contract* perd toute utilité.

- (2) A verse de l'argent à B :

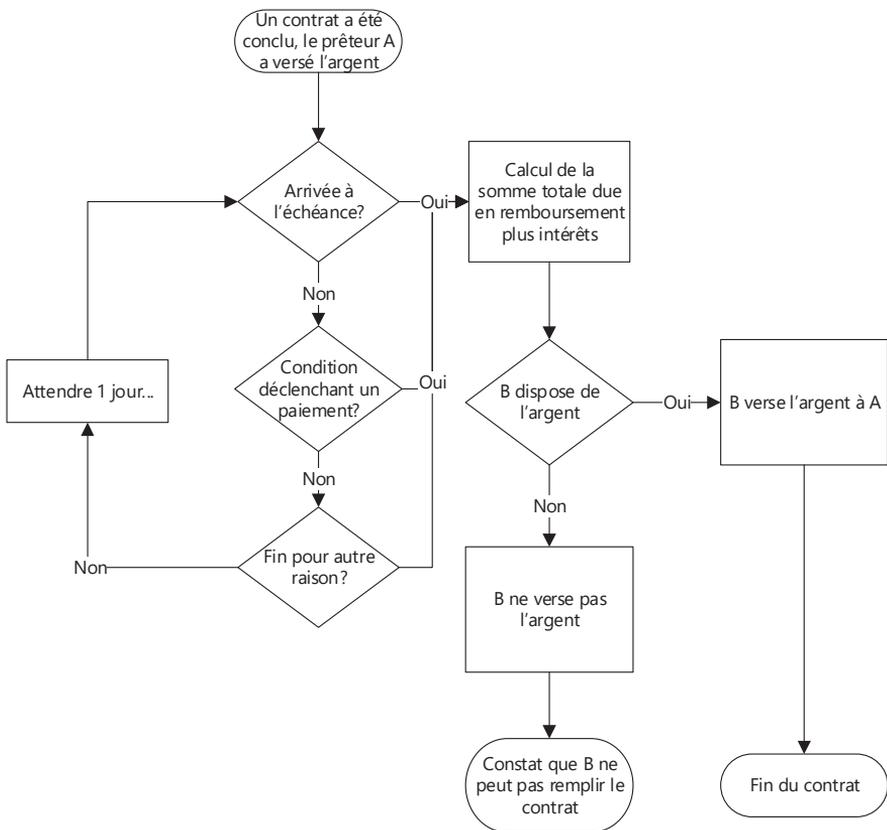


Le schéma de l'étape (2) est formellement identique au schéma de l'étape (1), comportant une seule condition à vérifier, qui permet ou non d'obtenir le résultat voulu. Cette identité des schémas cache toutefois une différence fondamentale qui tient à la nature de la condition : alors que la vérification de la condition dans l'étape (1) (soit, l'acceptation ou non de l'offre) dépend d'une manifestation de volonté qui est entre les mains d'une partie, la vérification de la condition dans l'étape (2) (soit, A dispose de l'argent) dépend d'une information externe aux parties. Il faut en effet que la machine puisse vérifier si A dispose de l'argent, indépendamment d'une information fournie par A elle-même.

Cela permet déjà de poser la question critique de la vérification des informations fournies à la machine. Alors que l'information à vérifier à l'étape (1) est considérée comme « vraie » lorsqu'elle émane d'une partie¹⁷ (information subjective), l'information à vérifier à l'étape (2) sera considérée comme « vraie » lorsqu'elle est sous le contrôle de la machine elle-même, ou lorsqu'elle émane d'une source à laquelle la machine « fait confiance » et dont le contenu est arbitrairement toujours « vrai » pour la machine.

Le schéma suivant permet d'illustrer ce problème de manière encore plus précise :

- (3) B rembourse la même somme à A, augmentée le cas échéant d'un montant d'intérêts



¹⁷ Avec ce que cela implique en termes de vérification de l'identité des parties dans un environnement informatique. Dans beaucoup d'activités de nature financière, l'identification de personnes à distance est un problème récurrent... et sujet à une réglementation qui risque bien de freiner l'adoption des technologies décrites (cf. p.ex. Circulaire 2016/7 de la FINMA, Identification par vidéo et en ligne).

Pour que le contrat puisse véritablement devenir un *smart contract*, il faut que chaque condition de l’algorithme puisse être vérifiée par la machine, et que chaque processus soit exécutable par la machine. Alors que cette vérification est aisée pour certaines conditions, car objectives ou déterminables par la machine exécutant le *smart contract* (par exemple : la date pour vérifier l’échéance du prêt), la vérification d’autres conditions nécessite que la machine ait l’accès à des informations externes : typiquement, le montant d’un compte en banque ou la survenance d’un évènement auquel il est fait référence dans le contrat.

De même, l’exécution des processus peut être le fait d’un mécanisme purement interne à la machine et donc simple (par exemple, calcul d’un montant d’intérêts), mais peut être plus compliqué lorsqu’il s’agit d’influer sur le monde externe (virement d’une somme d’argent).

Il est donc important de distinguer quels éléments – requêtes d’information et processus – sont internes, et quels éléments sont externes au contrat si on veut déterminer si le passage d’un contrat à un *smart contract* est possible, et si oui de quelle manière y arriver.

2. *Déconstruction du contrat exemple*

Dans cette partie, nous déconstruisons le contrat qui nous sert d’exemple en examinant chacune de ses parties sous un angle mécanique, sans égard au contenu matériel.

a) *Éléments de contexte*

Le contrat présenté comporte des éléments de contexte utiles à la compréhension, mais sans autre impact. Les parties suivantes du contrat de base relèvent du contexte dans lequel ce dernier est signé, sans pour autant faire partie de l’algorithme. Ils sont en effet parfaitement superflus à la mise en place d’un *smart contract* qui rendrait complètement compte du contrat et en serait une retranscription fidèle dans un environnement informatique :

Loan Agreement

[...]

Preamble

The Borrower is a company active in [business]. The Borrower has asked the Lender to grant a loan for an aggregate amount of [currency] [amount], which the Lender has accepted pursuant to the terms of this loan agreement (the Agreement).

La seule utilité de ces termes réside dans le contexte d'un litige, soit lorsque le contrat ou son exécution s'avère défectueux. Ils peuvent alors servir d'éléments d'interprétation.

Dans le développement d'un *smart contract*, il s'agirait de textes d'accompagnement servant à expliquer l'intention originelle des parties, dans la mesure où cette intention peut être difficile à déceler à la simple lecture du code informatique (cf. infra pour un exemple de simili-code montrant cette opacité).

En langage de développement, il s'agit de « commentaires ». Loin d'être inutile, l'utilisation de commentaires dans du code source est considérée comme une bonne pratique. A noter que le compilateur (soit le programme qui transforme le code source écrit en langage lisible par un humain en un code compréhensible et exécutable par l'ordinateur) ignore en principe les commentaires.

b) *Faits internes au contrat*

Le contrat comporte un certain nombre d'informations fournies par les parties elles-mêmes :

dated [date]
[name], [address] (the Lender)
[name], [address] (the Borrower)
Principal amount of CHF [amount]
Interest rate of 5%
The Loan is granted until [date]

Ces informations sont prises comme les données de base du contrat, et ne sont en principe pas vérifiées par le *smart contract* sauf pour s'assurer qu'elles répondent aux paramètres établis (par exemple, validité de la monnaie indiquée, ou validité de la date).

Les informations qui ne sont pas utiles à l'exécution d'une fonction quelconque dans le contrat (par exemple, le titre du document) ne constituent que du contexte ou des commentaires. Ainsi, le domaine d'activités de l'emprunteur est décrit dans le préambule, mais n'est utilisé dans aucune fonction du contrat présenté. Dans un autre contrat de prêt, le domaine d'activités pourrait être tout à fait pertinent (par exemple, un changement de domaine d'activités pourrait devenir la base d'une clause contractuelle prévoyant un remboursement anticipé du montant prêté). Dans un tel cas, les éléments de description deviendraient des faits contractuels.

Il convient donc de s'interroger sur l'utilisation de chaque élément factuel présenté dans le contrat afin de déterminer s'il s'agit d'un élément fonctionnel ou purement descriptif.

c) *Faits externes au contrat*

Le contrat se réfère à des faits dont la vérification nécessite le recours à une source d'information externe.

the Borrower has a resolution passed for its winding-up

the Borrower suspends all of its business operations

Les faits externes qui doivent être vérifiés par le *smart contract* sont parmi les plus importants qui, en pratique, vont établir les limites à ce qu'il sera possible ou pas d'automatiser.

Tout fait externe peut être réduit à une information numérique susceptible d'être utilisée par un *smart contract*. Il suffit en effet que l'information soit renseignée directement dans le *smart contract*, ou maintenue dans une base de données accessible au *smart contract*. La question est plutôt de déterminer qui fournit l'information ou comment elle est obtenue. Aux fins de notre analyse, il suffit de relever à ce stade que si l'information est objectivement déterminable (par exemple : faillite d'une société qui résulterait d'une inscription au registre du commerce), on a de bonnes chances de trouver une solution viable dans un environnement informatique.

Par contre, si l'information nécessite une appréciation ou un jugement (par exemple : exigence de « *best efforts* » dans le cadre d'une obligation), il sera vraisemblablement impossible d'obtenir une solution satisfaisante et le contrat ne sera pas susceptible d'être réduit à un *smart contract*.

La disponibilité de sources d'informations objectives externes (ou « *oracles* ») sera donc déterminante pour la mise en place de *smart contracts* utiles. L'informatisation progressive d'un grand nombre d'informations pertinentes pour l'exécution des contrats ainsi que l'avènement de la *blockchain* avec ce qu'elle implique en termes de mise à disposition en ligne de ressources et autres sources d'information sont évidemment de nature à faciliter l'exécution des *smart contracts*, et expliquent également pourquoi le sujet des *smart contracts*, auparavant marginal, est devenu d'intérêt plus général.¹⁸

¹⁸ MÜLLER, N 8, remarque avec raison que si la *blockchain* « *n'est pas une condition sine qua non pour que des « Smart Contracts » puissent s'exécuter, elle permet à ces derniers de réaliser leur plein potentiel* ».

d) *Règles externes*

Le contrat comporte certaines références à des règles « externes » au contrat, en ce sens qu'elles impliquent le recours à des algorithmes non décrits dans le contrat lui-même.

Certaines de ces références sont explicites et faciles à détecter, comme la clause d'élection de droit :

3.1. Governing Law. *This Agreement shall be governed by and construed in accordance with the laws of Switzerland.*

D'autres sont moins faciles à détecter pour le rédacteur, s'agissant de règles bien connues mais non juridiques :

[...] *interest at the rate of 5% on a 30/360 basis.*

En effet, le passage souligné constitue une référence à une règle de calcul d'intérêts connue du marché, mais non explicitée dans le contrat lui-même. Il sera donc nécessaire d'avoir recours à des règles ou à un processus externes au contrat pour effectuer le calcul.

D'autres encore sont bien de nature juridique, mais implicites :

– [...] *the Lender hereby grants to the Borrower a loan [...]*

En qualifiant la relation de « prêt » dans un document juridique, ce passage se réfère implicitement – dans le contexte du droit élu – à des dispositions légales spécifiques (en l'occurrence aux articles 312 à 318 CO, bien que cette référence elle-même résulte d'autres éléments à prendre en compte afin de se déterminer plus précisément sur la nature du prêt, entre prêt à usage et prêt de consommation).

En d'autres termes, une description complète des fonctionnalités du contrat nécessite l'importation d'un certain nombre de règles de l'extérieur, le contrat ne se suffisant pas à lui-même.

e) *Effets et processus contractuels*

Les parties en apparence les plus importantes du contrat, dans une perspective de transformation en un *smart contract* permettant une exécution sans heurts, seraient évidemment celles décrivant les processus contractuels et effets principaux d'un point de vue obligationnel.

On distingue tout d'abord dans cette catégorie les obligations affirmatives, c'est-à-dire nécessitant une action :

– *the Lender shall disburse the Principal Amount*

– [...] *the Principal Amount and all accrued interest under the Agreement shall be immediately due and payable [...]*

- *the Loan, including any interest, shall be reimbursed on the Maturity Date*

Ces obligations constituent le cœur du *smart contract*, dont la raison d'être est d'assurer leur exécution. Chacune de ces obligations implique, dans le monde réel, que son destinataire entreprenne une certaine action (en l'occurrence, il s'agit à chaque fois d'effectuer un paiement). En rendant l'exécution automatique, le *smart contract* remplace la phase d'exécution dans le monde réel et – en principe – élimine le risque de non-exécution.

Ensuite, le contrat décrit certains effets qui seraient automatiques même en dehors d'un *smart contract* :

- *The Loan, during its term, shall bear interest*

D'un point de vue informatique, ce type d'effet se confond en nature avec les obligations dites affirmatives telles que décrites plus haut. En effet, tant que l'opération est à la portée du *smart contract* il n'y a fonctionnellement aucune différence entre le virement (qui se résume à ajuster des valeurs dans deux registres) ou la comptabilisation des intérêts (qui se résume à ajuster une valeur dans un registre).

Enfin, la catégorie comprend également des obligations négatives, nécessitant une absence d'action :

- *The Loan may not be prepaid without the consent of the Lender*

A moins de permettre au *smart contract* d'influer sur un autre processus informatique (par exemple en empêchant le fonctionnement d'un mécanisme de paiement), les obligations négatives simples restent sans effet. Si le *smart contract* contient une description complète des processus, il suffit en effet de ne pas décrire un processus permettant un paiement anticipé pour atteindre le même effet que d'interdire le paiement anticipé.

Ce type d'obligation négative peut toutefois garder un intérêt lorsqu'il s'agit de contredire une règle par défaut qui pourrait sinon accidentellement être importée d'une bibliothèque externe – soit par référence au droit supplétif intégré au contrat par élection de droit.

De manière plus générale, si un ensemble de règles externes/bibliothèques de fonctions (*libraries*) décrivant la relation contractuelle de manière complète est disponible, la constitution d'un *smart contract* standardisé serait réduite à une référence aux bibliothèques pertinentes et à une description des faits uniquement – sans avoir besoin d'intégrer des fonctions propres. Il suffirait ainsi d'appeler les fonctions pertinentes au moment de l'exécution du contrat. Au vu de la pratique en matière informatique, consistant en la constitution progressive de bibliothèques standardisées comprenant un grand nombre de fonctions, il est à prévoir que les opérations économiques courantes

susceptibles de formalisation par *smart contracts* seront ainsi pré-décrites.¹⁹ La construction de *smart contracts* ad hoc et de fonctionnalités spéciales sur la base de contrats spécifiques écrits ferait figure d'exception.

f) *Autorité*

“Any dispute or difference arising out of or in relation to this Agreement shall be subject to the exclusive jurisdiction of the courts of Switzerland.”

Alors que le juriste comprend qu'une intervention judiciaire puisse être nécessaire à plusieurs niveaux, ce passage est très difficile à conceptualiser d'un point de vue informatique.

Le juriste vit en effet dans un monde dans lequel la principale fonction du juge dans le contexte contractuel est d'assurer l'exécution : d'abord en déterminant qui a raison dans un cadre litigieux (fonction également assurée, le cas échéant, par l'arbitre), et ensuite en déterminant les conditions de mise en œuvre de la force public que si, en dépit de cette détermination, il s'avère nécessaire d'avoir recours à des mécanismes d'exécution forcée. Or, le *smart contract* vise justement à éviter cette intervention en assurant que le mécanisme d'exécution soit intégré au contrat.

Ensuite, le juriste peut concevoir un désaccord sur des éléments de fait (par exemple, réalisation ou non d'une condition matérielle) ou un désaccord sur des éléments de droit (conséquences juridiques d'une situation non réglée exhaustivement dans le contrat). Or, le *smart contract* n'accepte aucune incertitude factuelle : dans une logique binaire, il doit être renseigné de manière certaine sur la réalisation d'une condition sous une forme objective²⁰, sans quoi il perd toute raison d'être et la question ne se pose plus. Pour le surplus, le *smart contract* n'admet aucun désaccord sur une question de droit. A compter qu'il soit bien codé, le *smart contract*, en tant qu'algorithme décrivant complètement une opération, se doit d'être déterministe et ne laisser aucune place à l'incertitude.

Est-ce à dire qu'il n'y a plus de place pour une simple éléction de for dans un *smart contract*, voire même qu'il n'y a plus de place pour un juge tout court ? Assurément non. Comme nous le verrons plus loin, le *smart contract* érigé comme réponse unique aux défauts d'exécution présente des risques très importants.²¹

Le mode d'intervention potentielle d'un système de justice nécessite une réflexion plus approfondie qu'une simple transcription sous forme de code

¹⁹ Voir p.ex. le projet Cicero qui vise à créer un tel catalogue (<https://github.com/accordproject/cicero>, consulté le 28.09.2020).

²⁰ Cf. *supra* III.C.2.c) concernant les faits externes au contrat.

²¹ Cf. *infra* IV.

d'une obligation exécutable par une partie. En effet, l'architecture même de la *blockchain* sur laquelle le *smart contract* est bâtie ne permet pas d'intervention externe simple lorsqu'il s'agit de corriger une situation. Sans mise en place d'une fonction lui permettant d'intervenir sur le système, le juge n'aurait ainsi pas les moyens matériels de modifier la *blockchain* sur laquelle le *smart contract* litigieux est construit. La mise en place d'une telle fonction, quant à elle, pourrait potentiellement créer des failles de sécurité – sans parler de la remise en question de la philosophie même qui sous-tend les *smart contracts* et qui est de se passer, justement, du juge.

En somme, une réflexion très approfondie s'impose sur la relation entre la justice et les mécanismes automatisés dans un environnement *blockchain* si l'on veut concevoir une utilisation généralisée de *smart contracts* non standardisés.²² Le risque maximal sera en effet atteint pour des *smart contracts* visant une retranscription sous forme électronique/sur *blockchain* d'une relation juridique négociée *ad hoc*.

3. *Reconstruction du contrat sous forme de code*

Enfin, nous présentons ci-dessous un exemple de simili-code pour donner une idée de ce à quoi notre contrat exemple pourrait ressembler une fois passé entre les mains d'un informaticien. A noter que l'exemple fourni présente des similitudes avec certains langages de programmation connus, mais est conçu uniquement comme exemple abstrait.

```
import "Swisslaw.lib"; import "interest.lib"  
contract loanoffer {  
    int loanamount  
    date maturity  
    date offerduration  
    function acceptloan() {  
        address borrower = caller.address  
        move (loanamount, contract.owner)
```

²² Cf. dans ce sens FAVRE-COUNE/BELET qui s'interrogent sur l'utilité de la clause compromissoire dans le cadre du *smart contract* et examinent la question épineuse de la validité d'une telle clause ainsi que des mécanismes de mise en œuvre. Les auteurs conçoivent ainsi d'une « *partie codée de la convention d'arbitrage* » qui « *permettrait à l'arbitre de modifier le code, lui conférant la possibilité d'interrompre l'exécution du contrat durant la procédure arbitrale et, par la suite, de le modifier conformément à sa sentence* » (p. 1114).

```

    date loan.disbursement = today
    event msg.contract.owner("Your loan offer has been accepted by"
    + caller.address)
    }
function reimburse(contract.owner) {
    if today.date >= (loan.disbursement + loan.maturity):
        move (loanamount + interest360actual(interest, today.date),
        borrower)
    elseif borrower == bankrupt AND msg.borrower == "pay me
    back":
        move (loanamount + interest306actual(interest, today.date),
        borrower)
    }
    event
    if today.date >= (contract.date + offerduration) AND acceptloan()=
    NULL:
        destruct
}

```

On notera les références aux sources externes d'information sous forme de bibliothèques importées (en particulier s'agissant des fonctions nécessaires de droit suisse et de calcul d'intérêts), puis une définition d'offre de contrat de prêt qui comprend un montant, une durée, une durée d'offre, deux fonctions réglant respectivement l'acceptation du contrat et le remboursement, et une dernière ligne permettant de faire disparaître l'offre (destruction du *smart contract*) en l'absence d'une acceptation dans le délai déterminé.

Etant donné que l'avenir du *smart contract* ira vraisemblablement de pair avec une standardisation, il est à prévoir que les fonctions essentielles – soit celles décrites sous *function acceptloan()* et *function reimburse(contract.owner)* – seront définies dans des bibliothèques à libre disposition. On arrivera donc probablement à un *smart contract* plus proche de la forme suivante une fois que notre hypothétique bibliothèque « *Swisslaw.lib* » aura été complétée avec les fonctions idoines :

```

import "Swisslaw.lib"
contract loanoffer {
    self.type = "loan"
    self.loanduration.fixed = TRUE
    self.interestbasis = "360actual"
}

```

Ce code définirait – toujours de manière théorique et simplifiée, bien entendu – un contrat de type « prêt » dont les caractéristiques découleraient de la librairie « *Swisslaw.lib* ». Les seuls éléments supplémentaires à définir seraient donc ceux qui s'écarteraient des valeurs par défaut fournies par la bibliothèque. En imaginant que les valeurs par défaut seraient, par exemple, que le contrat de prêt est sans échéance de remboursement particulière, qu'une offre est toujours limitée dans le temps, et que les prêts portent intérêts à un taux actuel/365, notre *smart contrat* devrait prévoir, par exception, une durée fixe de contrat et un taux actuel/360 pour se conformer à notre exemple.

Tout cela part évidemment du principe que des juristes et des ingénieurs seront capables de produire et de documenter des modèles de *smart contracts* utilisables à grande échelle. Le mouvement a déjà commencé sous forme d'initiatives privées, en particulier aux États-Unis. Face à ce dynamisme du secteur privé, le législateur peine à trouver ses marques et se concentre aujourd'hui essentiellement sur la gestion des conséquences d'une transcription en environnement *blockchain* de concepts juridiques existants.

Bien loin d'amener une révolution, la volonté des acteurs économiques de profiter des nouvelles technologies n'aura fait que mettre les forces publiques face au problème de la numérisation, pourtant déjà bien entamée, sur fond de *blockchain*. La neutralité technologique dans ce contexte est relativement facile à prôner²³, puisqu'il est uniquement question, dans la vaste majorité des cas,

²³ Voir p.ex. Rapport du Conseil fédéral du 14 décembre 2018, Bases juridiques pour la *distributed ledger technology* et la *blockchain* en Suisse - État des lieux avec un accent sur le secteur financier, p. 14 : « Approche fondamentalement neutre sur le plan technologique : en matière de législation et de réglementation, la Suisse doit continuer à suivre une approche basée sur des principes et neutre sur le plan technologique, tout en permettant les exceptions nécessaires. Les règles créées doivent être aussi neutres que possible du point de vue de la concurrence. Les dispositions légales ne doivent pas viser certaines technologies en particulier » (souligné dans l'original). On remarquera que le projet de Loi fédérale sur l'adaptation du droit fédéral aux développements de la technologie des registres électroniques distribués dans sa version soumise à consultation le 22 mars 2019 contenait de nombreuses références à la technologie des registres distribués (TRD) dans les modifications proposées au code des obligations, alors que le projet adapté par le Conseil fédéral le 27 novembre 2019 à l'issue de la consultation et adopté par l'Assemblée fédérale le 10 septembre 2020 (FF 2020 319) en est élagué... et bel et bien technologiquement neutralisé en grande partie, même si les références à la TRD restent très présentes dans les modifications proposées aux autres lois et en particulier dans la Loi sur l'infrastructure des marchés financiers qui accueillerait ainsi une nouvelle catégorie d'autorisation ad hoc. Le rapport du DFF du 27 novembre 2019 sur les résultats de la consultation relative à la loi fédérale sur l'adaptation du droit fédéral aux développements de la technologie des registres électroniques distribués établit en p. 6 : « *Dans le droit des marchés financiers, le manque de neutralité du projet sur le plan technologique a également été critiqué par*

de remplacer un mode de communication (écrit physique) par un autre (code) – l'opération restant par ailleurs fondamentalement la même que si le code n'existait pas. La discussion porte sur les questions de sécurité dans un combat non déclaré entre ingénieurs qui promeuvent la sécurité cryptographique du nouvel outil électronique et les juristes, qui défendent la sécurité juridique d'un système construit sur des outils analogiques qui aurait fait ses preuves depuis plus de cent ans.

Il ne s'agit toutefois pas d'un débat, mais plutôt d'une ligne de front : le but déclaré des milieux favorables à la mise en place de *smart contracts* à grande échelle est bien d'échapper aux aléas du système juridique, et non de convaincre les juristes d'adopter de nouveaux outils²⁴. Nous verrons dans le chapitre suivant que cette approche comporte des réels dangers.

IV. Exécution de la volonté : le *smart contract* entre hyper-efficacité réelle et inefficacité juridique ?

L'exemple type de l'expérience qui a mal tourné en matière de *smart contracts* déployés à grande échelle, et certainement le mieux connu, est celui du « DAO hack ».

En 2016, Christoph Jentzsch lança une organisation décentralisée ou *decentralised autonomous organisation*, plus communément connue sous le

divers participants. Cependant, certains participants concèdent également qu'il est impossible de s'en tenir au principe de la neutralité technologique dans le droit des infrastructures des marchés financiers (SBF [à laquelle TI, BAS, DS et Swisscom se rallient]) ».

²⁴ Cela étant, la nouvelle Loi fédérale sur l'adaptation du droit fédéral aux développements de la technologie des registres électroniques distribués réserve des surprises conceptuelles intéressantes. En premier lieu, le « droit-valeur inscrit » au sens du nouvel art. 973d al. 1 CO nécessite une « convention entre les parties » portant sur l'inscription dans un registre des droits-valeurs au sens de l'art. 973d al. 2 et sur l'impossibilité de faire valoir ou transférer ces droits autrement que par le biais du registre. Cette convention prendrait normalement une forme tout à fait classique qui ne présumerait en rien du contenu d'un éventuel *smart contract* sous-jacent... si ce n'était pour l'art. 973d al. 2 ch. 1 CO qui prévoit que le registre en question « [...] donne aux créanciers, mais non au débiteur, le pouvoir de disposer de leurs droits au moyen de procédés techniques ». Un *smart contract* donnant des pouvoirs de disposition à une seule des parties – et de surcroît au créancier –, serait donc un élément sine qua non de la mise en place d'un registre répondant aux exigences du CO afin de servir de point d'accroche à des droits-valeurs inscrits. Dans cette approche, l'existence de *smart contracts* spécifiques devient un élément nécessaire du fonctionnement du système juridique lui-même.

nom « the DAO », sur la *blockchain* Ethereum.²⁵ L'objectif du DAO était de servir de fonds capital risque auto-réglementé, sans gouvernance autre que le code des *smart contracts* qui le constituaient. Pour affranchir complètement le fonds de facteurs humains, le DAO fut créé sans direction ni conseil d'administration au sens classique, visant une gouvernance décentralisée et un écosystème dans lequel il n'est plus nécessaire aux parties d'avoir confiance les unes en les autres (*trustless*), cette confiance étant assurée par le *smart contract* lui-même, qui réglait également, sans intervention humaine, le choix et les modalités des investissements à mener par le DAO.²⁶ Le code informatique du DAO était ainsi élevé au statut de « loi », par l'adoption par tous les participants aux conditions d'utilisation suivantes :

« *The terms of The DAO Creation are set forth in the smart contract code existing on the Ethereum blockchain at 0xbb9bc244d798123fde783fcc1c72d3bb8c189413. Nothing in this explanation of terms or in any other document or communication may modify or add any additional obligations or guarantees beyond those set forth in The DAO's code. Any and all explanatory terms or descriptions are merely offered for educational purposes and do not supersede or modify the express terms of The DAO's code set forth on the blockchain; to the extent you believe there to be any conflict or discrepancy between the descriptions offered here and the functionality of The DAO's code at 0xbb9bc244d798123fde783fcc1c72d 3bb8c189413, The DAO's code controls and sets forth all terms of The DAO Creation* »²⁷.

Autrement dit, en cas de conflit entre le code et les conditions juridiques d'utilisation, le code prévaut – « *the code is the law* ». Le contenu du « contrat » au sens juridique serait donc aligné par définition ou, du moins, par consentement, avec le contenu du *smart contract* sous-jacent.

En juin 2016, une personne anonyme exploita le code afin d'obtenir un paiement indu de plus de USD 50 millions, laissant la communauté confrontée au dilemme de déterminer s'il s'agissait d'un « *hacker* » (ce qui impliquerait que la personne ait commis un acte de piratage en contournant une protection logicielle ou physique du système informatique), d'un voleur (ce qui impliquerait que le détournement constituerait une forme de fraude même s'il était permis par le *smart contract*), ou d'un bon citoyen (ce qui impliquerait

²⁵ Pour une description plus complète des événements qui suivent, cf. DUPONT.

²⁶ MORRISON et al., p. 3.

²⁷ Anonyme (2016), « An Open Letter to the DAO and the Ethereum Community », disponible en ligne : <https://pastebin.com/CcGUBgDG> (consulté le 29.06.2020). Avec une certaine ironie, l'auteur – qui signe « the Attacker » (entre guillemets) – remercie dans ce courrier la communauté Ethereum, affirme avoir consulté des avocats sur la légalité de ses actions, menace de poursuites judiciaires ceux qui voudraient s'en plaindre, et souhaite bonne chance à la communauté.

qu'il suffit qu'un acte soit permis par le *smart contract* pour, *ipso facto*, être conforme à l'intention des parties).

Alors que l'étiquette « *hacker* » a été rapidement écartée – en raison notamment du fait que la manipulation n'a nécessité rien de plus qu'une utilisation, à la lettre, des possibilités offertes par le *smart contract* et qu'une partie de la communauté (auto proclamée « Robin Hood ») a elle-même recouru à cette même fonctionnalité pour récupérer une partie des *ether* détournés – le débat a été moins facile à résoudre sur la question de savoir si l'acte devait être corrigé ou admis. Une « correction » – soit l'annulation des actions de l'auteur anonyme de la prouesse – nécessitait en effet une remise en question en apparence fondamentale du credo « *the code is the law* », alors que l'admission de l'acte revenait à cautionner ce qui était clairement un comportement indésirable aux yeux de la majorité des acteurs du système – en particulier auprès de ceux qui avaient investi des *ether* dans l'aventure.

Après des semaines d'intenses débats, au cours desquelles la communauté s'est déchirée,²⁸ la question a trouvé une réponse radicale sous la forme d'un *hard fork* : création d'une copie du système sous-jacent dans laquelle le paiement a été maintenu (aujourd'hui connu sous le nom d'Ethereum Classic), et d'une copie « corrigée » dans laquelle le paiement n'existe tout simplement plus (qui continue pour la majorité sous le simple nom d'Ethereum).

La solution n'est évidemment pas satisfaisante. Premièrement, elle implique une remise en question fondamentale de l'axiome sur lequel le système était bâti (« *the code is the law* ») sous couvert de démocratie tournée en dictature de la majorité.²⁹ Deuxièmement, un système dans lequel chaque exploitation malintentionnée du code nécessiterait l'intervention de la communauté entière, suivie d'une scission de cette même communauté, n'est pas viable.

Quels sont les enseignements à tirer de ce qui fut la première expérience de mise en pratique à grande échelle de la théorie des *smart contracts* ?

Tout d'abord, un *smart contract* est constitué de lignes de code informatique – comme un logiciel. Si le code contient une erreur, le *smart contract* produira des résultats indésirables. Sur ce point déjà, l'implication humaine est essentielle – le code doit être pensé, conçu, construit, rédigé, testé, amélioré, audité, corrigé, et idéalement publié pour minimiser le risque d'erreurs. Une fois déployé sur un réseau distribué, plus question en général – au nom du caractère immuable de la *blockchain* – d'y toucher à nouveau. Une

²⁸ Pour plus de détails, V. p.ex. M. JACCARD, Smart contracts, dumb code.

²⁹ V. p.ex. la réaction de Christoph Jentzsch : « *Although some do question the analogy "code is law". I do not. We just found out that we have a supreme court, the community!* » (<https://blog.slock.it/what-an-accomplishment-3e7ddea8b91d>, consulté le 06.07.2020).

« correction » (*bug fixing*) équivaudrait à une modification « rétroactive » par un tiers – l’architecte du *smart contract* - de l’accord passé entre parties à ce contrat – réduisant à néant le concept même de la confiance à placer dans le système (*trustless*) plutôt qu’entre parties à un contrat ou dans un tiers médiateur. Encore aujourd’hui – plusieurs années après l’épisode du DAO hack, peu de *smart contracts* sont correctement testés, et leurs auteurs sont encore trop réticents à publier en libre accès (*open source*) et permettre une véritable vérification (*audit*) du code et de la documentation. De leur côté, les utilisateurs de *smart contracts* – et leurs conseillers juridiques – ne sont pas non plus à l’aise pour étudier les codes sources à la base des écosystèmes et des protocoles qui y seront déployés.

Mais l’intervention humaine dans le processus ne s’arrête pas là : une fois les *smart contracts* activés, l’exemple du « DAO hack » a démontré la nécessité de mettre en place un mécanisme d’arbitrage et de décision en dehors du *smart contract* en cas de problème dans l’exécution. Comment traiter d’une tentative d’extorsion, comment corriger ou même faire évoluer un *smart contract* ? Dans le cas de DAO complexes, il est probable qu’il faille repenser de fond en comble les théories de gouvernance traditionnelles et concevoir des systèmes où l’humain interviendrait à nouveau, en cas de crise majeure ou de faille avérée du réseau de *smart contracts* déployés – à la manière d’un plan de gestion de crise.³⁰

Dans l’exemple du « DAO hack », la question peut donc légitimement se poser de savoir si l’échec de l’initiative résulte de l’impossibilité de régler dans des *smart contracts* tous les aspects liés à l’exécution automatique de la volonté des parties, sans garde-fous ni gendarmes, ou s’il n’est pas plutôt le fruit de l’omniprésence de l’intervention humaine, qui a en réalité, par la décision du *hard fork*, fait dérailler le projet.³¹

Enfin, même une fois résolues ces questions épineuses, la mise en œuvre de la gouvernance en lien avec des *smart contracts* éprouvés ne sera effective que si l’ensemble de l’écosystème en reconnaît la légitimité, et notamment les acteurs principaux dans un écosystème *blockchain*. Autrement dit, il faudra convaincre les relais incontournables des registres distribués, les fameux « nodes », de souscrire aux règles établies par un DAO, sachant que leur intérêt – essentiellement financier, puisque que la validation des chaînes de bloc par

³⁰ Pour des réflexions poussées sur ces questions, V. p.ex. MORRISON *et al.* qui envisagent la mise en place d’agents logiciels automatisés, voire d’humains dont le comportement serait dicté par des *smart contracts*, au sein de « *centralized autonomous organizations* » (CAO), capables de déléguer la prise de décisions à certains éléments en cas de crise, plutôt que de laisser le *smart contract* régler toutes les questions. Voir aussi MURRAY *et al.*, *Contracting in the Smart Era*.

³¹ <https://www.wired.com/2016/06/50-million-hack-just-showed-dao-human> (*consulté le 06.07.2020*).

ces relais est en général rémunérée – n’est pas forcément aligné avec celui des promoteurs du DAO.

V. Conclusion

A ce jour, un *smart contract* n’est pas intelligent par nature, et encore moins un contrat au sens juridique. Sa capacité à correctement traiter, de façon autonome, un certain nombre d’informations selon une logique algorithmique dépend essentiellement de l’intelligence de son concepteur – à l’heure actuelle, encore un humain – et de ses talents d’informaticien quand il s’agit d’écrire le code informatique sous-jacent au *smart contract*. Quant à l’analogie avec un contrat au sens juridique, elle n’est pas correcte, car le *smart contract* peut certes concrétiser un échange de volontés concordantes et réciproques, mais ne peut pas encore le remplacer.

L’expérience de *smart contracts* qui cherchent à gérer à grande échelle ou de façon complexe des transactions entre parties qui ne se connaissent pas est aujourd’hui mitigée. Il manque à ces écosystèmes une certaine maturité, et la mise en place de règles de gouvernance appropriées – à défaut, le système se grippe et l’humain doit intervenir pour rétablir une situation qui n’avait pas été correctement anticipée par le *smart contract*.

Ne nous méprenons pas. Ces errements et ce manque de maturité seront rapidement surmontés, par un meilleur dialogue et une plus grande compréhension entre informaticiens et juristes d’abord, puis par l’établissement de standards et de meilleures pratiques qui s’imposeront peu à peu. Même aujourd’hui, dans leur forme la plus embryonnaire, les *smart contracts* sont déjà très utiles dans certaines industries, et il incombe sans délai aux juristes de modifier leur approche de la rédaction contractuelle pour intégrer la logique algorithmique, et aux informaticiens de mieux réfléchir aux scénarios d’utilisation de leur code dans le monde réel avant de déployer leurs *smart contracts* sur une *blockchain*.

VI. Bibliographie

Blaise CARRON/Valentin BOTTERON, How smart can a contract be ?, in Daniel KRAUS/Thierry OBRIST/Olivier HARI (éd.), *Blockchains, Smart Contracts, Decentralised Autonomous Organisations and the Law*, Cheltenham 2019, p.101 ss. ; **Philippe COLLARD/Philippe FLAJOLET**, Algorithmique, *Encyclopædia Universalis* en ligne (<http://www.universalis.fr/encyclopedie/algorithmique>, consulté le 26.05.2020) ; **Aline DARBELLAY/Michel José REYMOND**, Emission et négoce de jetons digitaux, in *Expert Focus* 11/17, p. 880 ; **Quinn DUPONT**, Experiments in algorithmic governance - A history and ethnography of “The DAO,” a failed decentralized autonomous organization, in

Malcolm Campbell-Verduyn (éd.), *Bitcoin and Beyond*, Londres 2017 (<https://doi.org/10.4324/9781315211909>, consulté le 29.06.2020) ; **Jana ESSEBIER/Dominic A. WYSS**, Von der Blockchain zu Smart Contracts, in Jusletter du 24 avril 2017 ; **Pascal FAVROD-COUNE/Kévin BELET**, La convention d'arbitrage dans un smart contract, in PJA 2018, p. 1105 ss ; **Philippe GILLIÉRON**, La transformation numérique du monde du droit, Genève/Zurich/Bâle 2019 ; **Jacques IFFLAND**, L'émission d'actions digitalisées comme moyen de financement de l'innovation et des PME, in RSDA 2019, p. 498 ss ; **Gabriel JACCARD**, Smart Contracts and the Role of Law, in Jusletter IT, 23 novembre 2017 (cité : G. JACCARD) ; **Michel JACCARD**, La conclusion de contrats par ordinateur – Aspects juridiques de l'échange de données informatisées (EDI), Berne 1996 (cité : M. JACCARD, La conclusion de contrats par ordinateur) ; **Michel JACCARD**, Smart contracts, dumb code ou quand le blockchain déraile, (https://www.bilan.ch/opinions/michel-jaccard/smart_contracts_dumb_code_ou_quand_le_blockchain_deraille, consulté le 29.06.2020), (cité : M. JACCARD, Smart contracts, dumb code) ; **Peter JAEGGI** in Zürcher Kommentar zum Zivilgesetzbuch, Obligationenrecht, Kommentar zur 1. und 2. Abteilung (Art. 1-529 OR), 3^e éd., vol. V/1a, Obligationenrecht Art. 1-17, Zurich 1973 (cité : ZK-JAEGGI) ; **Stephan MEYER/Benedikt SCHUPPLI**, «Smart Contracts» und deren Einordnung in das schweizerische Vertragsrecht, in Recht - Zeitschrift für juristische Weiterbildung und Praxis 2017(3) p. 204 ss ; **Robbie MORRISON/Natasha C. H. L. MAZEY/Stephen C. WINGREEN**, The DAO Controversy : The Case for a New Species of Corporate Governance ?, Front. Blockchain 3 : 25, (<https://www.frontiersin.org/articles/10.3389/fbloc.2020.00025/full>), consulté le 04.07.2020 ; **Alex MURRAY/Scott KUBAN/Matthew JOSEFY/Jonathan ANDERSON**, Contracting in the Smart Era: The Implications of Blockchain and Decentralized Autonomous Organizations for Contracting and Corporate Governance, publié en ligne le 17 avril 2019, (https://www.researchgate.net/profile/Matt_Josefy/publication/332472143_Contracting_in_the_Smart_Era_The_Implications_of_Blockchain_and_Decentralized_Autonomous_Organizations_for_Contracting_and_Corporate_Governance/links/5cb8e35c299bfl20976f8eec/Contracting-in-the-Smart-Era-The-Implications-of-Blockchain-and-Decentralized-Autonomous-Organizations-for-Contracting-and-Corporate-Governance.pdf, consulté le 06.07.2020) ; **Christoph MÜLLER**, Les « Smart Contracts » en droit des obligations suisse, in Blaise CARRON/Christoph MÜLLER (éds), *Blockchain et Smart Contracts*, 3^e journée des droits de la consommation et de la distribution, Bâle/Neuchâtel 2018, p. 51 ss ; **Nick SZABO**, A Formal Language for Analyzing Contracts, 2002 (<https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/contractlanguage.html>, consulté le 30.06.2020) (cité : SZABO, A Formal Language for Analyzing Contracts) ; **Nick SZABO**, Formalizing and Securing Relationships on Public Networks, First Monday, Vol. 2, No. 9 – 1^{er} septembre 1997 (<https://firstmonday.org/article/view/548/469>, consulté le 30.06.2020) (cité : SZABO, 1997) ; **Pierre TERCIER/Laurent BIERI/Blaise CARRON**, *Les contrats spéciaux*, Genève/Zurich/Bâle 2016.