LE CONTRAT RICARDIEN

Ian Grigg
Systemics, Inc.
iang@iang.org

(Traduit par Ines Boussoffara)

Abrégé

Décrire la valeur numérique pour systèmes de paiement n'est pas une tâche insignifiante. Les méthodes simplistes de l'utilisation des nombres ou des codes du pays afin de décrire les devises et les symboles des télex pour émettre les obligations, les actions et les instruments financiers bientôt confrontés à des lacunes par rapport à leur capacité d'endurer la divergence et dynamicité des exigences. Les variations apparemment arbitraires concernant les significations des différents instruments sont capturées en tant que contrats entre émetteurs et détenteurs. Par conséquent, l'émission numérique des instruments pourrait être considérée comme l'émission des contrats.

Le présent document traite de la question du contrat : Un formulaire du document est décrit en tant qu'un qui comprend la nature intrinsèque du contrat de l'instrument financier. Cependant, il fait face au besoin d'être une partie intégrale du système de paiement.

I. Introduction

Peu de travail a été fait concernant la classification et la description de la valeur dans de la cryptographie financière. Le présent document présente le contrat ricardien sous forme de méthode qui sert à identifier et décrire les émissions d'instruments financiers

Comme étant des contrats [1]. Il a été conçu à l'origine par Ian Grigg et Gary Howland dans le cadre du système de paiement Ricardo.

I-1 Les Origines

La demande initiale était, de fait, un système d'échange et de négociation d'obligation [2]. Quant à ce système d'échange, l'élément de base est un système de transfert ou de paiement qui reçoit et agit selon les instructions de transfert pour déplacer les éléments sous forme de liquidités et d'obligations d'un compte à l'autre. Par conséquent, chaque instruction doit identifier l'instrument.

Un moyen a été exigé afin de saisir, identifier et décrire les instruments négociés. Il existe des milliers d'obligations et potentiellement des millions d'autres instruments pouvant être

émis et échangés et chaque instrument possède des caractéristiques uniques qui ne sont pas compatibles avec la compression dans les bases de données. Selon ce système, l'argent liquide n'est pas différent des obligations et il nécessite la même description.

I-2 Le problème

Lorsque quelqu'un émet une devise (une obligation ou une action) sur internet, qu'est-ce que c'est ? Qu'est-ce que le destinataire a ?

Peu de systèmes d'émission de valeurs (systèmes de paiement) traitent adéquatement de ces questions. Ils se réfèrent généralement aux unités externes existantes de la devise mettant de l'ordre dans *une entente d'utilisation*. PayPal, par exemple, un émetteur de dollars dépend de la familiarisation avec le dollar américain ce qui définit la plupart du service qu'il offre. Les émetteurs de l'or s'appuient fortement sur leurs ententes de l'utilisateur car l'unité métallique n'est pas si familière.

Quant à la négociation, il ne suffit pas de faire appel à des références familières bien connues tant que chaque instrument est différent de toutes les manières pesantes et ces différences importent aux commerçants. Mais même avec les devises, l'utilisateur éprouve des difficultés à déterminer la sécurité et la sûreté d'un dollar par rapport à un autre.

Le classement par chiffres ou symboles est un point de départ. Presque tous les systèmes d'émission numérique identifient leur problème de base en attribuant des chiffres ou des lettres comme devises (par exemple, 840, « USD », « AUG » [3]). Ces systèmes sont rapidement en difficulté.

Un émetteur avec de nombreuses devises ou de nombreux émetteurs avec la même devise nominale soulève des questions difficiles. Un émetteur peut-il avoir deux unités de dollars ou plus ? Par exemple, dans ISO3166-1, il y a trois dollars différents : 840/USD (espèces), 998/USS (même jour) et 997/USN (le jour suivant). De même, comment une monnaie d'or numérique ("DGC") différencie-t-elle son or de celui d'un autre émetteur quand tous sont connus comme « AUG » ?

I-3 La solution

Comme les obligations sont essentiellement des contrats entre émetteurs et utilisateurs, notre problème se réduit à un contrat d'émission. Alors que d'autres questions *ont* des contrats, les nôtres *sont* des contrats.

Notre innovation est d'exprimer un instrument émis comme un contrat et de lier ce contrat à tous les aspects du système de paiement. Dans ce processus, un document d'une grande utilité (lisible par l'utilisateur et le programme) est rédigé et signé numériquement par l'émetteur de l'instrument. Ce document, le Contrat ricardien, constitue la base pour comprendre un problème et chaque transaction dans ce problème.

Par extension, toutes les émissions de valeur, telles que les devises, les actions, les produits dérivés, et les systèmes de fidélisation et les bons peuvent bénéficier de cette approche.

I-4 La Structure

Le présent document est structuré comme suit. Dans la section 2, nous discutons des approches conventionnelles pour identifier et décrire l'émission et nous explorons les

questions et les doutes entourant ces approches. Ensuite dans la section 3, on présente un modèle pour exprimer l'émission sous forme de contrat. Pour finir dans la section 4, on ajoute des remarques finales.

II. Questions de valeur comme contrats :

II-1 Un régime de première génération

Prenons le cas de l'eCash, le système de trésorerie numérique pionnier, tel que présenté à l'origine par Digicash BV. La première monnaie de valeur, émis par Mark Twain Bank des États-Unis, a été identifié avec le nombre 4. Lore a que le système tôt attribué un petit numéro séquentiel à chaque monnaie. Les systèmes de test avaient déjà acquis 0,1,2,3 et donc 4 était le suivant. Les hypothèses marketing de Digicash ont ensuite changé pour assumer un problème par pays. Avec le temps, ce régime a été ajusté pour émettre des devises numérotées après les codes de composition internationaux (p. ex., 49 pour l'Allemagne, 61 pour l'Australie). Les insuffisances de ce régime sont devenues apparentes de sorte qu'un nouveau modèle a été créé [4]. Un numéro 32 bits pour décrire le problème a été utilisé sur l'hypothèse pragmatique que ce serait assez grand pour couvrir les éventualités prévisibles.

Pourtant, les tensions d'un émetteur d'une monnaie étaient évidentes presque immédiatement. Un schéma plus avancé pourrait utiliser un tuple de (émetteur, devise) pour décrire un système par lequel chaque émetteur est habilité dans un certain sens à émettre plusieurs devises concurrentes [5]. Il est facile de généraliser ce système en ajoutant des éléments supplémentaires au tuple : (émetteur, type, identifiant) tuple [6]. Par exemple, une obligation à coupon zéro émise par le Keiretsu commun universel et national qui paie en Janvier de 2100 pourrait avoir un tuple de (JUNK, zéro, Jan_2100).

II-2 Le problème avec les chiffres

Les nombres comme espace pour identifier les instruments numériques sont limitatifs et avoir des tuples comme extension n'est pas vraiment une réponse. Premièrement, qu'est-ce qu'ils décrivent ? Dans le cas des systèmes de trésorerie électroniques, ils peuvent décrire les devises et les émetteurs. S'agit-il d'un ou des deux ? et comment pouvons-nous généraliser à d'autres aspects ? Deuxièmement, quelle garantie avons-nous que ce qui est décrit est exact ? Alors que beaucoup peut être réalisé en s'appuyant simplement sur la réputation de l'émetteur, les initiés financiers savent que la valeur réelle est exprimée dans le détail et la fiabilité de la réclamation. Troisièmement, comment les nombres sont-ils dérivés ? Un registre central est-il nécessaire, ou tout émetteur de valeur numérique peut-il acquérir un nombre selon les exigences locales ? Enfin, y a-t-il une limite à l'espace ? Les nombres entiers exprimés en paquets sont généralement limités à une certaine quantité de bits comme 32. Pour l'ingénierie logicielle pratique, il doit y avoir des limites mais ces limites doivent-elles restreindre les possibilités commerciales?

II-3 Le défi du succès

Tout système efficace sera utilisé de façon à donner l'impression qu'il est défectueux. En tant qu'ingénieurs logiciels, nous devons présenter nos inventions avec l'humilité des outilleurs pour les générations futures de constructeurs pas comme les bureaucrates qui planifient le zonage de l'espace du commerce numérique. Que se passe-t-il lorsque les premiers adoptes dominent les mères et les grands-pères et que la concurrence s'acharne sur les retraités âgés ? Imaginez des pastilles de menthe dans les poches de milliards de personnes âgées qui jouent au ralenti. Autrement, imaginez un monde avec un émetteur de points de fidélité numériques sur chaque parcomètre, ou où les étudiants doivent payer les frais de scolarité avec des parts de gains futurs. Nous avons déjà vu des musiciens populaires vendre des obligations adossées à leur musique [7], et des propositions de correction de bogues de logiciels financés par des problèmes sécurisés à des utilisateurs anonymes [8].

II-4 L'obligation à coupon zéro

Prenons l'obligation à coupon zéro, un instrument qui paie la valeur *nominale* d'une devise à une date donnée. Le *zéro* est peut-être l'instrument financier général le plus simple d'usage courant et il a constitué la référence pour notre conception.

Pour décrire la valeur nominale, la devise de la valeur nominale et la date d'expiration de cette obligation, nous ajouterions des éléments supplémentaires au tuple ci-dessus. Mais ce n'est qu'un début. Dans sa description des Eurobonds, Noel Clarke attend des dizaines ou des centaines de champs [9]. Si nous n'examinons qu'une seule de ces caractéristiques, par exemple *les options de vente liées à un événement*, nous constatons qu'un cautionnement doit décrire ce qui se passe dans le cas de :

- Une prise de contrôle hostile ou amicale de l'émetteur
- Une prise de contrôle par l'émetteur d'une autre partie
- Une recapitalisation
- Un programme de rachat par l'émetteur de ses propres actions, ou bien
- Une répartition des actifs au-dessus d'un certain pourcentage de la valeur nette de l'émission

Ces éléments sont étroitement liés à l'instrument en question, mais ils représentent des difficultés pour l'architecte logiciel. Nous pouvons faire un certain nombre d'observations. Premièrement, chaque événement n'est pas simple. Aujourd'hui, on peut être en mesure de transformer la notion de « prise de contrôle hostile ou amicale » en une seule paire nom - valeur, mais cela ne survivrait pas à la scène évolutive de la réglementation et des litiges qui s'appliquent à de tels événements. Deuxièmement, il n'y a aucune raison de croire que la liste ci-dessus est complète. Troisièmement, non seulement il sera difficile de concevoir un seul domaine pour y faire face, mais il sera surtout rempli de textes juridiques.

Envisageons un point de vue de la mise en page des données. Pour décrire le document qui constitue la base d'un cautionnement, nous aurons besoin au minimum d'une base de

données arborescente de tuples. De plus, cette disposition ne fonctionnera que pour un seul instrument ou un ensemble d'instruments extrêmement étanche et presque fongible.

II-5 L'argent comptant est roi

Les devises, ou l'argent comptant, pourraient être si serrés. Après tout, un dollar est un dollar est un dollar. Pouvons-nous décrire l'argent avec un simple ensemble de tuples ? Même pour l'argent, nous soutenons qu'une disposition de tuples n'est pas suffisante.

Prenons le cas d'un dollar numérique émis par une banque. Le dollar numérique serait un dérivé, souvent adossé à des dépôts du même montant. Cela peut être suffisant à des fins de commercialisation, mais cela ne survivrait pas à une analyse financière sérieuse.

Comparez ces dollars dérivés à ceux émis par le Federal Reserve Board des États-Unis. La Fed doit encore refuser l'acceptation de ses notes si elles sont présentées avec la même, ne serait-ce que comme une réclamation sur un autre ensemble du même instrument, ou pour les passifs fiscaux. Des interprétations radicales mis à part, la Fed n'a jamais fait faillite et reste un pari assez solide.

On ne peut pas en dire autant de n'importe quel émetteur bancaire de dollars dérivés. Ses dollars numériques seraient garantis par des dépôts auprès... de la toute même institution. Une telle banque peut fermer ses portes à tout moment et étant donné l'histoire du secteur bancaire au 20ème siècle, un analyste devrait prendre ce risque au sérieux. De plus aux États-Unis du moins, la FDIC a déjà statué que les fonds ainsi détenus sur le PC d'un utilisateur sont considérés comme des dépôts non assurés [10].

Cela ne veut pas dire qu'une banque donnée est sur le point de fermer ses portes mais que se passe-t-il lorsqu'un émetteur manque à sa promesse ?

Tout détenteur d'un bien encourt un risque. Un détenteur de dollars électroniques assumera le risque que l'émetteur fasse faillite, et le détenteur de dollars d'un autre émetteur assume un risque similaire, comparable, mais *distinct*. Chacun de ces risques entraîne un coût qui devrait être soustrait de la valeur nominale du dollar pour calculer une valeur comparative. Dans cette distinction de risque réside le fait inévitable que n'importe quel dollar donné n'est pas de valeur constante même lorsqu'il est mesuré par rapport à un dollar bien connu comme celui émis par la Réserve fédérale.

II-6 Les petits caractères du contrat

S'il n'y a pas un seul dollar, que reste-t-il ? Évidemment, il faut décrire chaque dollar pour ce qu'il est. Cela semble être une tâche de précision et de détail et, de fait, *chaque devise émise distincte est un contrat distinct entre l'émetteur et le détenteur.*

Un contrat peut encapsuler le détail. Considérons les contrats de change souverains originaux, dans lesquels l'émetteur a promis de payer le porteur en onces de métal précieux. C'est déjà quatre données dans le contrat : « Quel souverain ? Payer au porteur, quoi payer et combien de celui-ci ? »

Il en va de même pour chaque obligation, chaque devise et tout instrument financier de toute complexité. En effet, dans le domaine numérique, la question de savoir comment traiter un instrument financier se réduit en grande partie à la façon de traiter un contrat.

Sinon, un problème *est* un contrat. Les problèmes dans d'autres systèmes de paiement ont des contrats, mais seulement à titre de documents complémentaires, comme les ententes avec les utilisateurs. Souvent, leur rôle et leur importance sont sujets à des batailles ; le Marketing veut qu'ils soient cachés, tandis que Legal demande qu'ils soient poussés au visage de l'utilisateur en tout temps. Une fois que nous acceptons que le problème soit un contrat, la tâche devient simple : créer un contrat qui peut être lié dans le système de paiement comme pièce maîtresse. C'est le sujet de l'article suivant.

III. Système de contrats numériques et d'émission

Presque tous les aspects des contrats ricardiens sont mieux vus en examinant des exemples et cette section ne couvre que brièvement les détails saillants avant de discuter des ramifications. Des exemples sont disponibles sur webfunds.org/ricardo/contracts/.

III- 1 Définition

Un Contrat ricardien peut être défini comme un document unique qui est : a) un contrat offert par un émetteur aux détenteurs, b) pour un droit précieux détenu par les détenteurs et géré par l'émetteur, c) facilement lisible par les gens (comme un contrat sur papier), d) lisible par les programmes (analysable comme une base de données), e) signé numériquement, f) porte les clés et les informations du serveur et g) associé à un identifiant unique et sécurisé. Dans les termes les plus simples, un Contrat ricardien est un document définissant un type de valeur à émettre sur Internet [11]. Il identifie l'Emetteur, en tant que signataire et toutes les conditions et clauses que l'Emetteur juge bon d'ajouter pour faire du document un contrat. Le même document doit être à la fois lisible par les gens et par les programmes. Le Contrat ricardien est formaté comme un fichier texte qui peut être facilement lu (affiché ou imprimé) et les programmes peuvent le convertir en formulaires internes pour la recherche de paires nom - valeur. Il comprend une section spéciale pour chaque type de contrat comme l'obligation, l'action, la devise etc. D'autres sections dans « décrire, en termes analysables par programme, l'utilisation de points décimaux, les titres et les symboles. »

En tant que signataire légal, l'Emetteur signe le document dans le formulaire Openpgp cleartext avec sa clé de signature de contrat [12]. Il inclut la chaîne complète de clés Openpgp dans le document pour permettre aux programmes de vérifier et d'authentifier directement.

Pour identifier de manière unique le contrat, n'importe quel utilisateur peut calculer *un condensé de message canonique* sur le document clairement signé. Ce condensé de message est inclus dans tous les enregistrements de transactions et fournit un lien sécurisé (infalsifiable) du document à la comptabilité de la question. Par exemple, « e3b445c2a6d82df81ef46b54d386da23ce8f3775 » est le condensé complet des messages pour l'émission par Systemics Inc de dollars de services prépayés. Communément appelé hachage, le condensé de message est une technique cryptographique pour créer un nombre relativement petit qui est un à un avec le document. C'est-à-dire pour chaque document,

il n'y a qu'un seul hachage et le hachage se réfère uniquement à ce document. L'algorithme est le standard bien connu, SHA1.

III-2 Quelques observations

Les observations suivantes soulignent la force du résultat.

Le **Hash Limits Frog-Boiling :** Un changement graduel dans le contrat par la partie plus forte au fil du temps est connu comme « *grenouille-ébullition* ». La partie la plus forte est généralement l'émetteur et on peut s'attendre à changer le contrat s'il y a un avantage. Il s'agit d'une attaque fréquente. L'un des résultats de l'utilisation de l'identificateur de hachage n'est qu'aucune des parties ne peut modifier le contrat arbitrairement ou subrepticement. Pour voir que cela s'avère vrai, nous devons examiner les dossiers qui font référence au hachage. Une demande peut signer tous les documents importants (ex. paiements, jetons, reçus, soldes) et ces documents signés comprennent le hachage d'un contrat ricardien. Le hachage dans l'enregistrement ne peut pas être modifié sans perdre sa capacité à passer un test de validité de signature. De même, le contrat ne peut être modifié sans perdre sa relation avec des documents déjà signés et livrés. En d'autres termes, chaque enregistrement, détenu par chaque utilisateur, incorpore une copie inaltérable de ce hachage. Toute modification du contrat crée un nouveau « hash » et ce nouveau « hash » n'est pas celui que les utilisateurs ont ou apprécient. Cela cristallise le contrat pour les deux parties empêchant la partie la plus forte de modifier subtilement le contrat à un stade ultérieur. Dans une certaine mesure, cela corrige le déséquilibre de pouvoir entre le fournisseur et le client dans l'offre d'un contrat type. La partie inférieure n'a pas d'option pour négocier mais la partie supérieure n'a pas non plus l'option de réclamer un contrat distinct à une date ultérieure. La limitation a un certain coût car elle peut être une nuisance pour l'équipe de soutien de cet instrument financier.

L'ICP ricardien apporte de la clarté : Les contrats ricardiens possèdent leur propre infrastructure à clé publique (« PKI »). La clé publique de haut niveau de l'Emetteur est incluse dans le contrat et elle signe sa clé de signature de contrat, également incluse. La clé de signature du contrat signe le contrat lui-même. Cela permet plusieurs choses. Premièrement, le logiciel client peut vérifier l'ensemble de la chaîne de signature numérique dans une séquence automatisée. Deuxièmement, il n'y a pas besoin d'un complexe multi-partis PKI. Toutes les clés sont présentes et il n'y a pas besoin d'aller les chercher sur le net. Cela élimine les attaques de substitution où une clé qui pourrait passer quelques contrôles pourrait être inséré dans une phase de recherche de clé. Cela réduit aussi considérablement les coûts. Troisièmement, le hachage canonique du contrat représente également une signature sur le contrat. Il est consigné dans tous les dossiers pertinents et, par conséquent, entre le contrat et ces activités [13]. Une fois que le contrat est en vigueur depuis un certain temps, il établit sa provenance par la présence et la confiance du public utilisateur. Cela fournit une preuve beaucoup plus convaincante que la signature de l'émetteur lui-même ; une fois que l'émetteur et le public ont dépensé du temps et de l'argent en s'appuyant sur le contrat et par le biais du hachage, il est difficile pour l'émetteur de revenir sur la nature du contrat ou de sa signature. Le résultat est une ICP qui offre une fiabilité de bout en bout solide et basée sur un seul document. Cela n'est tout simplement pas présent dans d'autres modèles pour les ICP [14]. Cette fiabilité porte ses fruits dans la phase de règlement des différends où, à notre avis, le contrat ricardien peut être autonome sur ses mérites et ne nécessite aucune description complexe de l'ICP, des signatures numériques ou des références à des tiers parties incertaines pour renforcer

sa provenance. En incluant les clés, nous pouvons dessiner quelques lignes simples dans le contrat affirmant que « cette clé signe cette clé et que cette dernière signe le contrat. La première clé est la clé de premier niveau de la personne qui a signé ce contrat. C'est toute l'histoire, vot' Honneur ! ».

Validation de la clé de l'émetteur : Tous les bons protocoles de crypto se divisent en deux parties, dont la première dit à la seconde : "faire entièrement confiance à cette clé." La clé de premier niveau de l'Emetteur authentifie finalement le contrat. Les clés et autres informations contenues dans le contrat permettent également à un protocole tel que SOX de démarrer une connexion fortement sécurisée au serveur [15]. Comment alors vérifier que cette clé ultime est bien celle de l'Emetteur ? Ce n'est pas difficile. Le processus opérationnel de l'émission numérique implique beaucoup de relations entre les émetteurs et les utilisateurs. De nombreuses interactions différentes impliquent des chances d'établir la confiance. Par exemple, à partir de son site Web, l'Emetteur peut publier le contrat, les clés et les hachages et faire en sorte que d'autres sites les reflètent. La valeur ainsi émise sera distribuée par le biais de paiements qui comprennent le hachage. Une partie déjà de confiance livre habituellement ces paiements. Les paiements validement identifier le contrat et dériver leur propre validité via le hachage. Comparez ceci aux hypothèses dans l'ICP x.509 derrière la navigation SSL/HTTPS (ce qui suit est très discutable mais est présenté à titre de comparaison seulement). Dans ce PKI, il a été initialement affirmé qu'un utilisateur présenterait sa carte de crédit à des sites avec lesquels elle n'avait aucune relation antérieure et aucun moyen pour elle d'établir la provenance de la clé du site. Ainsi, un tiers de confiance, l'Autorité de certification, a été mis en place pour confirmer la clé. Les paiements, le commerce et les questions financières sont fondamentalement riches en relations. La nature de l'argent et des finances c'est que les participants font toujours preuve de diligence raisonnable. Ils préfèrent écouter les pairs en qui ils ont déjà confiance et n'acceptent pas facilement la parole d'une partie indépendante. Ainsi, il n'y a pas de place pour un tiers central pour se tenir debout et authentifier les joueurs. Avant que l'utilisateur souhaite placer une valeur sur un paiement donné, elle a presque certainement été mis au courant du contrat par d'autres moyens.

Présomption de possession : L'utilisation du hachage comme identificateur est un compromis car elle est inintelligible pour les humains [16]. Pourtant, ce compromis offre un avantage inattendu : *l'utilisation de la question conduit à une présomption que l'utilisateur a le contrat*. Pour utiliser une question de valeur comme une devise, l'utilisateur doit avoir le hachage dans les enregistrements applicables. C'est-à-dire que si l'utilisateur reçoit un paiement, ce dossier de paiement comprendra le hachage. Comme le hachage n'est pas descriptif, cela implique que l'utilisateur a le contrat pour interpréter le problème. Pour voir que c'est vrai, imaginez avoir un enregistrement avec le hash mais sans avoir le contrat. La première chose dont l'utilisateur aura besoin est une base de données de paramètres lui indiquant à quoi le hachage se réfère. Contrairement à un paiement en 10 de « GBP », un paiement de 1000 en « 972097bb... » n'est pas intelligible. Pourtant, comment le logiciel pourrait-il prédire ce que l'utilisateur doit savoir au sujet du hachage ? Très rapidement, il devient évident que le logiciel est mieux de stocker la source de l'information - le contrat complet lui-même - car il supprime un degré illimité de complexité dans le stockage des informations intermédiaires ou secondaires. Le logiciel peut toujours fonctionner avec le hachage seulement. Cependant, il serait entièrement aveugle à la sémantique de l'instrument. Une telle approche cavalière pourrait être acceptable pour les communications et le stockage mais pour les logiciels d'utilisateur, elle

équivaut à une défaillance traumatique. Pour faire face à cela, le logiciel côté client prend particulièrement soin d'acquérir et de conserver des contrats. Par conséquent, nous pouvons énoncer la présomption avec un certain degré de confiance : dans un système de fonctionnement, l'utilisateur dispose de l'intégralité du contrat ricardien (bien que sous le contrôle d'un logiciel). Ce n'est qu'un petit pas pour le logiciel client mais c'est un grand pas en avant pour la relation entre l'émetteur et le détenteur. Plus précisément, le fait de présumer fortement que l'utilisateur dispose de la totalité du contrat simplifiera de nombreux aspects juridiques concernant les responsabilités de l'émetteur. (Nous suggérons et reconnaissons donc les ramifications juridiques du terme présomption mais ni l'espace ni l'expertise ne permettent plus dans ce document.)

III- 3 Les quatre coins de la page

Le Contrat Ricardien fournit une riche source d'informations primaires et complètes. L'histoire complète est ici, sous forme textuelle, dans des paramètres analysables et dans la chaîne de signature. Ainsi, dans le cadre d'un litige, une attaque judiciaire hostile a moins de marge de manœuvre et ne peut que confirmer les faits tels qu'ils sont énoncés dans le contrat. Notre intention est que le contrat soit le début et la fin de la discussion ; nous appelons ce principe *la règle d'un contrat*. La fraternité juridique se réfère à « le contrat étant limité par *les quatre coins de la page* ». En montrant comment nous avons soigneusement exposé un document lisible avec une signature numérique vérifiable et un identificateur impardonnable reliant à chaque enregistrement, nous pouvons plus facilement demander à la magistrature d'accepter que le document unique qui est présenté est effectivement le contrat valide convenu par les parties.

IV. Conclusion

Le contrat est la clé de voûte de l'émission [17]. Notre innovation est d'exprimer tous les détails saillants d'une émission comme un contrat impardonnable et indissociable dans chaque action dans un système de paiement. De cette façon, l'innovation financière peut évoluer dans le sens qu'elle a toujours suivi - par l'innovation dans les contrats. En traduisant l'institution du contrat dans le domaine numérique, nous nous appuyons sur des siècles et même des millénaires d'expérience dans la documentation, le partage et la contestation de la signification des accords entre les parties.

IV -1 Le défi de la complexité

Pour saisir la complexité, nous pouvons mettre des documents comme des contrats sous forme électronique et les signer à l'aide de technologies de signature numérique comme Openpgp. Le résultat est un analogue raisonnable des contrats de papier et d'encre que la plupart des gens et des entreprises connaissent bien avec une intégrité cryptographique renforcée.

Avec le hachage comme identificateur, le logiciel peut désormais identifier de façon unique un arrangement financier donné et peut confirmer une solide chaîne de signatures. Le hachage implique fortement que l'utilisateur a le contrat disponible en tout temps et il ne peut pas être modifié sans être remarqué. Le contrat ricardien offre un énorme avantage à l'émetteur - clarté dans de nombreuses questions juridiques et de soutien à la

clientèle. L'utilisateur bénéficie d'une baisse des coûts globaux et d'une meilleure présentation de l'information dans un cadre plus cohérent.

IV -2 Enseignements

Le formulaire est utilisé avec succès depuis 1996. Depuis, il a livré une vingtaine d'instruments financiers sans défaillance.

Litiges: Le Contrat Ricardien est apparu dans deux forums distincts de règlement des litiges pour résoudre les réclamations [18]. Anecdotiquement, chaque réclamation a été résolue directement et efficacement et sans tracas indus, simplement en se référant au contrat ricardien applicable.

Automatisation: Il n'y a pas grand-chose à automatiser. Dans la pratique, des champs ont été insérés et standardisés afin que les programmes puissent extraire la décimalisation (dollars contre cents), les étiquettes pour les unités (USD contre $) et les titres pour l'émetteur et l'émission. Contrairement aux attentes, il n'y a pas eu de demande pour analyser chaque champ.

Coût: Le coût du concept a été comparé favorablement à celui d'autres systèmes de paiement. La préparation du texte du contrat comporte certains coûts mais pas plus qu'un accord d'utilisateur. Les exigences d'infrastructure Openpgp (clés et signature) ajoutent des coûts mineurs aux émetteurs mais ils sont facilement compensés par les avantages de la réduction des risques liés à la distribution des contrats. Les éditeurs de signatures personnalisées ont contribué à réduire ces coûts [19].

IV -3 Les défis de l'avenir

La superposition : La superposition des contrats est un besoin imminent. De nombreuses entreprises peuvent s'inspirer directement d'une norme et d'un ensemble de conditions définies. D'autres contrats résultent de contrats antérieurs et doivent y faire référence.

XML : Les premiers efforts ont suggéré que XML briserait la règle d'un contrat mais il semble que nous aurons besoin de quelque chose de mieux que le format INI archaïque [20]. Une proposition récente, le Voucher XML, ne se présente pas comme un contrat [21].

Le droit du contrat : Le traitement du contrat ricardien comme un contrat peut soulever plus de questions juridiques qu'il ne répond. Par exemple, ce formulaire est-il en effet un contrat ? Comment les juridictions distinctes voient-elles le concept (droit commun, droit civil, UCC, code coranique) ? S'agit-il d'un contrat négocié ou d'un contrat type ? Quand l'utilisateur a-t-il accepté le contrat ? Dans quelle mesure la présomption que l'utilisateur a le contrat est-elle solide ou réfutable ?

Les contrats intelligents : En unifiant toutes les informations dans un fichier lisible par programme, il y a un potentiel accru de contrats intelligents [22]. Nous ne sommes pas allés plus loin que des méthodes pour traiter les décimales. C'est en partie par manque de demande et en partie parce qu'il n'est pas clair comment un tribunal traiterait un programme informatique présenté comme un contrat.

V. Références

[1] Initialement introduit dans Ian Grigg, "Financial Cryptography in 7 Layers," *4th Conference on Financial Cryptography,* Anguilla, 2000, Springer-Verlag LNCS 1962. All papers are at http://iang.org/papers/

[2] Ian Grigg, "Digital Trading," *Virtual Finance Report*, November 1997.
[3] Country and Currency Codes, ISO3166-1.
[4] Bryce Wilcox, open design review, *DigiCash's developer list,* <ecash-dev@digicash.com>, August 1997.
[5] Ibid, Rachel Willmer, 14 August 1997.
[6] Robert Hettinga, " What's a Digital Bearer Bond?" *e$ rants* , 19th November, 1995
[7] Alex Tajirian, " David Bowie Bonds,"
[8] Ian Grigg and C. Petro, "Using Electronic Markets Achieve Efficient Task Distribution," *1st Conference on Financial Cryptography* , Anguilla, 1997, Springer-Verlag LNCS 1318.
[9] Noel Clarke, *Guide to Eurobonds*, The Economist Intelligence Unit, 1993.
[10] FDIC General Counsel's Opinion No. 8; Stored Value Cards, *Federal Register*, August 2, 1996. Also see the (readable) Press Release entitled FDIC will Continue to rely on General Counsel Opinion rather than issue rules on Stored-Value Cards, 24 June 97.
[11] Ian Grigg, Guide to Ricardian Contracts, *WebFunds project*.
[12] Jon Callas, et al, "OpenPGP Message Format," *Internet Draft*, RFC2440bis (-10 draft).
[13] Petros Maniatis, Mary Baker "Secure History Preservation through Timeline Entanglement" , *11th USENIX Security Symposium*, San Francisco, USA. August 2002.
[14] Jane K. Winn, " Couriers without Luggage" 49 *South Carolina Law Review* 739 (1998)
[15] Gary Howland, "Development of an Open and Flexible Payment System" 1996.
[16] Bryce Wilcox, "Names: Decentralized, Secure, Human-Meaningful: Choose Two", 2003
[17] Metaphor by Martin (Hasan) Bramwell. See "The Contract is the Keystone of Issuance," *Financial Cryptography blog*, 19th September 2003.
[18] *DigiGold v. Systemics*, before the Supreme Court of Anguilla (2001), and thereafter referred to the American Arbitration Association (2002).
[19] Edwin Woudt, ContractSignWizard, *WebFunds project*.
[20] Erwin van der Koogh, "Ricardian Contracts in XML," (presented at) *Edinburgh Financial Cryptography Engineering* (EFCE-2), 2001.
[21] Ko Fujimura and Masayuki Terada, XML Voucher: Generic Voucher Language, *Internet Draft*.
[22] Nick Szabo, " The Idea of Smart Contracts," 1997.