

FOCUS
ON

IL MITO DEGLI SMART CONTRACT: CONTRATTI INTELLIGENTI, MA NON TROPPO...

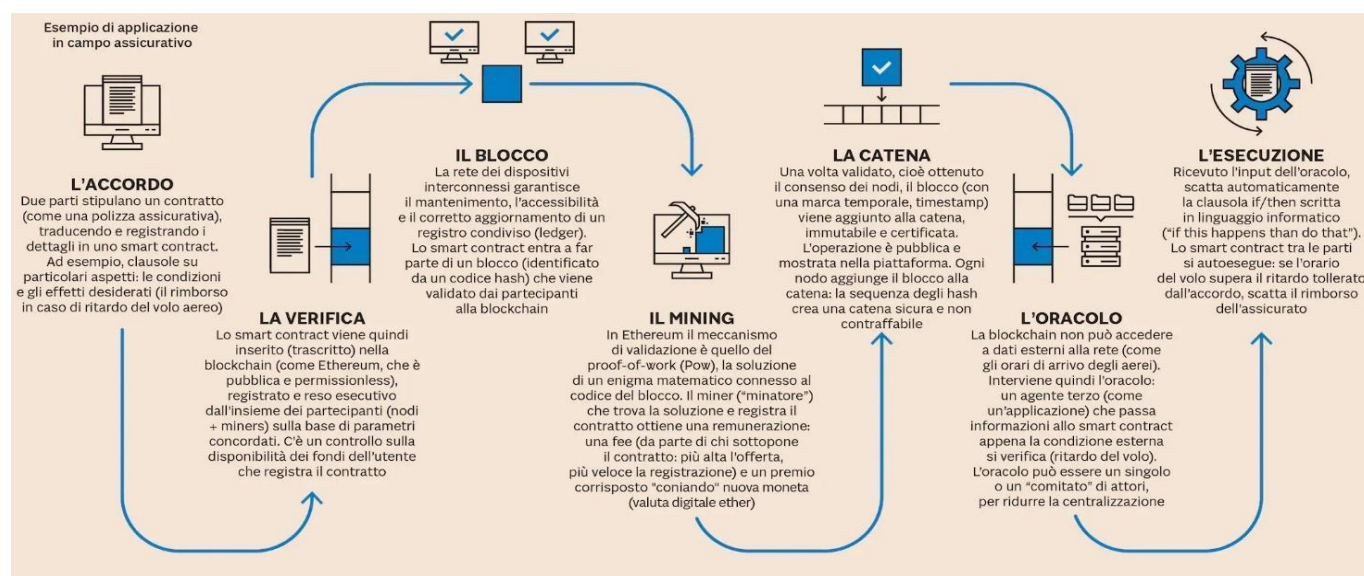
Cosa sono (e come funzionano) le clausole su Blockchain che non hanno nulla né di smart né di contract.

**LE NUOVE
FRONTIERE
DELLA
LEGGE**

www.lexant.it

ottenere *X* deve interagire con il protocollo creato da *Bob* trasferendo, in caso di accettazione dei termini dello scambio, la somma *Y*. Essendo integrate le condizioni dello scambio, l'algoritmo dello *smart contract* rilascia *X* ad *Alice* e trasferisce *Y* a *Bob*, eliminando il divario temporale tra le prestazioni collegate, nonché ogni spazio per il volontario inadempimento delle parti.

Nella sostanza, il meccanismo imita un deposito presso terzi, con lo *smart contract* quale autonomo deposito di informazioni e valori digitali.



In Figura 1 - Esempio di smart contract in ambito assicurativo

Gli *smart contract* possono essere programmati anche per tenere in considerazione degli *input* provenienti da fonti esterne, ritenute affidabili. Un protocollo può essere istruito al fine di vendere/acquistare un certo tipo di bene virtuale ovvero un certo numero di partecipazioni azionarie **solo se e quando** il prezzo raggiunga una certa soglia oppure vengano soddisfatte condizioni ulteriori. Ancora, l'algoritmo potrebbe autonomamente pagare il venditore una volta che il bene acquistato in rete è consegnato al compratore.

Le informazioni necessarie allo svolgimento di tali operazioni (es. prezzo dei beni e conferma di avvenuta consegna) sono fornite da c.d. **Oracoli (Oracles)**, ovverosia programmi indipendenti dalla *Blockchain* che monitorano dati esterni al sistema decentralizzato (come gli indici delle quotazioni azionarie oppure il *database* del vettore) e comunicano agli *smart contract* collegati il soddisfacimento delle condizioni rilevanti.

3. SMART CONTRACT: LE DUE FACCE DELLA MEDAGLIA.

La formalizzazione ed esecuzione digitale di uno scambio attraverso un sistema *Blockchain* riduce al minimo - quanto meno in linea teorica - il rischio di inadempimento implicito nella conclusione di qualsiasi contratto.

Lo *smart contract* si esegue al verificarsi delle condizioni programmate: si fa affidamento sul rigore della cornice tecnologica piuttosto che sullo spontaneo adempimento delle obbligazioni. L'adempimento di *Bob* è subordinato, inscindibilmente, a quello di *Alice*: **l'esecuzione dei termini dell'accordo è idealmente simultanea, così che non sarebbe possibile che una delle parti trattenga il pagamento o non consegni il bene promesso.**

Ulteriori vantaggi potrebbero derivare dall'eliminazione di profili di incertezza derivanti dall'ambiguità intrinseca al linguaggio naturale. Non diversamente da qualsiasi serie di informazioni comprensibili da un computer, gli *smart contract* operano sulla base di un linguaggio formale inequivocabile e altamente prevedibile, il quale non lascia spazio ad attività ermeneutiche di tipo testuale. Il rigore del codice del calcolatore impedisce letture discordanti delle stesse clausole, evitando dispute fondate sulla diversa interpretazione delle locuzioni utilizzate.

Infine, alla certezza sostanziale dell'adempimento si affiancano aspetti di certezza formale, derivanti dal c.d. **time-stamp** ovvero dal timbro - con giorno e ora esatta - apposto digitalmente ogni qualvolta l'istruzione viene inserita nella *Blockchain*.

Posta in questi termini, la differenza tra un comune contratto cartaceo e quello c.d. «intelligente» sembrerebbe data dalla sola capacità di quest'ultimo di dare esecuzione automatica ed indipendente alle obbligazioni assunte dalle parti, riportate in linguaggio macchina.

In realtà, volendo esaminare - più nel dettaglio - gli aspetti legali connessi allo *smart contract* emergono non poche criticità: di seguito una breve rassegna delle più rilevanti.

- a. **comprensibilità**: al pari di ogni altra istruzione destinata all'elaboratore, gli *smart contract* implicano un radicale spostamento dalla sfera del linguaggio naturale al codice. L'inesperienza tecnico-digitale delle parti dell'accordo riproporrebbe l'intermediazione nella relazione *on-line*: programmatori, prestatori di servizi oppure *software* di traduzione costituirebbero terze parti necessarie di ogni rapporto, filtro tra una volontà manifestata *off-line* e la sua espressione *smart*.
- b. **codice macchina**: i *computer* hanno bisogno di proposizioni univoche e predefinite per comprendere le istruzioni loro impartite. Si pone, dunque, il problema delle definizioni giuridiche oggetto di interpretazione che, in quanto tali, sarebbero difficilmente declinabili in codice crittografico.
- c. **valore legale**: la dimensione digitale degli *smart contract* crea alcuni ostacoli e difficoltà in relazione alla efficacia e vincolatività dell'accordo. Si pensi alla necessità - affinché il codice costituisca un contratto in base alla

governing law prescelta - che siano presenti gli elementi fondamentali che tale legge prevede come caratterizzanti un accordo contrattuale. In generale, l'approccio dovrebbe essere quello di verificare se lo *smart contract* possieda tutti i requisiti essenziali previsti dalla normativa locale per essere definito un contratto. Verifica che, nel caso del linguaggio macchina, appare assai complessa.

- d. **modifica/estinzione del vincolo**: uno *smart contract* legale - potendo far riferimento a un rapporto/evento complesso - potrebbe richiedere un intervento modificativo delle parti ovvero dell'Autorità giudiziaria, sia dal lato soggettivo (recesso, cessione) che dal lato oggettivo (modifica del contenuto). La comunità informatica e la piattaforma *Ethereum* (la più utilizzata in tema di *smart contract*) stanno già sviluppando dei modelli satellite che possano facilitare lo scioglimento e/o le modifiche contrattuali. D'altronde, rimettere ad un contratto satellite la modificazione dell'accordo è necessario ogni qualvolta non si possano prevedere *ex ante* tutte le variabili che possono presentarsi nel rapporto contrattuale.
- e. **certificazione esterna dei dati. Gli Oracles**: come detto, alcuni *smart contract* per poter funzionare devono ricevere dati che provengono da fonti esterne alla *Blockchain* (c.d. *Oracle*). Il meccanismo degli *Oracles* ha, tuttavia, l'inevitabile svantaggio di introdurre un grado di incertezza nel sistema: di fatto l'Oracolo diventa il soggetto che determina i comportamenti dello *smart contract*, per cui è essenziale che il *provider* sia affidabile e che le informazioni o notizie che raccoglie nel mondo esterno siano, a loro volta, certificabili.
- f. **hacking**: una problematica esclusiva degli *smart contract* è rappresentata dalla possibilità che gli stessi possano essere oggetto di attacco *hacker*: la *Blockchain* è sicura, ma i programmi che girano sulla piattaforma potrebbero avere vulnerabilità considerevoli.

4. SMART CONTRACT: RISORSA O MINACCIA?

In realtà, alcuni degli elementi alla base dell'utopia di Szabo si sono rivelati non del tutto realizzabili: **a)** gli *smart contract* non hanno, al momento, realizzato la destatalizzazione delle regole contrattuali; mentre **b)** l'effetto deflattivo sui contenziosi stenta a marginalizzare; anzi paradossalmente sono emersi nuovi ed inaspettati *points of failure*.

Oltre alle criticità tradizionali inerenti formazione del consenso e nullità/invalidità del contratto, l'uso degli *smart contract* presta il fianco ad una serie di rischi potenziali che potrebbero inficiarne l'effetto evolutivo, se non approcciati in maniera consapevole.

In primo luogo, resta il fatto che gli *smart contract* incontrano inevitabili limiti tecnico-giuridici. La loro applicazione è profittevole solo quando è semplice tradurre le clausole contrattuali in linguaggio informatico: se c'è un ritardo del volo, scatta il rimborso; se c'è una scadenza, parte il pagamento. A ciò si aggiunga che i programmi *software* richiedono - per loro stessa natura - che vi siano delle entità calcolabili, inserite nei parametri del codice, affinché al determinarsi di alcune situazioni vengano compiute le azioni corrispondenti.

La complessità, allo stato, non è gestibile. Anzi, la tendenziale immodificabilità della *Blockchain* può ritorcersi contro: cosa accadrebbe nel caso di codice errato? Il linguaggio macchina, che regge l'intera intelaiatura degli *smart contract*, è creato da un programmatore, il che non esclude l'errore umano.

Ed ancora, la scarsa flessibilità del codice pone interrogativi non di poco conto anche in caso di modifiche/revisioni legislative intervenute in un momento successivo all'attivazione del *contratto intelligente*.

In conclusione, nonostante la denominazione accattivante, gli *smart contract* non devono necessariamente qualificarsi come contratti giuridicamente intesi (sebbene possano esserlo ove ne integrino i requisiti). Andrebbero assimilati - più semplicemente - a degli strumenti per la negoziazione, conclusione e automatica applicazione di rapporti contrattuali o relazioni para-contrattuali: **un canale per la conclusione di accordi, piuttosto che un accordo in re ipsa.**

Il dibattito, dunque, non dovrebbe soffermarsi tanto sul come ricondurre i contratti intelligenti alle classiche categorie di diritto quanto, piuttosto, sulle scelte legislative da compiere nel caso in cui l'ordinamento risulti inidoneo a fornire una disciplina applicabile al caso di specie. L'Italia - anche sul solco delle indicazioni contenute nella Risoluzione del Parlamento europeo del 3 ottobre 2018 [*Risoluzione del Parlamento europeo del 3 ottobre 2018 sulle tecnologie di registro distribuito e blockchain: creare fiducia attraverso la disintermediazione (2017/2772(RSP)*] - è stata tra i primi membri UE a stabilire una disciplina positiva degli *smart contract* [art. 8-ter, comma 2 Legge 11.02.2019 n. 12 (*Conversione in legge, con modificazioni, del decreto-legge 14 dicembre 2018, n. 135, recante disposizioni urgenti in materia di sostegno e semplificazione per le imprese e per la pubblica amministrazione*), di conversione del D.L. 14.12.2018 n. 135)], ma i dubbi permangono.

Avv. Antonio Gerardo Giso