
BLOCKCHAIN A PRAWO

DARIUSZ SZOSTEK

pyt. - Rozdział XXV



WYDAWNICTWO C.H. BECK
WARSZAWA 2018

Rozdział V. „Inteligentne kontrakty”

Bardzo ciekawa definicja *smart contract* została wprowadzona w *Virtual Financial Assets (VFA) Bill* (przyjęty 5 lipca przez Parlament Malty¹⁷). W art. 2 ust. 2 określono „inteligentny kontrakt” jako formę organizacji technologii składającej się z: a) protokołu komputerowego lub b) umowy zawartej w całości lub części w postaci elektronicznej, która jest automatyczna i możliwa do wyegzekwowania przez komputer, chociaż niektóre jej części mogą wymagać ingerencji człowieka i kontroli przez człowieka i która może być również możliwa do wyegzekwowania za pomocą zwykłych metod prawnych lub za pomocą połączenia obu tych metod. Definicja ta w chwili obecnej najtrafniej odzwierciedla istotę *smart contract*.

„Inteligentne kontrakty” powoli stają się rzeczywistością, także uregulowaną prawnie. Wiele wskazuje na to, iż w najbliższym czasie także inne państwa na świecie wprowadzą odpowiednie regulacje w tym zakresie. Warto więc bliżej przyjrzeć się temu zagadnieniu.

§ 3. Pojęcie, cechy i podział »inteligentnych kontraktów»

I. Pojęcie i cechy

Zwrot *smart contract* czy też „inteligentny kontrakt” bądź „inteligentna umowa” należy uznać za nieszczególnie trafione, gdyż nie do końca odpowiada ono rzeczywistej roli, czy też pojęciu, jakie on spełnia, często wywołując wątpliwe jego zrozumienie. W publicznych twierdzeniach, w szczególności *start-upowców*, często pojawia się twierdzenie, iż *smart contract* zastępuje umowę, nie podlega pod żadną jurysdykcję terytorialną i funkcjonuje wyłącznie w sieci. Teza ta jest wyjątkowo nietrafna i wyraża tzw. myślenie życzeniowe¹⁸.

Pojęcie *smart contract* zarówno w literaturze, jak i w praktyce jest bardzo niejednoznaczne i obejmuje swoim zakresem szereg różnych zdarzeń. Podstawowymi elementami wyróżnianymi w większości definicji jest: zapis w kodzie programistycznym, oraz samowykonywalność, czy też automatyzowany spo-

§ 3. Pojęcie, cechy i podział »inteligentnych...

sób wykonania/realizacji. Czasami, chociaż nie zawsze, wskazuje się jeszcze na konieczność zapisu w *blockchain* lub w DLT. W praktyce większość „inteligentnych kontraktów” jest tak zapisywana. Na taki element wskazują także powyżej zaprezentowane definicje legalne. W literaturze jednak nie zawsze element ten jest wskazywany jako przesądający. Wskazuje się także na element: wbudowania w kod modułów zawierających klauzule umowne lub inne funkcjonalności, a także ich nieodwoływalność (realizację w związku z zapisem w programie).

Użyty w opisywanym pojęciu zwrot *contract* (czy też po polsku „kontrakt” lub „umowa”) należy uznać za wyjątkowo niefortunny, gdyż sugeruje, iż dorazowo mamy do czynienia z umową¹⁹, a tak w rzeczywistości nie jest. Nie zawsze „inteligentne kontrakty” są umowami, i to zarówno w znaczeniu prawnym, jak i potocznym²⁰. W praktyce „kontrakty inteligentne” należy podzielić na dwa rodzaje: 1) jako rzeczywistą umowę, zawierającą wyłącznie w sieci, poprzez jego akceptację (ale także często dodatkowo poprzez ściągnięcie oprogramowania), „samowykonywająca się” – przykładem takiej umowy jest umowa górników dotycząca wydobycia *bitcoina*; 2) jako narzędzie (nośnik) zapisu (z reguły w *blockchain*) odzwierciedlający realnie, w sposób tradycyjny, zawartą uprzednio umowę (np. na papierze bądź też w formie dokumentowej), będącą często umową ramową, lub warunkową, której wykonanie (tj. w ujęciu programistycznym uruchomienie dalszych procesów) dokonywane jest automatycznie poprzez program. W takim ujęciu *smart contract* nie zawsze będzie umową, a tylko narzędziem ją odzwierciedlającym i uruchamiającym jej wykonanie. Prawdopodobnie byłoby używanie zwrotu „inteligentny kontrakt” tylko dla pierwszego przypadku. Z tego też powodu w niniejszej pracy zwrot ten jest wpiśwany w cudzysłowie, jako określenie pewnego autonomicznego pojęcia, a nie jako umowa w rozumieniu prawnym. Nie przypadkowo definicje legalne nie odnoszą się w ogóle do pojęcia umowy. Definiują „inteligentne kontrakty” jako program lub jako kod programowy, a więc jako narzędzie technologiczne pozwalające albo na: 1) zawarcie umowy i jej całościową lub częściową realizację w sposób zautomatyzowany, albo 2) tylko na jej całościową lub częściową

¹⁹ Ang. *contract* – w tłumaczeniu to: kontrakt, umowa, ale także zlecenie, a w znaczeniu czarnosłownym – zawarcie umowy.

²⁰ Tak też A. Sierborne, *Blockchain, smart contracts and lawyers*, 2017, s. 5 https://www.google.com/url?sa=t&ct=1&q=&esrc=s&source=web&cd=1&ved=0ahUKEwi_y2944jcA9WMBiwKHa3CUMQFggvMAA&url=https%3A%2F%2Fwww.ibanet.org%2FDocument%2FDocument%2FDefault.aspx%3FDocumentId%3D317176-a-d-07a-4031-5301-000000000000

¹⁷ Zob. <http://justiceservices.gov.mt/DownloadDocument.aspx?app=lp&itemid=29079&l=1> (dostęp: 23.10.2018 r.).

¹⁸ J. Rogers, H. Jones-Fenleigh, A. Sanitt, *Arbitrating Smart Contract disputes*, International arbitration report, Norrthon Rose Fulbright, October 2017, s. 22 <http://www.norrthonrosefulbright.com>

realizację w sposób zautomatyzowany, zapisaną w DLT lub *blockchain* w technice gwarantującej autentyczność i integrację oraz niezaprzeczalność (nie tyle umowy, ile zapisu). Ostatecznie to, czym jest dany *smart contract*, determinuje jego treść lub treść umowy, która staje się podstawą jego uruchomienia.

„Inteligentny kontrakt”, a właściwie umowa go determinująca nie funkcjonuje w próżni prawnej ani poza prawem, czy też go zastępując. I to niezależnie od tego, czy jest zawierana w postaci elektronicznej bezpośrednio poprzez kod programistyczny stanowiący *smart contract*, czy też w sposób tradycyjny. Odnoszą się do niej kwestie m.in. zdolności prawnej, zdolności do czynności prawnej, sposobu jej zawarcia (oświadczeń woli, formy, kasualności czy też abstrakcyjności), treści umowy, klauzul abuzywnych (niedozwolonych), wykonania, wygaśnięcia, nieważności, dopuszczalności jej zmiany itd. A także „klasyczne” regulacje odnoszące się wyboru prawa właściwego oraz jurysdykcji, a w braku wyboru pozwalające na ich poszukiwanie. Tak jak odnosiły się te zasady do *bitcoin*a będącego *de facto* „inteligentnym kontraktem”. Zagadnienia te dotyczą klasycznego wykładu prawa prywatnego, nie miejsce zatem w niniejszej monografii na ich szersze omówienie.

Problemem w odniesieniu do „kontraktów inteligentnych” nie jest umowa, która jest wpisana lub wykonywana poprzez to narzędzie, ile technologia nie zawsze pozwalająca na synergię zdarzenia wynikającego z regulacji prawnych z działaniem programu. Problem stanowić też może anonimowość podmiotów (beneficjentów) zawierających kontrakt w cyberprzestrzeni. Jest to o wiele szerszy problem, dotyczący nie tylko *smart contract*, lecz także innych umów zawieranych elektronicznie, i wymaga osobnego, wykraczającego poza zakres niniejszej pracy, omówienia.

Istota „inteligentnego kontraktu” polega na jego samowykonywalności na podstawie trwałych i *de facto* nieusuwalnych zapisów w blokach *blockchain* lub DLT. Zmiany, w zależności od rodzaju *blockchain*, albo są w chwili obecnej niemożliwe (w przypadku stosowania *blockchain* publicznego o dużej mocy obliczeniowej), albo istotnie utrudnione (w niektórych *blockchainach* prywatnych), bądź odwracalne (w przypadku niektórych DLT bądź *blockchainów* prywatnych). Może to powodować, iż pomimo pojawienia się zdarzeń uzasadniających brak wykonania zobowiązania wynikającego z umowy (np. wada oświadczenia woli, nieważność bezwzględna, bezskuteczność zawieszona itd.) zobowiązanie zostanie wykonane, i to także w przypadku prawnomocnego orzeczenie sądowego i nie będzie możliwości „zatrzymania” jego

nie wykonywanych umów. Prawnikom nie są obce narzędzia prawne, pozwalające na przywrócenie stanu, jaki powinien zaistnieć w związku ze wzruszeniem umowy²¹. Nie zawsze jednak są one skuteczne czy wystarczające²². Sytuacja się poważnie komplikuje, nie jednak w znaczeniu prawnym, ale ewentualnych rozszerzeń zwrotnych, w odniesieniu „inteligentnych kontraktów” w ramach których funkcjonuje szereg podmiotów, a jedno zdarzenie wywołuje realizację kolejnego (tak jak w domino), gdzie niemożność zatrzymania wykonania umowy (np. w zakresie pierwszego zdarzenia) może powodować uruchomienie kolejnych i dalszych itd.

Podnosi się, iż atutem „inteligentnych kontraktów” jest ich niezaprzeczalność i pewność wykonania, a co za tym idzie – brak konieczności ich egzekwowania przed sądem. W ujęciu dzisiaj funkcjonujących regulacji prawnych jest to twierdzenie mające charakter życzeniowy. Wprawdzie umowa „sama się wykonuje”, jednakże nie oznacza to braku możliwości jej wzruszenia i dochodzenia swoich racji przed sądem. W krajach, w których wprowadzono w przepisach proceduralnych domniemanie zgodności treści zapisanej w *blockchain*, ułatwione jest dowodzenie faktu zawarcia umowy i jej treści, jednakże nie oznacza zakazu podnoszenia rzutów dotyczących samej umowy²³.

II. Podział

„Inteligentne kontrakty”, a także umowy ich dotyczący są bardzo różnorodne i nie można ich jednoznacznie sklasyfikować. W zależności od przyjętego kryterium można dokonać różnych ich podziałów²⁴.

Ze względu na sposób zawarcia umowy dotyczącej „inteligentnych kontraktów” możemy je podzielić na: 1) zawierane wyłącznie przez kod programi-

²¹ Przykładowo, gdy wykonanie „inteligentnego kontraktu” polega na dokonaniu płatności, w przypadku nieważności lub wzruszenia umowy, zapłata powinna być zwrócona. W przypadku braku dobrowolnego jej zwrotu, mogą się pojawić trudności w jej egzekucji, w szczególności gdy umowa ma charakter międzynarodowy, a także gdy płatność dokonywana jest poprzez kryptowaluty.

²² Budując *smart contract* warto stworzyć możliwość ingerencji w zapis umowy i jej wykonalność np. w przypadkach prawomocnie orzeczonej nieważności umowy.

²³ Tak też M. Kohwart, P. Margus, R. Addi, Smart Contracts, w: T. Kerikmaa, A. Rull, The Future of Law and eTechnologies, Heidelberg–New York–London 2016, s. 137 podnoszący iż w większości przypadków stosowania inteligentnych kontraktów strony mogą założyć brak konieczności szczególnego potwierdzenia umowy w sądzie, ale nie uchyla to ingerencji i prawa dochodzenia

styczny zawarty w „inteligentnym kontrakcie”; 2) zawierane wyłącznie w sposób tradycyjny (np. na papierze lub w formie dokumentowej); 3) zawieranej hybrydowo, gdzie umowa ramowa dotycząca „inteligentnego programu” jest zawarta tradycyjnie, jednakże jej uszczegółowienie i elementy szczególne w kodzie programistycznym, bądź równoległe jest zawierana umowa w kodzie i jej tradycyjnej wersji²⁵. Podział ten ma istotne znaczenie w aspekcie postępowania dowodowego przed sądem w przypadku sporu. Kod programistyczny, poprzez który zawierana jest umowa (pkt 1 i 3), nie zawsze jest zrozumiały dla nieprofesjonalistów²⁶. Treść umowy jest „zaszyta” w modułach programistycznych, które są uzupełniane przez strony. Najczęściej brak tu wizualizacji umowy, z jaką mamy do czynienia przy tradycyjnym sposobie ich zawarcia. A strona/strony nie zawsze zdają sobie sprawę z mechanizmu i sposobu działania, a nawet treści umowy. Nie jest to sytuacja nowa, wszakże od wielu lat zawierane są umowy *on-line*, z wykorzystaniem najpierw formularzy pasywnych, później formularzy aktywnych, akceptacja i uruchamianie oprogramowania itd., bez znajomości modułów oraz zasad funkcjonowania. W *smart contract* dochodzi do tego zapis w *blockchain* lub DLT oraz samowykonywalność umowy. Strony, zawierając umowę poprzez kod programistyczny, składają oświadczenia woli, na zasadach ogólnych zgodnie m.in. zgodnie z zasadą swobody wyrażania i składania oświadczeń woli. To, że ją wyrażają poprzez program, nie ma znaczenia dla przypisania im skutków oświadczenia woli. Zagadnienie wykozystania systemów informatycznych dla składania oświadczeń woli, w sposób poglądony zostało opisane prawie 20 lat temu (m.in. w aspekcie zaprogramowanej poczty elektronicznej czy też EDI), a „inteligentne kontrakty” jako nowy elektroniczny nośnik niczego tu nie zmieniają. Wykorzystując kod programistyczny do zawarcia umowy, należy uwzględnić ryzyko jego wadliwości, błędów programistycznych, wad oprogramowania, ryzyka związanego z atakami hackerskimi itd. (czego przykładem była sprawa eDEO²⁷). Pomimo że *smart contracts*, co do zasady, mają być pewne i przewidywalne, pozostają one (jak każde oprogramowanie) podatne na błędy i pomyłki w programowaniu, które

dodatkowo pogłębia nieodwracalny charakter łańcucha blokowego²⁸. Wpisany w *blockchain* program nie zawsze może być debugowany (wyszukana i usunięta wada oprogramowania), a konsekwencje jego samowykonywalności, pomimo wady, mogą być poważne i trudne (choćby nie niemożliwe z prawnego punktu widzenia) do odwrócenia. Błędy oprogramowania mogą wpływać na wadliwą wykonywalność *smart contract*, mogą się pojawić rozbieżności pomiędzy zakodowaną wersją a tradycyjną wersją umowy, a także mogą funkcjonować na podstawie niedokładnych danych²⁹.

Ze względu na budowę „inteligentnego kontraktu” można wyróżnić: a) *smart contract* zadeklarowany; b) *smart contract* modułowy. Z pierwszym mamy do czynienia w prostych umowach, zarówno dwustronnych, jak i wielostronnych, gdzie całość kontraktu jest zaszyta w kodzie, a sposób jego zawarcia ma charakter adhezyjny, tj. poprzez przystąpienie i akceptację całości, albo brak możliwości zawarcia kontraktu. Przykładem takiego kontraktu jest umowa pomiędzy górnikami w systemie *Bitcoin*. Modułowy *smart contract* polega na alternatywnych wyborach przez stronę odpowiednich modułów wcześniej zaprogramowanych w kodzie programistycznym. Strona ma wprawdzie swobodę w ich wyborze, jednakże nie ma wpływu na treść modułów, a także ich sekwencję. Jej wybór ogranicza się do wcześniej zadektowanego w systemie. Modułowe *smart contracts* wykorzystywane są zarówno dla prostych umów, jak i bardziej skomplikowanych czy też wielostronnych.

Ze względu na język programistyczny kodu możemy dokonać podziału na: 1) „inteligentne kontrakty” imperatywne oraz 2) deklaratywne. W chwili obecnej większość *smart contracts* przyjmuje podejście imperatywne, zgodnie z którym „inteligentny kontrakt” bezpośrednio określa operacje obliczeniowe, które mają być wykonane w celu realizacji umowy. „Podczas programowania w języku imperatywnym programista zapisuje wyraźną sekwencję kroków, które mają być wykonane w celu uzyskania zamierzonego rezultatu. Programista musi napisać, co należy zrobić i jak to wykonać. Języki deklaratywne są alternatywą dla języków imperatywnych³⁰. Konwencjonalne algorytmy mogą być analizowane pod kątem dwóch składowych: komponentu logicznego określającego, co ma być zrobione i komponentu sterującego określającego, jak ma to być zrobione. Komponent logiczny ma na celu wyrażenie wiedzy, która

²⁵ Tak też J. Rogers, H. Jones-Fenleigh, A. Saniti, Arbitrating Smart Contract disputes, s. 21 <http://www.nortonrosefulbright.com/files/20170925-international-arbitration-report-issu-e-9-157156.pdf> (dostęp: 25.7.2018 r.).

²⁶ Przykład funkcjonowania oraz zaprogramowania „inteligentnego kontraktu” por. <https://>

może być użyta w algorytmie, podczas gdy component sterujący wpływa jedynie na jego efektywność. W konsekwencji, gdy programowanie odbywa się w języku deklaratywnym, najlepiej, gdy program nie musi wyraźnie zapisywać kolejności kroków, aby określić, co należy zrobić. Programista opisuje tylko to, co ma być zrobione, nie precyzując, jak to zrobić. Inteligentne umowy deklaratywne mogą być sporządzane w różnych językach deklaratywnych, takich jak języki funkcjonalne i języki oparte na logice³¹.

Przyjmując kryterium-ekosystemu, w jakim funkcjonuje „inteligentny kontrakt”, możemy wyróżnić *smart contracts*: 1) funkcjonujące w zamkniętym ekosystemie; 2) posiadające zewnętrzne źródła, pozyskujące dodatkowe dane. Pierwsze opierają się głównie na imperatywnym języku programowania i wszystkie funkcje, czynności oraz zdarzenia zadektrowane są w kodzie kontraktu. Bardziej skomplikowane są „inteligentne kontrakty” odwołujące się, czy też korzystające z danych (pozyskujące je) z innych źródeł. Mogą one mieć charakter referencyjny, uzyskany od zaufanej osoby trzeciej (np. sąd, notariusz, podmiot zaufany w znaczeniu rozporządzenia EIDAS) lub od innego podmiotu.

Ze względu na rodzaj zapisu *smart contractu* 1) w DLT; 2) w *blockchain*; 3) w inny sposób. W odniesieniu do zarówno pierwszego, jak i drugiego rodzaju „inteligentnego kontraktu” możliwe jest wykorzystanie wielu różnych systemów informatycznych (publicznych, prywatnych itd.), tak jak wiele jest systemów DLT i *blockchain*.

Kolejnym kryterium pozwalającym na podział *smart contracts* jest sposób wykonania umowy. Automatyzm czy też samowykonawalność jest jedną z cech „inteligentnych kontraktów”. Jednakże różny może być poziom samowykonawalności. Kontrakty możemy podzielić na: 1) całkowicie samowykonujące się; 2) częściowo samowykonujące się, w których dla pełnego wykonania zobowiązania konieczne są dodatkowe działania, czy też dokonywane przez inne oprogramowanie, urzędzenia czy też przez inną osobę/podmiot.

Według innych kryteriów możemy dokonać podziału na: „inteligentne kontrakty” samoniszczące się, tj. takie gdy po wykonaniu zobowiązania kod ulega samodestrukcji; samouczące się, tj. oparte na algorytmach zbliżonych lub będących sztuczną inteligencją, wykorzystujących źródła zewnętrzne do „uczenia” się, a właściwie to zmiany sposobu wykonania kontraktu, w związku

ze zmianą zewnętrznych źródeł (np. zmiana wysokości odsetek wpływa na zmianę sposobu wykonania zobowiązania).

Ze względu na dostęp i możliwość zawarcia umowy możemy dokonać podziału na *smart contracts*: 1) otwarte – dostępne dla nieograniczonej grupy osób (w tym także dla podmiotów zagranicznych); 2) częściowo otwarte (np. dla podmiotów na określonym terytorium np. UE); 3) zamknięte – tylko dla określonej grupy. Ze względu na ilość podmiotów przystępujących do *smart contract*: 1) dwustronne; 2) wielostronne.

Uwzględniając przedmiot umowy: 1) dotyczące gospodarki cyfrowej; 2) dotyczące gospodarki realnej (tradycyjnej); 3) hybrydowe. W tym można uwzględnić kontrakty wykorzystujące token lub działające wyłączenie na zapisach w *blockchain* bez transferu tokenu.

Ze względu na transgraniczność: 1) międzynarodowe; 2) krajowe. W odniesieniu do pierwszych konieczny jest albo wybór prawa właściwego do kontraktu, albo jego poszukiwanie na zasadach ogólnych. W drugim przypadku prawo jest z góry określone podobnie jak jurysdykcja, ze względu na brak elementu transgranicznego. Ze względu na sposób rozstrzygania sporów wynikających z kontraktów: 1) podlegające pod arbitraż; 2) podlegające pod tradycyjne sądownictwo.

Powyższe jest wyłącznie przykładem podziału „inteligentnych kontraktów” i nie jest wyczerpujące. Ich mnogość, różnorodność, a także możliwość zastosowania wielu różnych kryteriów nie pozwala na pełny i wyczerpujący podział.

§ 4. Token w „inteligentnych kontraktach”

I. Wprowadzenie

Jednym z narzędzi wykorzystywanych w „inteligentnych kontraktach” są tokeny. Nie jest to konieczny wymóg dla funkcjonowania *smart contract*, ale w ostatnich czterech latach mamy wręcz tokenizację „inteligentnych kontraktów” oraz coraz większy obrót nimi na „giełdach kryptowalut”. *Smart contracts* oraz tokeny coraz częściej wykorzystywane są do pozyskiwania środków na inicjatywy dotyczące *blockchain* oraz kryptowalut. Przykładem jest ICO (skrót od *Initial Coin Offering*), czyli inicjująca oferta „monet” odnosząca się do zbywania tokenów w publicznych ofertach zwykle za kryptowaluty. Tokeny *właszcza* te o udanej ICO, są zazwyczaj notowane na giełdach krypt-