|  |  |  |
| --- | --- | --- |
|  | **Departamentul Automatică și Informatică Industrială**  **Facultatea Automatică și Calculatoare**  **Universitatea POLITEHNICA din București** |  |

**RAPORT STIINTIFIC NR. 1**

**Tehnologia Blockchain**

George-Aurel Ristoiu

**Conducător științific de doctorat:** Prof.dr.ing. Alin Iftemi

**Tema:** Managementul documentelor cu blockchain

**Program de masterat:** Managementul si Protecția Informației

Cuprins

[1 Introducere 2](#_Toc125236034)

[2 Părți componente si mod de operare 3](#_Toc125236035)

[3 Clasificare 5](#_Toc125236036)

[3.1 Rețele private cu permisiuni 5](#_Toc125236037)

[3.2 Rețele private fără permisiuni 5](#_Toc125236038)

[3.3 Rețele publice cu permisiuni 5](#_Toc125236039)

[3.4 Rețele publice fără permisiuni 6](#_Toc125236040)

[4 Contracte inteligente 7](#_Toc125236041)

[4.1 Avantaje ale contractelor inteligente 7](#_Toc125236042)

[4.2 Limitări ale contractelor inteligente si masuri de combatere 8](#_Toc125236043)

[5 Concluzie 11](#_Toc125236044)

[6 Bibliografie 12](#_Toc125236045)

Introducere

Blockchain este un registru distribuit, o rețea de noduri ce reprezintă calculatoare, mașini virtuale sau chiar containere de docker si este cunoscut pentru capabilitățile sale în sistemele bazate pe criptomonede precum Bitcoin si Ethereum. Orice lucru are o valoare, fie că este fizic precum o casă sau o mașină, virtual precum o subscripție la un serviciu sau este un anumit brand, iar deținătorii vor să se asigure ca își păstrează drepturile asupra lor. Din acest motiv a apărut si sistemul centralizat, o instanță imparțială care sa urmărească fiecare bun precum si tranzacțiile pe care le implica. Putem lua ca exemplu procesul de angajare care implica semnarea unui contract. Acesta devine dovada ca angajatul va oferi niște servicii, iar angajatorul îl va compensa monetar pentru munca depusa. În cazul în care una dintre parți nu își respecta promisiunea, exista o instituție la care se poate apela pentru a remedia situația.

Blockchain vine însă cu posibilitatea de a elimina acel intermediar prin introducerea unui registru de date ale cărui copii sunt răspândite printre toate nodurile unei rețele, așadar un sistem descentralizat. Acel registru este o copie al întregului sistem si se asigura de urmărirea tranzacțiilor si a modificărilor suferite de acesta.

Cu toate acestea, primul prototip pentru blockchain a apărut în anii 90’ si consta în aplicarea unor algoritmi criptografici într-un lanț de blocuri într-o încercare de a păstra integritatea datelor ce se aflau în documente digitale.

Părți componente si mod de operare

Un bloc este alcătuit din datele pe care le conține, un număr unic generat automat la crearea blocului numit “nonce” si un hash care reprezintă legătura către blocul precedent. Funcția hashing este o metoda de criptare unidirecționala, ireversibila, care generează o valoare unica ale datelor inițiale. În cazul Bitcoin, un bloc este “minat” o data la 10 minute, iar acest lucru este realizat prin modificarea nivelului de dificultate al calculării hashului prin impunerea unei reguli. Headerul unui bloc care a trecut prin funcția hash va fi mereu mai mic ca o valoare impusa, condiție împlinita prin adăugarea unui sir de zero la începutul hashului.

După ce blocul este creat, majoritatea membrilor din rețeaua distribuita trebuie sa fie de acord ca noua tranzacție apăruta este valida si devine parte din acel registru care va fi din nou împărțit tuturor membrilor. Nu trebuie totuși sa se ajungă la o decizie de comun acord, dar măcar 51% din noduri trebuie sa aibă același rezultat. În acest sens sunt definite niște reguli clare care trebuie urmate:

* Procesul este unul finit, iar fiecare nod va recunoaște o singura valoare ca fiind corecta
* Decizia luata de majoritate va fi finala
* Indivizii trebuie sa si pună interesele individuale deoparte si sa se concentreze pe cele ale grupului
* Fiecare vot reprezentat de fiecare nod este egal si cu cat mai mulți indivizi fac parte din rețea, cu atât mai bine.
* Fiecare individ dispune de un singur vot si trebuie sa participe active în luarea deciziilor

Algoritmii de decizie sunt Proof of Work (PoW), Proof of Stake(PoS) si Proof of Capacity(PoC). Proof of Work descrie un mecanism descentralizat care descurajează atacurile cibernetice prin rezolvarea unei probleme matematice complexe. Puterea de procesare necesara este procurata de membri rețelei ceea ce pune un efort considerabil asupra lor. Ideea a fost ulterior adaptata pentru securizarea băncilor de către Hal Finney în 2004 conform [1]. Proof of Work este folosit în principiu în minatul de criptomonede. “Mining” este procesul de a crea un nou bloc (o moneda noua în cazul criptomonedelor) si adăugarea lor în blockchain prin găsirea numărului specific al acelui bloc (“nonce”), dar numai prima persoana care reușește poate deține acel bloc. Nu exista un algoritm în spate, ci mai degrabă este ghicit printr-un număr foarte mare de încercări, deci forță bruta. Securitatea pe care o oferă acest proces este aceea ca daca dorești sa schimbi ceva în un bloc anume, trebuie sa minezi acel bloc precum si toate blocurile ce îl urmează, lucru care necesita o putere de procesare imensa si automat un cost pe măsura.

Proof of Stake presupune blocarea unui număr de token-uri pentru a valida tranzacțiile, iar la rândul lor, deținătorii acestora câștiga din urma taxelor percepute fiecărei tranzacții [2]. Acest concept este unul mai puțin riscant deoarece nodul ales pentru următorul bloc este ales la întâmplare si sunt mai puține de câștigat din urma unui atac cibernetic. Dezavantajul este ca pentru a lua parte în acest proces, trebuie sa ai un anumit număr de criptomonede ce vor fi blocate, lucru care este o investiție destul de mare având în vedere ca nu poți vinde acele criptomenede pentru o perioada îndelungata de timp.

Proof of Capacity este mai avantajos ca PoW si PoS deoarece oferă posibilitatea de a folosi spațiu gol de pe sistemul lor pentru a mina blocuri si a valida tranzacții [3]. Eficienta sa consta în posibilitatea de a genera o lista a tuturor hash-urilor înainte si salvate în spațiul disponibil de stocare. După adăugarea header-ului la hash-uri, blocul este minat si apoi un nou bloc, ce conține numărul unic pe care tocmai l-am aflat si hash-urile, este adăugat la blockchain. Acest proces este de doua ori mai rapid ca în metoda originala, PoW.

Clasificare

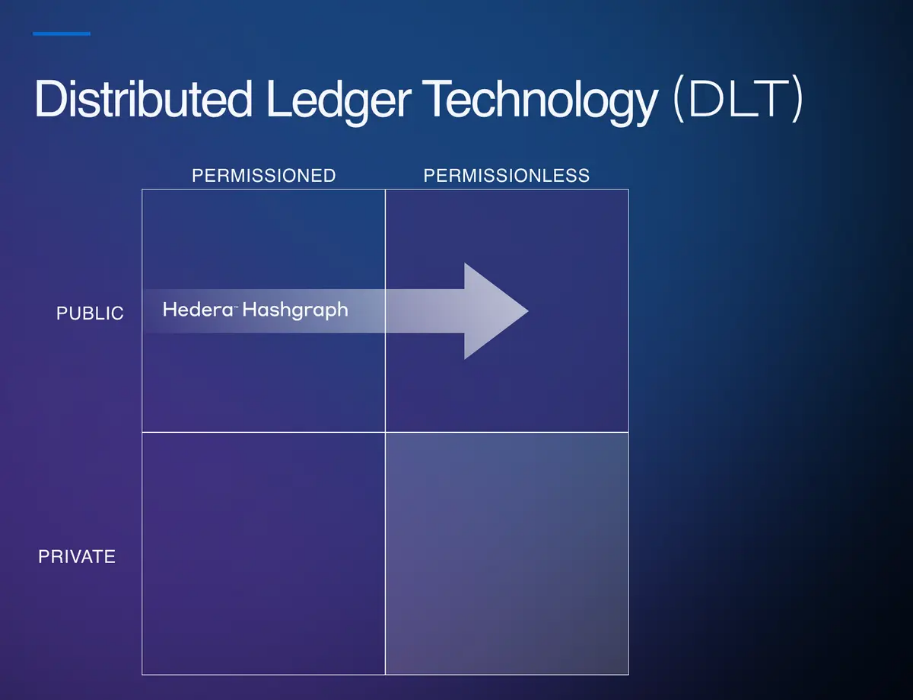


Figura 1 Tipuri de registre distribuite [4]

Rețele private cu permisiuni

Numai nodurile invitate pot face parte din rețea si acestea trebuie sa îndeplinească anumite condiții ca apoi sa li se verifice identitatea. Nodurile pot fi oricând eliminate din rețea, fără vreun avertisment

Rețele private fără permisiuni

Nodurile trebuie sa fie invitate înainte sa devina parte din rețea. Nu este nevoie sa li se mai verifice identitatea după si pot contribuii într-o rețea în mod anonim în schimbul criptomonedelor, dar în același timp pot fi oricând eliminate.

Rețele publice cu permisiuni

Permit nodurilor sa se alăture rețelei fără sa li se verifice identitatea, dar trebuie invitate în rețea în mod inițial. Nodurile pot fi oricând eliminate din rețea.

Rețele publice fără permisiuni

Apar în special la Proof of Stake deoarece nodurile pot fi adăugate oricând la rețea pentru a contribuii. Nu trebuie sa ofere o identitate, un nod poate contribuii într-o rețea în mod anonim în schimbul criptomonedelor, dar în același timp pot fi oricând eliminate fără avertisment.

Contracte inteligente

Contractele inteligente au fost propuse pentru prima dată în 1994 de Nick Szabo, un criptograf si informatician american care a inventat o monedă virtuală numită „Bit Gold” în 1998 conform [5]. Intenția lui a fost de a extinde capabilitățile tranzacțiilor digitale într-un mod securizat, dar nu dispunea de tehnologia necesara pentru a își implementa ideile.

Contractele inteligente sunt aplicații descentralizate proiectate pentru a executa, controla si documenta niște acțiuni în urma unor evenimente. Aceste acțiuni diferă în funcție de conținutul contractului si pot include tranzacționarea monedelor virtuale, livrarea unor servicii sau deblocarea unor resurse, precum fonduri, conținut cu drepturi de autor sau date personale, dar numai atunci când sunt îndeplinite condițiile predeterminate. Din acest motiv, contractile inteligente sunt stocate în blockchain, sau alt registru distribuit, deoarece oferă imutabilitate si transparenta, deci tranzacțiile pot fi urmărite. Un alt motiv pentru alegerea tehnologiei blockchain este flexibilitatea de a stoca orice tip de date.

Ethereum este platforma principala ce găzduiește contracte inteligente prin crearea unor limbaje de programare ușor de înțeles precum Vyper si Solidity. Acestea sunt folosite pentru a scrie codul din spatele aplicațiilor menite sa ruleze într-un mediu de execuție numit Ethereum Virtual Machine (EVM)[6]. Un alt mediu popular de programare este WebAssembly (WASM), care permite dezvoltarea si integrarea contractelor inteligente în registre distribuite prin diverse limbaje de programare precum C, Rust, JavaScript si Rust. Limbajul de modelare a activelor digitale (DAML) este folosit pentru modelarea cazurilor de utilizarea, sau la păstrarea confidențialității.

Avantaje ale contractelor inteligente

Câteva dintre avantajele utilizării contractelor inteligente sunt:

* Costul redus
* Viteza de procesare si acuratețea
* Autonomia
* Fiabilitate
* Stocare a si redundanta

Ideea din spatele contractelor inteligente este de automatizare, fiind un proces ce rulează în blockchain fără intermediar. Eficienta costului se observa de la început atunci sunt economisite resurse atât hardware si software, cat si resurse umane. Astfel, nu mai este nevoie de programe complexe care sa controleze procesele si resursele si deci, nu mai este nevoie nici de persoane care sa le utilizeze. Puterea de calcul este si ea minimizata deoarece nu este suportata de o singura instituție, ci este împărțita între membri rețelei în schimbul monedelor digitale.

Viteza de procesare este si ea îmbunătățita deoarece contractele inteligente elimina nevoia proceselor birocratice care pot depinde de la o instituție la alta. De asemenea, Evita si posibilitatea erorilor umane în timpul proceselor si timpul pierdut în încercarea remedierii acestora.

Autonomia este asigurata prin instrucțiunile de cod ce sunt executate si se asigura ca tranzacțiile sunt duse la capăt fără nevoia unui intermediar.

Fiabilitatea este data de tehnologiile bazate pe registre distribuite care elimina potențialul de eroare umana si păstrează o evidenta a tuturor proceselor în starea lor inițială, neputând fi modificate ulterior.

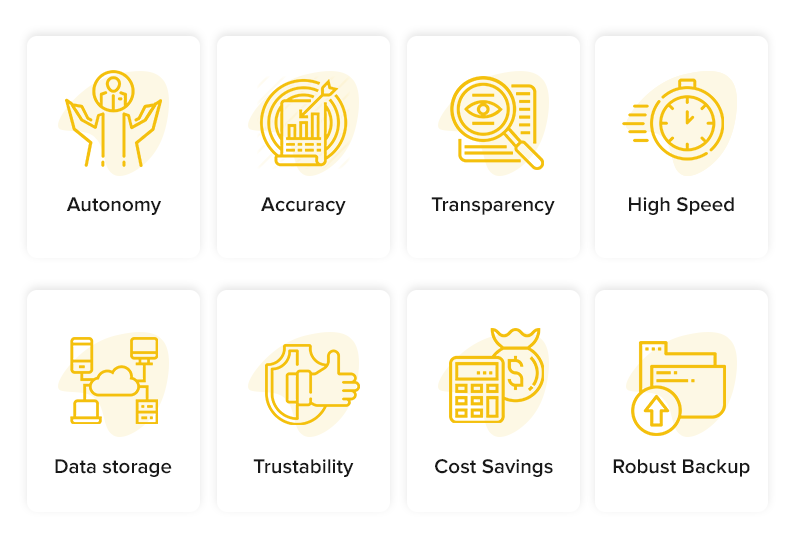


Figura 2 Avantajele contractelor inteligente [7]

Limitări ale contractelor inteligente si masuri de combatere

Există [numeroase probleme și provocări care trebuie luate în considerare](https://www.techtarget.com/searchcio/tip/5-challenges-with-blockchain-adoption-and-how-to-avoid-them) atunci când înclinăm să folosim contracte inteligente precum dimensiunea limitata la 24kb sau faptul ca nu pot primi sau trimite informații în timp real prin HTTPS.

Din punct de vedere al securității, tehnologia este în continua dezvoltare si lasă loc de breșe de securitate care pot fi folosite în scop malițios. Un bun exemplu este atacul DAO din 2016 când au fost furate monede Ethereum în valoare de 3.6 milioane de dolari care în prezent ar valora un miliard de dolari [8]. Problema principala consta în caracteristica de imutabilitate, care nu face posibila modificarea contractelor pe parcurs pentru a rezolva aceste posibile slăbiciuni. De aceea trebuie urmate niște reglementari de la stadiile inițiale de creare a contractelor:

* Implementarea unor politici de acces stricte
* Crearea unor teste ce iau în considerare analiza statica si dinamica a modului de operare
* Verificarea manuala de către mai multe persoane
* Propunerea unui plan de acțiune în cazul unei breșe
* Reducerea complexității codului la minim

Contractele inteligente conțin funcții publice si private. Cele publice sunt necesare pentru interacționarea cu acesta în orice fel, dar cele private descriu funcționalități la care nu ar trebui sa fie accesibile de conturi externe precum crearea de monede digitale noi. Pentru a evita asta, putem restricționa accesul prin introducerea accesului pe baza de roluri. Putem avea un singur ce conține permisiunile către funcțiile protejate si nicio alta adresa nu va avea acces, dar intervine problema centralizării. Având o singura entitate ce deține toate drepturile, exista riscul ca cheia private a acelui cont sa fie expusa si acesta sa fie compromise. O opțiune alternativa este introducerea mai multor conturi, fiecare având o mica parte din acele permisiuni. Daca avem nevoie de masuri de Securitate suplimentare, putem folosi un cont deținut de mai multe persoane, care ar necesita un număr minim de semnături pentru a executa o tranzacție

În privința testării contractului, nu sunt suficiente testele unitare clasice regăsite în orice proiect [9]. Ele sunt concepute pentru a se asigura ca acel contract se comporta cum a fost gândit, dar se bazează pe niște seturi de date limitate. Prin urmare, sunt la fel de eficiente ca persoana care le-a gândit si este nevoie de o analiza atât statica cat si dinamica. Analiza statica se refera la reprezentarea structurii codului sub forma unui arbore de sintaxa abstract, care poate fi folosit pentru a găsi erori de sintaxa si tipare. În plus este nevoie de un graph care descrie toate cazurile de utilizare si stările posibile în care poate ajunge. Similar, analiza dinamica propune rularea codului cu diferite date de intrare pentru a încerca sa se ajungă într-o stare ce nu a fost considerate la proiectare.

Separat de testarea automata este nevoie si de testare manuala care se poate face de persoanele care dezvolta programul respectiv, cat si prin audit. Auditul este făcut de mai mulți programatori care încearcă sa găsească defecte si greșeli de proiectare, chiar daca nu se așteaptă sa găsească orice posibila problema de Securitate ce ar putea apărea. Doar furnizează un set nou de păreri asupra produsului. O alta metoda, de obicei utilizata în fazele finale ale proiectului este oferirea unei recompense pentru oricine descoperă vulnerabilități în aplicație.

Cu toate ca încercam sa evitam acest lucru, propunerea unui plan de operare în cazul unei breșe de Securitate este un pas esențial în procesul de proiectare. O posibila opțiune de rezolvare a unei amenințări este posibilitatea de a schimba versiunea codului. Cu toate ca nu pot fi făcute schimbări directe datorita proprietății de imutabilitate, codul poate fi conceput pe baza unei metode de proiectare pe baza de Proxy (Proxy design pattern). Astfel, contractual este separat în logica si stare. Partea de stare acoperă conține variabile precum valoarea monetara aflata un portofelul digital legat de o anumita adresa, care nu poate fi modificata, si partea de logica, care descrie cum se va comporta contractul în anumite situații care poate fi modificata în mod restrictive. O alta opțiune este includerea unei opțiuni de oprire a funcționalităților contractului prin introducerea unei condiții suplimentare. Se poate introduce o variabila de tip Boolean care dictează daca anumite acțiuni sunt funcționale sau nu, iar drepturile de editare pentru a schimba aceasta valoare pot fi atribuite unui cont prestabilit. În plus, pot fi urmărite evenimentele care duc la schimbarea variabilelor de stare, lucru care micșorează timpul de răspuns la o posibila problema.

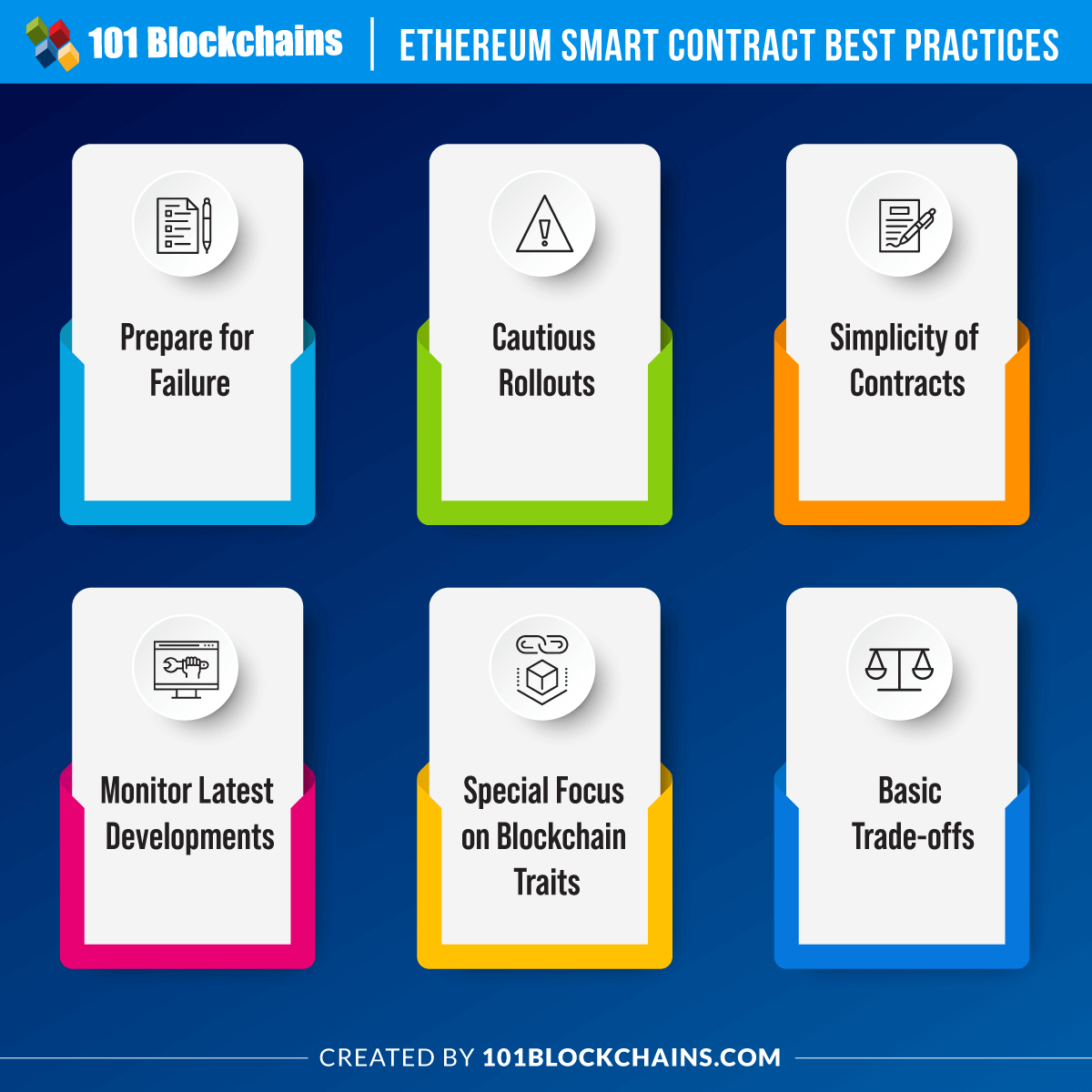


Figura 3 Cele mai bune practici de proiectare [10]

Concluzie

Atât tehnologia blockchain cat si contractele inteligente sunt complexe, cu un potențial imens ce nu se rezuma doar la simple tranzacții. Industriile în care pot fi folosite sunt nenumărate, de la cel financiar prin automatizarea proceselor si înlocuirea băncilor si pana la IoT. Dispozitivele IoT ar putea fi folosite pentru a monitoriza starea unui bun care a fost transportat pe o distanta mai lunga, iar câteva caracteristici care ar putea fi luate în calcul ar fi temperatura si umiditatea la care a fost expus si daca a fost bruscat. Fiind în continua dezvoltare, momentan sistemele distribuite nu pot fi incorporate în dinamica tranzacțiilor din societate, dar cu timpul din ce în ce mai multe probleme de Securitate vor fi remediate.

Bibliografie

[1] Proof of Work

<https://www.investopedia.com/terms/p/proof-work.asp>

[2] Proof of Stake

<https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/>

[3] Proof of Capacity

<https://www.naukri.com/learning/articles/proof-of-capacity-in-blockchain/#:~:text=protocols%20by%20PoS.-,What%20is%20Proof%20of%20Capacity%20(PoC)%3F,%2C%20which%20stakes%20miners%27%20coins>.

[4] Registre distribuite

<https://hedera.com/learning/distributed-ledger-technologies/what-are-distributed-ledger-technologies-dlts>

[5] Începutul contractelor inteligente

<https://www.investopedia.com/terms/s/smart-contracts.asp>

[6] Bazele contractelor inteligente

<https://www.techtarget.com/searchcio/definition/smart-contract>

[7] Avantajele contractelor inteligente

<https://appinventiv.com/blog/smart-contract-guide/>

[8] Atacul cibernetic DAO

<https://www.gemini.com/cryptopedia/the-dao-hack-makerdao#section-origins-of-the-dao>

[9] Contracte inteligente

<https://ethereum.org/en/developers/docs/smart-contracts/security/>

[10] Cele mai bune practici ale contractelor inteligente

<https://101blockchains.com/ethereum-smart-contract-best-practices/>