Dalla macchina di Babbage alla Ethereum Virtual Machine di Buterin

Cabling & Wireless DIGITAL NEWS

Novembre 2021

Dalla macchina di Babbage alla Ethereum Virtual Machine di Buterin

[a] *Antonio Josè Di Scala*
[bc] *Marcello Maggiora*

La storia dell'informatica è relativamente recente. I primi computer nascono intorno al 1930 principalmente per usi militari e fino ai primi anni '70 il calcolatore non esce dalle grandi organizzazioni e dai C.E.D. per iniziare il suo viaggio verso la scrivania dell'utente. Meno di 100 anni di storia che hanno rivoluzionato ogni settore produttivo e attività umana.
La ricerca verso un'automazione del calcolo ha però origini molto più lontane. L'esigenza di far eseguire lunghi e ripetuti calcoli ad una macchina piuttosto che all'uomo, nasce secoli prima, così come alcuni concetti fondamentali che ancora oggi sono alla base della matematica e delle architetture dei calcolatori.
I tanti progetti, pensieri e intuizioni nel tempo si sono intrecciate, modificate, fuse, fino a creare le basi sulle quali la scienza moderna ha costruito il mondo degli elaboratori di oggi.

Uno dei primi passi fu quello di inventare la cifra Zero. Un numero considerato alla stregua degli altri ma che ha avuto un processo di maturazione molto più complesso delle altre cifre. La storia dello Zero è lunga e affascinante e, per quanto articolata e per certi tratti incerta, gli storici concordano sul fatto che sia stata inventata (o forse scoperta) dal matematico indiano **Brahmagupta** (598-668) nel 600 d.C..
Pare infatti che Brahmagupta sia stato il primo ad utilizzare la cifra Zero associata al valore posizionale della cifra, cioè in modo tale da attribuire un diverso valore alle altre cifre in base alla sua posizione.

Lo zero fu portato in Italia da Leonardo Pisano detto Fibonacci nel 1202 e fu poi diffuso in tutto l'occidente.

Dobbiamo poi ricordare **al-Khwārizmī** (780-850), da cui deriva per altro il termine *algoritmo*. Diede un contributo fondamentale all'Algebra, scrisse l'opera 'Algoritmi de numero Indorum' intorno all'825 e fu probabilmente la prima opera completa sulle tecniche di numerazione indiane con la quale e contribuì alla loro diffusione prima in Oriente e poi in Europa.
Muovendoci rapidamente nella storia arriviamo a **Ramon Llull** (1232-1316), italianizzato in Raimondo Lullo, ebbe un ruolo rilevante nell'Europa del '200. Nella sua opera l'Arte, Llull descrive come affrontare ogni possibile problema utilizzando la tecnica della scomposizione. In pratica procedeva attraverso una riduzione della questione in parti sempre più piccole, più semplici, fino ad arrivare alle lettere dell'alfabeto. Le lettere poi facevano parte di ruote in grado di fornire infinite combinazioni. Nonostante la descrizione rapida e sommaria, non è difficile percepire i concetti di base attraverso i quali vengono affrontati i problemi con i calcolatori odierni e immaginare come le lettere del 1200 possano essere i bit di oggi.
Arriviamo al '600 e all'opera di **Blaise Pascal** (1623-1662) il quale inventò quella che fu considerata la prima calcolatrice. Blaise Pascal nell'osservare il padre, contabile, fare lunghi calcoli decise di aiutarlo e così progettò e realizzò diversi prototipi della macchina che venne poi chiamata la *pascalina*.

La pascalina ebbe una grande risonanza perché citata nella famosa *Encyclopédie* di Denis Diderot e Jean-Baptiste Le Rond d'Alambert.
Pascal ottenne poi da Luigi XIV il privilegio in esclusiva di produrre la macchina calcolatrice che fu realizzata in legno e metallo. La calcolatrice essendo pensata per eseguire operazioni contabili non lavorava su base 10 ma secondo l'unità della moneta del tempo, la Lire. Una Lire era costituita da 20 soldi formati a sua volta da 12 denari e in questo modo erano organizzate le diverse ruote: denari, soldi, ecc.
La macchina era in grado di eseguire solo addizioni e sottrazioni ma l'aspetto più interessante della pascalina fu l'ideazione di un sistema automatico per la gestione del riporto risolto attraverso un collegamento tra le ruote numerate in modo tale che un completo giro di una ruota facesse scattare di una posizione la ruota alla sua sinistra. La geniale soluzione si scontrò con la capacità delle lavorazioni dei materiali, infatti la calcolatrice soffrì di molti problemi dovuti alla complessa meccanica.

A Pascal fece seguito l'importante contributo di **Gottfried Wilhelm Leibniz** (1646-1716). Leibniz infatti pensava ai numeri come un linguaggio universale e sosteneva che è possibile rappresentare ogni altro numero con una serie di zeri e uni. Inventò di fatto il sistema di numerazione in base 2 cioè il calcolo binario. Nel 1703 Leibniz pubblicò l'articolo "Explication de l'Arithmétique Binaire" dove spiegò le basi del sistema binario con vari esempi pratici di addizione, sottrazione, moltiplicazione e divisione. In seguito anche Leibniz costruì una calcolatrice, chiamata Macchina di Leibniz, ampliando di fatto l'approccio seguito qualche decennio prima da Pascal.
La Macchina di Liebniz era in grado di eseguire tutte e quattro le operazioni (la pascalina solo addizione sottrazione) e molti aspetti della macchina di Pascal furono migliorati e potenziati significativamente. Le lavorazioni meccaniche erano tuttavia ancora deboli e anche la macchina di Leibniz soffrì degli stessi problemi che afflissero la calcolatrice di Pascal.

Nell'approdare alle soglie del '800 osserviamo come idee, concetti, oltre a qualche primo tentativo pratico, crearono le condizioni affinchè uno scienziato, di nome **Charles Babbage** (1791-1871), progettasse la prima macchina da calcolo programmabile ed acquisisse poi nel tempo il titolo di primo informatico della storia.

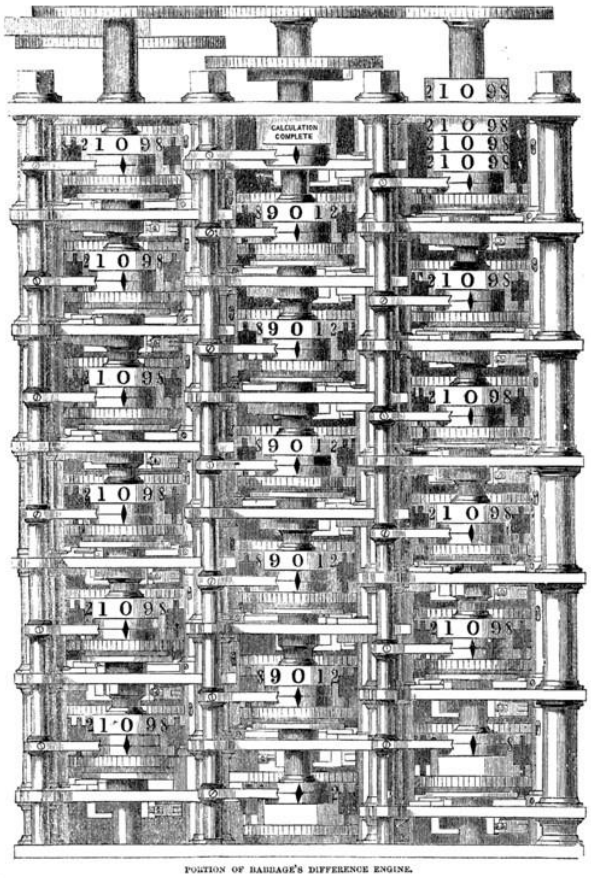


Fig. 1 - Macchina differenziale di Charles Babbage (Fonte Babbage difference engine drawing.gif – Wikipedia)

Inquadrare la figura di Babbage non è semplice, fu un matematico che potremmo definire eccentrico e poliedrico. Si occupò di tavole logaritmiche, di fisica, di meccanica, di filosofia, di ferrovie e anche di assicurazioni. Sembra che anche l'introduzione dei francobolli postali derivi da suoi studi sul costo del trasporto della corrispondenza. Alla costante ricerca della perfezione intravedeva spesso in ogni progetto una nuova evoluzione prima ancora di terminare la precedente. Questo atteggiamento da un lato lo spinse a migliorare sempre più le sue idee, dall'altro fu un ostacolo non da poco quando si trattava di ottenere dei finanziamenti. Gli interlocutori infatti con difficoltà si facevano convincere a finanziare progetti che sarebbero stati probabilmente interrotti in favore di nuove idee, anche se migliorative.
Indomabile curioso aveva un profilo energico e piuttosto lontano dallo stereotipo dell'epoca del professore di matematica. Fu più preoccupato di studiare, scoprire, indagare che di pubblicare e comunicare i risultati del suo lavoro.

Forse per questa ragione la sua figura, per quanto centrale nello sviluppo dei calcolatori, a lungo è stata poco nota.
Charles Babbage e il primo calcolatore, sono particolarmente legati alla città di Torino. Babbage fu infatti invitato alla prestigiosa Accademia delle Scienze di Torino per il secondo Congresso Nazionale degli Scienziati italiani del 1840 proprio per presentare la sua macchina calcolatrice. L'allora Presidente dell'Accademia, Giovanni Plana (astronomo e matematico) e gli scienziati presenti al congresso, furono affascinati dalle idee di Babbage, così come lo fu il re di Sardegna Carlo Alberto che lo ricevette diverse volte.

La prima macchina calcolatrice progettata da Babbage si chiamava Macchina alle Differenze e si basava sul metodo delle differenze finite per calcolare funzioni polinomiali da cui derivò l'evoluzione che, presentò appunto al congresso di Torino, chiamata Macchina Analitica.
Lo scienziato inglese non riuscì mai a costruire l'intera Macchina Analitica ma i disegni dei progetti furono ritrovati dopo molti anni dietro ai molti volumi custoditi nella grande sala dei Mappamondi dell'Accademia delle Scienze di Torino dove sono ancora oggi custoditi.

La Macchina Analitica rappresentò una vera innovazione poiché introdusse diversi concetti ancora oggi alla base dei calcolatori. Si trattava di una macchina programmabile in grado affrontare diverse tipologie di problemi sulla base di un programma utente. Il concetto di programmazione era una enorme novità. Il programma utente veniva descritto su schede di cartone di cartone traforato (collegate tra loro come una sorta di nastro) in grado di fornire tutte le informazioni necessarie alla macchina per svolgere il problema specifico.
In effetti la descrizione del programma avveniva attraverso tre tipologie di schede: una dedicata alle operazioni, una per le variabili ed una per i numeri. Introdusse il concetto di microcode cioè quella parte di codice che prevede tutta una serie di istruzioni ricorrenti come ad esempio le istruzioni per il calcolo del fattoriale o della radice quadrata. Una serie di istruzioni programmabili che nei calcolatori moderni sono mantenute in memorie di tipo Eeprom oppure in memorie a sola lettura.

La Macchina Analitica prevedeva a questo scopo un tamburo, su cui attraverso una serie di tacche e opportuni manovellismi, venivano codificate appunto le operazioni di base più frequenti.

Fig. 2 – La grande Sala dei Mappamondi dell'Accademia delle Scienze di Torino
(Fonte : Accademia delle Scienze di Torino)

Il progetto di Charles Babbage prevedeva due livelli di programmazione, uno utente ed uno di sistema, un'impostazione oggi non sorprendente ma straordinaria però nella seconda metà dell'800. Un'impostazione quella della Macchina Analitica, seppur basata solo su soluzioni meccaniche, che anticipa le architetture moderne con soluzioni concettuali, maturate ed evolute ma non così diverse nelle implementazioni dei moderni calcolatori elettronici.
Nel concedarci dal primo informatico della storia dobbiamo ricordare anche **Augusta Ada Byron lady di Lovelace** (1815-1852), figlia del poeta Lord Byron. Ada Lovelace era una matematica, profilo per altro non solito per una donna di quei tempi, e quando conobbe Charles Babbage rimase entusiasta delle sue idee tanto che tradusse l'accurata descrizione della Macchina Analitica scritta a Torino da Federico Luigi Menabrea, collaboratore di Plana, e alcuni articoli contribuendo anche con molte sue idee alcune delle quali forse ancora più innovative di quelle di Babbage.
In un articolo suggeriva ad esempio che attraverso il principio della Macchina Analitica si potessero elaborare non solo numeri ma molte altre tipologie di problemi scientifici di qualsiasi livello di complessità. Nella traduzione di un articolo tra le note aggiunte da Ada Lovelace comparve anche la prima sequenza scritta di un programma pensato per risolvere alcuni problemi matematici. Con questo articolo Ada Lovelace si guadagnò a sua volta il titolo di prima programmatrice della storia tanto che anche un linguaggio di programmazione fu battezzato Ada proprio in suo onore.

Il tessuto di conoscenze ed esperienze sviluppato nel corso del tempo, i progetti di Babbage e lo sviluppo tecnologico, hanno consentito di imprimere un'accelerazione al settore che ha portato in poche decine di anni a sviluppare i primi grandi calcolatori elettromeccanici.
Nella prima metà del '900 altre due figure fondamentali segnarono lo sviluppo dei calcolatori: von Neumann e Turing.

John von Neumann (1903-1957), è stato uno dei grandi matematici della storia e contribuì con il suo lavoro allo sviluppo di molti settori della scienza. Con la teoria dei giochi applicò la matematica ai problemi dell'economia e delle scienze sociali, contribuì allo sviluppo della Meccanica Quantistica, alla Teoria degli Insiemi, all'Analisi Funzionale e anche allo sviluppo della Meccanica Quantistica, alla Teoria degli Insiemi, all'Analisi Funzionale e anche allo sviluppo del calcolatore, a cui diede un contributo fondamentale, non tanto dal punto di vista tecnologico, quanto dal punto di vista concettuale. Definì infatti la logica da cui deriva l'architettura von Neumann utilizzata ancora oggi negli elaboratori.

Tutto ebbe inizio con un incontro fortuito nell'estate di 1944 tra Herman Goldstine, e John von Neumann presso il binario della stazione ferroviaria di Aberdeen (Marylan, USA). Goldstine aveva lavorato al progetto dell'ENIAC, il primo calcolatore *general purpose* della storia. La capacità di von Neumann di comprendere problemi e tutte le questioni coinvolte quasi istantaneamente, era leggendaria tanto che un suo amico di infanzia, Eugene Wigner, premio nobel per la fisica nel 1963 scrisse di von Neumann: "Conoscendo von Neumann mi sono reso conto di quale sia la differenza tra un matematico di primo livello ed uno come me". John von Neumann aveva infatti una fama quasi mitica anche tra i colleghi tra i quali godeva di una enorme stima.
Ritornando all'incontro con Goldstine, il contributo di von Neumann fu determinante a definire l'architettura dei primi calcolatori. Nel progetto Eniac infatti il programma era cablato e i dati inserite attraverso schede perforate. Messo in contatto con il team di Goldstine, in particolare con Mauchly and Eckert, von Neumann capi che i dati e i programmi dovevano essere inseriti e memorizzato nello stesso modo nel calcolatore, approccio sul quale si sviluppo l'architettura ancora oggi alla base dei calcolatori chiamata appunto Architettura von Neumann.

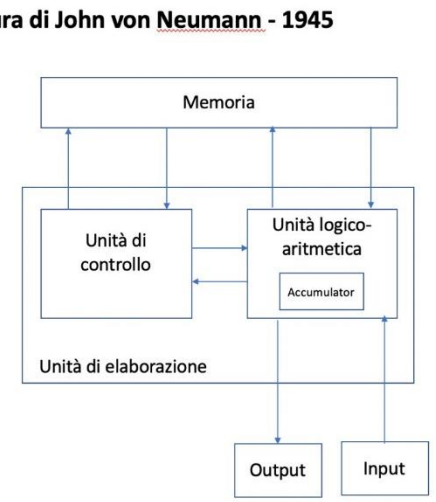


Fig. 3 – Schema dell'Architettura di von Neumann

Ed è la volta di un altro grande matematico **Alan Mathison Turing** (1912-1954) che nello stesso periodo contribuì con altrettanta importanza alla storia dell'informatica e del calcolatore. Turing durante la seconda guerra mondiale fu uno dei matematici che lavorò alla decifrazione dei messaggi codificati con Enigma la macchina cifrante messa a punto dei tedeschi. Il team di Bletchley Park, il centro della crittoanalisi britannico, comprendeva molti scienziati tra i quali Turing e Max Newmann.
Per affrontare il complesso problema della decifrazione, ricordiamo che Enigma era in grado di generare chiavi di codifica (tendo conto di tutte le possibili opzioni) nell'ordine di 105.456 x 100.391.791.500, l'unico modo per tentare di affrontare l'analisi del codice era utilizzare un sistema di calcolo molto più rapido di quanto potesse fare un uomo. Turing partì dal lavoro dei crittoanalisti polacchi, in particolare dalla macchina progettata agli inizi degli anni '30 da Marian Rejewski denominata Bomba. Il progetto fu sviluppato e vide la luce una macchina molto più efficace della Bomba e sulla base di questa evoluzione Max Newmann a Bletchley Park progetto il Colossus, realizzato per la fine del 1943.

Il lavoro di Turing influenzò in modo significativo lo sviluppo dei calcolatori e più in generale quello dell'informatica. Formalizzò infatti i concetti di algoritmo e calcolo attraverso quella che venne chiamata la Macchina di Turing, modello che assunse un ruolo centrale nella creazione del computer moderno, per poter studiare e rispondere matematicamente al problema *Entscheidungsproblem* posto dal matematico David Hilbert. Simultaneamente vengono proposti altri modelli come il Lambda Calcolo, Funzioni Ricorsive, Macchine di Post, ecc. che risultano tutti equivalenti tra loro cosa che da nascita al concetto di macchina di Turing Completa. In parole povere una macchina fisica o ideale che possa eseguire qualsiasi algoritmo è detta Turing Completa.
La completezza di una macchina di Turing è riconducibile alla possibilità di programmare loop con istruzioni del tipo "if … then …" e "go to". Una affermazione famosa nota come Tesi di Church-Turing sostiene che qualsiasi algoritmo può essere eseguito su una Macchina di Turing.
Alan Mathison Turing, tra le altre cose, anticipò lo sviluppo dell'intelligenza artificiale ipotizzando negli anni '30 che nel 2000 si sarebbero potute creare delle macchine in grado di riprodurre il funzionamento della mente umana.

Dalla seconda guerra mondiale la corsa ai calcolatori fu in continua accelerazione. Dopo gli elaboratori americani, nel 1961 arriva il primo calcolatore italiano, la CEP (Calcolatrice Elettronica Pisana): progettato su suggerimento di Enrico Fermi e con il supporto di Adriano Olivetti, sotto la supervisione di Mario Tchou. Sono gli anni del Prof. Gianfranco Capriz dell'Università di Pisa e del Rettore Prof. Alessandro Faedo. Poi nel 1971 Federico Faggin inventa il microprocessore presso la neonata Intel. Faggin fonda poi la Zilog progettando il microprocessore Z80.
Ormai tutto è pronto per l'era dell'informazione.

Nel 1986 dal CNUCE (Centro Nazionale Universitario di Calcolo Elettronico), poi istituto del CNR, viene inviato il primo *ping* verso gli Stati Uniti. Nel 1989 è la volta del World Wide Web per mano di Tim Berners Lee e così poco alla volta la capacità computazionale e la rete diventa un alleato quotidiano di ogni persona e non solo più motore della scienza.

Arriviamo così a **Vitaly Dmitriyevich (Vitalik) Buterin** (classe 1994), che ha rivoluzionato il concetto di elaboratore e di macchina di Turing, così come lo abbiamo sempre interpretato.
Siamo infatti abituati a pensare ad un computer come un apparecchio fisico, più o meno grande, un frigorifero, un televisore, un divano, etc. Questa idea deriva dai *padri fondatori* dell'informatica: a partire da Llull fino a von Neumann passando da Leibniz e Babbage, si è sempre alimentata l'idea della costruzione d'una macchina per eseguire algoritmi. La macchina fisica viene fabbricata e provvista di quello che viene chiamato linguaggio macchina. Utilizzando i compilatori, cioè un programma capace di tradurre linguaggi di più alto livello come il Fortran, il Pascal o il C, in istruzioni di linguaggio macchina il codice utente viene eseguito. Questo lo scenario a cui siamo abituati.
Nel whitepaper del 2013 "Ethereum Whitepaper", Vitalik Buterin propone però una macchina di Turing completa il cui supporto fisico è una rete di nodi, cioè la rete Ethereum e si avverte un cambio di scenario significativo.

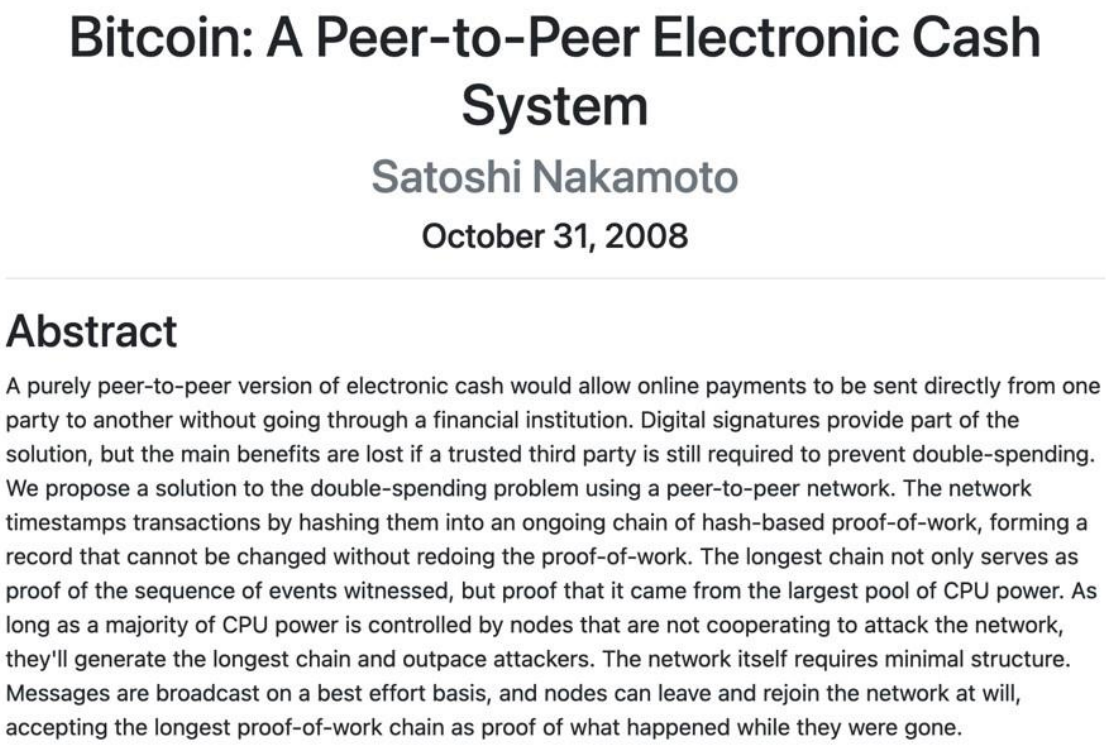
Facciamo qualche passo indietro per inquadrare meglio il contesto che riguarda gli elementi che entrano in gioco, le Crittovalute e le Blockchain. Nel 2008 **Satoshi Nakamoto** creò simultaneamente la tecnologia blockchain e la sua prima applicazione, la crittomoneta bitcoin. La figura di Nakamoto è piuttosto misteriosa, non si sa neppure se si tratta di un nome vero o di uno pseudonimo. Nel 2008 Satoshi Nakamoto scrive ad una lista di crittografia annunciando di aver di aver messo a punto una moneta elettronica che funziona su una rete peer to peer senza la necessità di una terza parte.
La terza parte nel sistema monetario centralizzato attuale è sempre necessaria, ed è di solito rappresentata da uno Stato che garantisce la moneta o da banche delegate appunto dallo Stato.

Nakamoto annuncia che rilascerà il codice e così fa nel gennaio 2009. Nel 2010 Satoshi Nakamoto scompare nel nulla possedendo circa 1.000.000 di Bitcoin, circa 4 miliardi di USD, patrimonio per altro mai utilizzato.

La sua uscita di scena avviene in modo progressivo e in circostanze che non sembrano potersi ascrivere ad una scomparsa prematura.

La Blockchain è una sorta di libro mastro di transazioni distribuito nella rete secondo la tecnologia peer to peer. Si tratta di un registro accessibile a chiunque, immediatamente, che tiene traccia di tutte le transazioni eseguite sin dalla nascita della moneta bitcoin. Le transazioni sono contenute in blocchi che vengono inseriti come pagine del libro mastro utilizzando la crittografia.
Perciò il registro non è modificabile ed essendo distribuito non può essere distrutto.

La rete Bitcoin (normalmente con la B maiuscola si intende il protocollo mentre con b minuscola la moneta) è anche una macchina di Turing ma non completa perché il linguaggio di programmazione non permette i loop. Satoshi Nakamoto ha infatti escluso questa possibilità per proteggere la rete da attacchi di tipo DoS (Denial of Service).



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto

October 31, 2008

Abstract

A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Fig. 4 – Il paper di Satoshi Nakamoto che ha aperto l'era della crittomoneta

Arriviamo ora all'idea di Buterin. Ethereum è in effetti un calcolatore che non esiste fisicamente come unità tradizionale ma è distribuito nella rete e può essere ovunque dove è disponibile Internet. Non può essere fermato, bloccato o censurato. E' una piattaforma decentralizzata del web 3.0 e si basa sulla tecnologia Blockchain. Se vogliamo fare un parallelo con le architetture classiche, il codice macchina (microcode) è il bytecode di EVM (Ethereum Virtual Machine) un linguaggio di programmazione stack-based il cui funzionamento ricorda la notazione polacca inversa.

A livello superiore troviamo i linguaggi come Serpent, LLL (un tipo di LISP), Solidity e altri ancora. Sono inoltre disponibili anche tools più amichevoli che permettono di programmare la EVM in modo abbastanza semplice. Secondo la falsariga dell'architettura di von Neumann, il programma, scritto in bytecode EVM, è memorizzato in un blocco della Ethereum Blockchain.

Ma come si coniuga una Blockchain con una Macchina di Turing e come si esegue un programma? Dove vengono salvati i risultati? I dubbi e le domande possono essere tanti.
Per eseguire un programma in EVM sono necessarie delle risorse. Si tratta di un approccio nuovo. Il vincolo di pagare in qualche modo per l'esecuzione di un programma protegge il sistema da attacchi di tipo DoS. Una tecnica proposta già nel 1992 per affrontare il problema delle junk mail, se vogliamo in qualche modo simile al Denial of Service.

Nel paper "Pricing via processing or combatting junk mail" Cynthia Dwork e Moni Naor proponevano proprio di applicare un costo ad ogni mail inviata al fine di bloccare il fenomeno delle mail indesiderate che, se inviate in grande quantità, sono un attacco DoS mirato al sistema di posta elettronica.

Naturalmente il costo non necessariamente è denaro, si tratta più di un concetto di risorse finite. Per l'esecuzione di un programma EVM si ricorre al concetto di *gas*. Per comprendere meglio il funzionamento pensiamo all'automobile e al carburante necessario per farla muovere. L'analogia è con il programma EVM è forte, l'auto non si muove senza carburante come il programma EVM non può essere eseguito senza gas.

In modo analogo un programma EVM per muoversi nella rete ha bisogno di gas e per acquistare gas si usa la crittomoneta ETH che nasce insieme alla rete Ethereum.
Un programma EVM deve perciò controllare di disporre di sufficiente gas per consentire all'algoritmo di giungere al termine.

Infine un accenno a come si salvano i dati in ambiente EVM. Se tutti abbiamo presente i File System distribuiti su più nodi di un Cluster, anche di tipo stretched, il concetto qui viene esteso con l'IPFS (InterPlanetary File System) il cui acronimo dice già molto sulle caratteristiche che si sono volute dare al File System.
Si tratta di uno spazio di archiviazione distribuito decentralizzato pensato per Ethereum ma che sta estendendo il suo ambito di applicazione.
E' un FS completamente cifrato basato sulla memorizzazione di piccoli chunks nei quali viene scomposto il file da archiviare. Ogni nodo della rete che utilizza IPFS mantiene solo i dati *utili* al nodo più alcune informazioni di indicizzazione.
Ogni unità di memorizzazione è identificata con un crypto hash per cui la sostituzione del file produce un hash differente, una sorta di fingerprint.
In questo modo un file non è sostituibile in modo trasparente come avviene sui File System tradizionali poiché il fingerprint cambiarebbe.

I programmi EVM sono spesso conosciuti come Smart Contracts. Esempi di applicazioni sono i tokens, pagamento sotto condizioni come HTLC (Hash Time Locked Contracts) e Atomic Swap (scambio sicuro di criptovalute tra differenti blockchain) che permettono di programmare i pagamenti ponendo condizioni su modi e tempi.

Molti di questi servizi sono per altro già disponibili su piattaforme dedicate.

Lo scenario è certo cambiato moltissimo da quando Babbage ha progettato il primo calcolatore programmabile. Le figure storiche citate non sono chiaramente un elenco esaustivo, molti altri hanno contribuito alla *costruzione* del mondo attuale, non ultimo Konrad Zuse che nel 1941 realizzò il primo calcolatore Turing-complete.

Con un perimetro così vasto l'impatto sulle sovrastrutture organizzative di dominio, ma anche sul contesto politico/economico, può trasformarsi in un vero terremoto tanta è la differenza tra modelli che, per quanto perfezionati, risalgono a centinaia di anni fa e lo scenario che queste tecnologie ci fanno intravedere.

La complessità oggi è forse aumentata rispetto al passato ma il punto fondamentale, il cambio epocale potremmo dire, riguarda i confini e l'ambito di applicazione. Oggi il perimetro di un modello come quello di Ethereum è globale, il limite è determinato solo dalla disponibilità della rete, il vero collante di ogni tecnologia digitale. ✓

[a] DISMA, Dipartimento di Scienze Matematiche, Politecnico di Torino, c.so Duca degli Abruzzi 24, 10129 Torino
antonio.discala@polito.it

[b] Compagnia di San Paolo Sistema Torino, p.zza Lorenzo Bernini 5 - 10138 Torino
[c] CNR-IEIIT, Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni, c.so Duca degli Abruzzi 24, 10129 Torino
marcello.maggiora@ieiit.cnr.it