# 區塊鏈完全解析

Author: 陳冠儒 ( Darren K.J. Chen ) / 版權所有，未經授權請勿複製使用

---

【序】近年來，比特幣的出現顛覆大家對於貨幣的想像，造成極大的轟動與改變，而其所使用的就是區塊鏈的技術。但大家卻常常以為比特幣＝區塊鏈，其實這個想法是錯誤的，事實上如同 Bitcoin Chaser 創始人，Marc Kenigsberg 所說：「區塊鏈是種技術。比特幣僅僅是當中第一個有潛力的主要應用」(Blockchain is the tech. Bitcoin is merely the first mainstream application of its potential - Marc Kenigsberg[1])。也就是說，區塊鏈這項技術不僅僅能應用在加密貨幣(Cryptocurrency)的實現上，諸如證書發行、認證追蹤、打造智慧城市(Smart City) …… 等都能用得上它。

【架構篇】那麼區塊鏈究竟是什麼呢？區塊鏈的架構其實很好理解，我們可以試著先畫一個表格，如[圖 1]，圖中我們把每一縱列當作是一個最基本區塊，每一個區塊會有最基本的五項要件，對應到縱列的第一到第五格，每一格都有其需要扮演的角色。第一格：索引(Index) ，即區塊的編號，每個區塊都會有一個編號，區塊會依照其產生的順序而產生編號，這個編號我們就會存在每個區塊的 Index 欄位(Field)中。第二格：隨機數(Nonce)，顧名思義，即一個隨機數，為什麼要用到隨機數，我們等一下談到挖礦的時候會提到。第三格：資料(Data)，顧名思義，每個區塊上都會有一個專門記錄資料的欄位，像是加密貨幣可能就是紀錄交易資訊(如[圖 1])、證書鏈可能就是紀錄被發行者的資訊、病歷鏈可能就是紀錄患者病歷、病史 …… 等等。第四格：哈希值(Hash Value)，一個獨一無二，只屬於這個區塊的一長串通常以 16 進位表示的數值，至於為甚麼獨一無二我們等一下談到挖礦的時候會提到。第五格：前哈希值(Previous Hash Value)，顧名思義，即前一個區塊的哈希值，為甚麼需要放上一區塊的哈希值，同樣等之後介紹哈希值時會一並講解。以上就是一個區塊最基本的架構，只要了解這五格，你就知道如何組成一個最基本的區塊了！而回歸原題，區塊鏈究竟是甚麼呢？我想大家都已

Blockchain Struc.

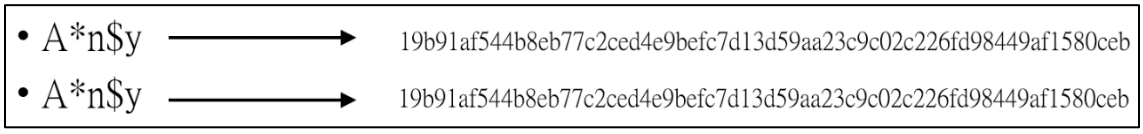
| | Block 1 | Block 2 | Block 3 | Block 4 | Block 5 | ..... |
|---|---|---|---|---|---|---|
| Index | 01 | 02 | 03 | 04 | 05 | ..... |
| Nonce | 11316 | 35230 | 12937 | 35990 | 56265 | ..... |
| Data | A -> B: $50 | B -> C: $30 | C -> A: $20 | B -> A: $20 | C -> B: $10 | ..... |
| Hash | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 12834fec6 6a73e78e6 9a14874ce af97e0626 df4c3783f1 575d00c00 2c0eea24f | 10874f3be e50bcf516 31eab028b 5e8d040f3 c06564d42 213cd944b 53e6950cb 1 | 30b2c2549 774b7bc57 45ac78558 c6a0e795d d0789d58b 301adb02c 6c1b6d43f 7 | ..... |
| Pre. Hash | 000000000 000000000 000000000 000000000 000000000 000000000 000000000 0 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 12834fec6 6a73e78e6 9a14874ce af97e0626 df4c3783f1 575d00c00 2c0eea24f | 10874f3be e50bcf516 31eab028b 5e8d040f3 c06564d42 213cd944b 53e6950cb 1 | ..... |

【圖 1】

[1] 民國 109 年 04 月 24 日，取自：https://twitter.com/marckenigsberg/status/768976469951406081?s=20

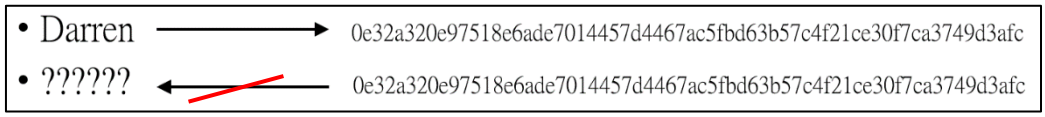
經猜到了，其實就是如[圖 1]，我們知道每一縱列就是一個區塊，如此我們只要把每一區塊(縱列)按照順序相連，就形成一個最基本的區塊鏈架構了[2]。

【挖礦篇】區塊鏈的架構講完，我們就可以來討論挖礦(Mining)了，在比特幣的世界中我們只知道「挖礦」可以獲得比特幣，卻不知道為甚麼，在此要讓大家先認識一個新名詞：「共識算法(Consensus Algorithms)」，區塊鏈最大的特色是去中心化(Decentralization)，所謂的去中心化指的是在區塊鏈系統中並不是由少數人或是一個組織、機構所管理的，而是所有餐與在區塊鏈上的使用者大家共同管理與監督，而為了讓所有參與者都能夠共同來管理跟監督，大家就需要有一個「共識」才能運作下去，必須要有一種「方法(Method)」，讓大家使用區塊鏈都必須遵循區塊鏈的共識規則，才能維持整個區塊鏈運作不會亂掉，因此在區塊鏈技術的世界中，我們就用演算法(Algorithms)作為這個方法，於是就產生了「共識算法」。共識算法有很多種，像是 PoW, PoS, DPoS …… 等都是共識算法的一種，而目前大家所聽過的像是比特幣、乙太幣 …… 等都是使用 PoW 的共識算法，因此現階段我們只需要先了解 PoW 簡單的原理即可，剩下的我們之後有機會可以繼續討論。了解完共識算法，接著我們可以開始講解挖礦到底該怎麼挖了，PoW (Proof of Work) 算法又稱為「工作量證明」算法，重點在於「工作量(Work)」，在字面上就可得知比的是誰的工作量多，是不是聯想到比特幣挖礦了呢？誰挖越多，賺的就越多，沒有錯就是這樣的概念，那麼問題來了，做甚麼「工作」呢？這裡就要再介紹一個名詞了：「哈希值(Hash Value)」，又稱為雜湊值，這是甚麼呢？要講哈希值，就必須先了解哈希是甚麼？我會這麼形容它：「一個非常龐大的、且封閉的對照表」，你們可以認為這個對照表的對應是透過非常複雜的計算產生的，並不是已經規定好的，因此這張表目前無人可取得或是破解。在哈希的對照表上面，你能想像到的任何文字、符號(包含空格)或檔案(技術上檔案也可轉換為文字)都會有一個唯一的對應值，這就稱作該文字、符號或檔案的「哈希值」，如下圖，同一明文，無論何時何地你用完全




相同的一段文字，所得到的哈希值也一定會一樣，除此之外，哈希值還有以下的特點需要注意：

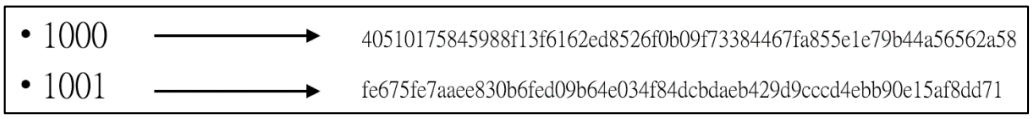
1. 它是一串 256 位的 2 進位數字，因數字太大、太長了，故常將其轉換為 16 進制數字表示之。(註：適用於 SHA-256 哈希函數)
2. 單向函數(One-Way Function)：常有人會講哈希值尋找對應的一個方式(即從哈希表中尋找對應哈希值的方式)誤解為一種加密方法，但其實這是錯誤的，在密碼學的定義當中，所謂密碼必須是要能夠被解密的，而哈希值不一樣，它並不是加解密的關係，而是對應的關係，如下圖，明文(Plaintext)與哈




[2] 區塊鏈架構模擬器：https://andersbrownworth.com/blockchain/blockchain

希值的轉換就像是我們把水果透過果汁機打成果汁，打成果汁後我們就無法將其變回原來的水果，從而得知原來的水果長甚麼樣子了，哈希的轉換同樣是如此，我們一旦把一段文字(明文)透過哈希函數(Hash Function)[3]轉換成哈希值之後，我們就無法通過其他的任何手段將其轉換回原來的那段文字。

3. 偽隨機(Pseudorandom)：哈希值並非隨機數，因為隨機數會重複，哈希值不會，但卻有著類似之處，哈希值的轉換是沒有任何規律的，如下圖 ，我們




可以看到<1000>與<1001>兩段「文字」(記住，我們將其當作文字看待，而非數字) 雖然相近，但所得到的哈希值卻是完全不同的。

大家知道了哈希值的概念，再來看 PoW (工作量證明) 就容易多了，前面說了工作量證明就是比大家的工作量，我們也猜到了所謂的工作就是挖礦，那怎麼挖呢？外面常常在講挖礦就像是在算數學題目，比誰算的多就得到最多的獎勵，但其實筆者覺得這個比喻並不是非常恰當，因為實際上挖礦就像是簽樂透，靠的是運氣，為甚麼這麼說呢？還記得之前講到一個區塊有五個最基本的組成要件嗎？如[圖 2]，當時提到了隨機數(Nonce)及哈希值(Hash Value)，但當時並未細講，其實這裡的哈希值就是將一個區塊除了哈希值以外的其他欄位合起來成一長串文字，並將其轉換成哈希值所得到的結果，要注意的是它會連前一個區塊的哈希值(也就[圖 2]中每一個縱列的第五格)一起轉換，這樣一來只要有人對前面的區塊進行竄改，該區塊的哈希值就會變動，從而導致後面所有區塊的哈希值皆對不上，因此只要能夠驗證出所有區塊的哈希值前後是能對應上的，就能保證在區塊鏈上的資訊沒被竄改，但你們一定很好奇誰來找出這個哈希值？答案就是「礦工(Miner)」，礦工負責找出區塊對應的哈希值，並驗證區塊，礦工每驗證一個區塊就能獲得一定回報，以比特幣為例，比特幣礦工每驗證一個區塊現在(2020 年 04 月)可以獲得 12.5 BTC，那問題來了像是比特幣每 10 分鐘才產生一個區塊，可是產生哈

Blockchain Struc.

| | Block 1 | Block 2 | Block 3 | Block 4 | Block 5 | ..... |
|---|---|---|---|---|---|---|
| Index | 01 | 02 | 03 | 04 | 05 | ..... |
| Nonce | 11316 | 35230 | 12937 | 35990 | 56265 | ..... |
| Data | A -> B: $50 | B -> C: $30 | C -> A: $20 | B -> A: $20 | C -> B: $10 | ..... |
| Hash | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 12834fec6 6a73e78e6 9a14874ce af97e0626 df4c3783f1 575d00c00 2c0eea24f | 10874f3be e50bcf516 31eab028b 5e8d040f3 c06564d42 213cd944b 53e6950cb 1 | 30b2c2549 774b7bc57 45ac78558 c6a0e795d d0789d58b 301adb02c 6c1b6d43f 7 | ..... |
| Pre. Hash | 000000000 000000000 000000000 000000000 000000000 000000000 000000000 0 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 0000d07be fa55318e3 a45882c05 f0d3be4bd 0d00aa6fe a76e2e01f 542a4d00c 7 | 12834fec6 6a73e78e6 9a14874ce af97e0626 df4c3783f1 575d00c00 2c0eea24f | 10874f3be e50bcf516 31eab028b 5e8d040f3 c06564d42 213cd944b 53e6950cb 1 | ..... |

【圖 2】

[3] Online Hash Function: https://emn178.github.io/online-tools/sha256.html

希值並驗證卻不用這麼久，況且又有這麼多人參與，那到底比特幣是如何 10 分鐘才產生一個區塊的呢？原因就在於它增加難度了！它告訴所有礦工如果你想要成功驗證一個區塊，那麼你就必須要讓你的哈希值在 16 進制下前面 1~10 位數字必須為 0 (隨狀況調整，使其維持在約略每 10 分鐘產生一個區塊)，並增加了隨機數 (Nonce)，換句話說，就是讓礦工不斷嘗試各種隨機數不斷產生哈希值，直到前面補足了足夠的 0，這是非常吃電腦硬體計算資源的，也相當難猜對，但只要你猜對了數字，那麼你將可以直接獨得該區塊的獎勵，聽起來是不是就像是在賭博？那你一定就會問了，10 分鐘才一個，這樣一般人怎麼拚的過人家，因此就出現了礦池 (Pool)，所謂的礦池就是讓所有人把計算資源集中在一起，挖到之後依照大家貢獻的比例，分配獎勵，現在大家挖礦多採行這種模式，很少人只靠自己的。

【特性篇】到這邊就已將區塊鏈基本的組成架構講完了，接下來講解區塊鏈的五大特性，分別為：去中心化(Decentralization)、開放性、匿名性、獨立性與更強的安全性(Enhanced Security)，區塊鏈去中心化就不必多說了，沒有中心組織或是機構管理，而要達到該目的就需要有一個系統機制來讓區塊鏈能不被任何人干擾，在講挖礦時我們提到在 PoW 的共識算法，我們讓礦工去驗證區塊，而非特定的組織或是個人，而礦工也僅能驗證區塊，這就讓區塊鏈產生了「獨立性」，即運作完全與世獨立，不會因為任何單一外部因素而干擾到其運作，那大家有沒有想過，礦工驗證完區塊後，區塊資訊跑到哪裡去了？如何能夠找到？事實上，區塊鏈上的資訊是在區塊鏈上的全網路同步的，意思就是說，所有參與在區塊鏈上的人都擁有並可以知道區塊鏈上所有的資料與內容，這就是區塊鏈的「開放性」，而在沒有法律監管需求的情況下，單就技術來說，參與區塊鏈網路並不需要任何身分驗證，區塊鏈網路更無法記錄你的身分資訊，因此具有「匿名性」，最後，因為所有人都擁有區塊鏈上的內容與資訊，因此如果有人想竄改區塊鏈上的資訊，並讓其他人都相信，就變得極其困難了，在 PoW 共識算法的機制當中，碰到資訊衝突的時候，會選擇相信多的一方為真，也就是說你想竄改區塊鏈上的資訊時，你必須同時竄改區塊鏈網路上超過 50%的節點(Node)也就是參與者手上的區塊鏈資訊，並且因為每個區塊上都記錄著上一個區塊的哈希值，因此若想竄改前面某個區塊的資訊，則必須連同後面所有區塊的資訊同時一起竄改，不然會對不上，這就使得區塊鏈「安全性」相對地提升許多，因為想竄改以現行技術來說是幾乎不可能得。

【總結，運作演練】學到這邊，相信你已經完全的了解區塊鏈的結構、共識算法、挖礦以及其特性了，最後我們就要來嘗試將其拼湊起來，帶大家簡單演練一遍區塊鏈運作的過程，順便複習。

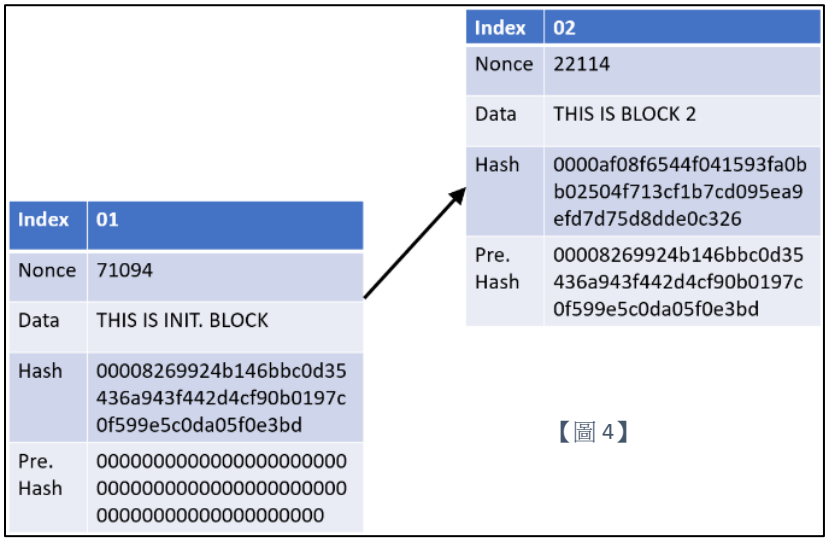
| Index | 01 |
|---|---|
| Nonce | 71094 |
| Data | THIS IS INIT. BLOCK |
| Hash | 00008269924b146bbc0d35436a943f442d4cf90b0197c0f599e5c0da05f0e3bd |
| Pre. Hash | 0000000000000000000000000000000000000000000000000000000000000000 |

【圖 3】

1. 如[圖 3]，區塊鏈要產生創世區塊(Genesis Block)即第一個區塊，因沒有上一個區塊，故前哈希值(Pre. Hash)為 0，Hash 為 Index + Nonce + Data + Pre. Hash 轉換過後之哈希值，礦工開始不斷

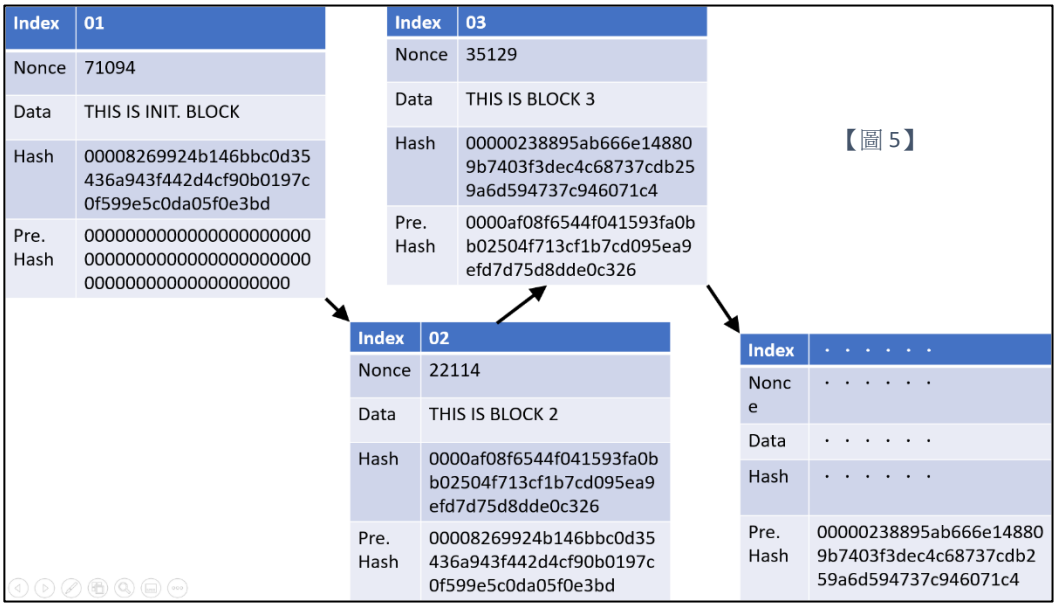
換隨機值直到出現前四碼為 0 之哈希值，後將此哈希值附上區塊，表示成功驗證，區塊資訊在區塊鏈網路上廣播(Broadcast)讓所有節點(參與者)都收到。

2. 接下來，如[圖 4]，區塊鏈要產生下一個區塊，大部分步驟都跟上一區塊的步驟相同，比較需要注意的就是 Pre. Hash 不再是 0，而是填入上一區塊也就是區塊 01 的 Hash。



【圖 4】

3. 之後便是重複上一區塊的步驟，無限地產生無數的區塊，如[圖 5]。



【圖 5】

以上，就是區塊鏈運作的過程了，如果都沒有問題那就恭喜你已經完全懂了，也非常歡迎如果有細部不懂的地方，或是任何建議都可以聯絡筆者[4]，筆者只要看到盡量都會回覆，也謝謝大家的閱讀以及支持！

[4] My E-mail Address: kjchen@protonmail.ch