DUBU4 BLOCKCHAIN SYSTEM

WHITEPAPER

DISCLAIMER

The software and technology you are about to use functions as a free, open source, and multi-signature digital blockchain and wallet.

The software does not constitute an account where the developer of this software or other third parties serve as financial intermediaries or custodians of your notes or other valuables.

While the software has undergone beta testing and continues to be improved by feedback from the open-source user and developer community, we cannot guarantee that there will be no bugs in the software.

You acknowledge that your use of this software is at your own discretion and in compliance with all applicable laws.

You are responsible for safekeeping your passwords, private key pairs, PINs and any other codes you use to access the software.

IF YOU LOSE ACCESS TO YOUR DUBU4 WALLET, YOU ACKNOWLEDGE AND AGREE THAT ANY NOTES OR OTHER VALUABLES YOU HAVE ASSOCIATED WITH THAT DUBU4 WALLET WILL BECOME INACCESSIBLE.

All transaction requests are irreversible.

The authors of the software cannot retrieve your private keys or passwords if you lose or forget them and cannot guarantee transaction confirmation as they do not have control over the DUBU4 network.

To the fullest extent permitted by law, this software is provided "as is" and no representations or warranties can be made of any kind, express or implied, including but not limited to the warranties of merchantability, fitness or a particular purpose and no infringement.

You assume any and all risks associated with the use of the software.

In no event shall the authors of the software be held liable for any claim, damages or other liability, whether in an action of contract, tort, or otherwise, arising from, out of or in connection with the software.

We reserve the right to modify this disclaimer from time to time.

당신이 사용하고자 하는 소프트웨어 및 기술은 무료, 오픈소스, 다중 서명 블록체인 및 전자 지갑입니다.

이 소프트웨어의 개발자 및 제 3 자가 귀하의 자산을 금융 중개하거나 수탁하는 역할을 수행하는 기능을 제공하지 않습니다.

이 소프트웨어는 베타 테스트를 진행중이고 오픈소스 사용자와 개발자 커뮤니티를 통한 피드백으로 계속 개선되고 있지만, 저희는 이 소프트웨어에 버그가 없다는 것을 보증하지는 못합니다.

귀하는 이 소프트웨어를 사용하는 것은 귀하의 재량이며 모든 해당 법률을 준수함을 인정합니다.

당신의 암호, 개인키, PIN 그리고 이 소프트웨어에 접근하기 위해 필요한 기타 코드들을 안전하게 보관하는 것은 귀하의 책임입니다.

만일 DUBU4 지갑에 접속할 수 있는 방법을 분실했다면, 귀하는 해당 지갑과 관련된 모든 자산들도 분실될 수 있다는 것에 동의합니다.

모든 거래 요청은 불가역적이며 되돌릴 수 없습니다.

이 소프트웨어의 개발자는 당신이 분실하거나 잊어버린 패스워드나 개인키를 복구할 수 없으며 아울러 DUBU4 네트워크에 대해 통제를 할 수 없는 관계로 거래 승인은 보장할 수 없습니다.

법이 허용하는 한도 내에서 이 소프트웨어는 '있는 그대로' 제공되며 상품성, 적합성 또는 특정 목적 및 침해하지 않았음에 대한 보증을 포함하여 (이에 국한하지 않고) 명시적인 혹은 묵시적인 어떠한 종류의 진술이나 보증도 할 수 없습니다.

귀하는 이 소프트웨어 사용에 관련된 모든 위험을 감수합니다.

어떠한 경우에도 이 소프트웨어의 개발자는 계약이나 불법행위 혹은 이 소프트웨어로 인해 직간접적으로 발생하는 모든 청구, 손해 또는 기타 책임을 지지 않습니다.

저희는 이 면책조항을 수시로 수정할 권리가 있습니다.

ABSTRACT

2015 년은 '암호 화폐(Cryptocurrency)'가 폭발적으로 성장했던 한 해 였습니다. 이전에는 단순히 비트코인의 데이터 구조로만 알려졌던 블록체인이라는 기술도 같이 주목받기 시작했습니다. 나카모토 사토시(Nakamoto Satoshi 라는 필명으로 알려진)에 의해 그의 백서에서 언급한 '블록'과 '체인' 말하자면 '연결된 블록'이라는 개념이 '인터넷의 정보의 혁명' 이래로 '가치의 혁명'이라는 새로운 시대의 도래를 촉진했습니다.

비트코인과 함께 시작된 이 시기를 Blockchain 1.0 이라고 합니다. 이때 비트코인의 프로토콜을 더욱 확장하여 블록체인을 탈 중앙화된 어플리케이션 플랫폼의 영역까지 확대한 것이 이더리움(Ethereum)이었고 사람들은 이 시기를 Blockchain 2.0 이라고 분류합니다.

2017 년까지 성공적이었던 이 두 기반 기술은 사용하는 사람이 늘어나고 블록체인의 크기가 점점 커질수록 선형 구조의 블록이 링크되는 데이터 구조만으로는 해결할 수 없는 문제들이 나타나기 시작합니다. 사람들은 새로운 패러다임이 나타날 시기라고 생각하게 되었고 많은 개발자들과 학자들이 새로운 대안을 내놓기 시작합니다. DAG(Direct Acyclic Graph) 즉 방향성 비 순환 그래프가 그 대안으로서 강력하게 제시되기 시작했습니다.

비로소 Blockchain 3.0 시대가 도래하기 시작한 것입니다.

블록체인은 이제 새로운 시대적 요구에 부응해야 할 시기가 되었습니다. '공유'와 '당사자간 직접 거래'라는 블록체인의 최대의 장점은 그 양날의 검인 비효율성을 최대한 제거하고 실제 사용할 때 요구되는 다양한 방법에 대해 포괄적이고 실존 가능한 해결책을 제시해야 합니다. 그래서 저희 DUBU4 는 블록체인에서는 이러한 블록체인에 대한 미래적 확신과 현실적 대안 사이에서 고민하고 연구했던 방법들을 제안하고자 합니다.

이 백서는 DUBU4 Blockchain 시스템에 대한 디자인 지향과 기술적 설계들에 설명입니다.

DUBU4 WHITEPAPER

1 지금까지의 블록체인

1.1 개요

비트코인에 의해 촉발된 암호화폐는 암호화폐 자체가 가져오는 제 3 자 개입이 없는 '당사자주의' 경제라는 새로운 패러다임에 대한 기대와 함께 블록체인 기술 자체에 대한 관심을 증폭시켰으며 다양한 연구와 새로운 보완 기술들이 발전하는 계기가 되었습니다.

2008 년 소개되어 2012 년 폭발적인 성장을 하며 이 당시의 블록체인 기술은 사용자들이 갖고 있는 블록체인에 대한 개념과 요구에 충분히 부응할 수 있었습니다. 그 수요자가 많지 는 않았으며 거래가 제한되어 있어 보안적 이슈나 기술적인 요구 사항도 많지 않았습니다. 몇가지 기술적인 오류들을 제외하고는 당시에는 나름 만족할 만한 기술로 등장, 자리를 잡아가고 있었습니다.

그러나 사토시 혹은 뷰테린도 생각하지 못할 만큼 빠른 속도로 관심과 기대가 증폭되고 그에 비례하여 사용성에 대한 구체적인 요구가 폭발적으로 증가하게 되자 더 이상 기존의 시스템으로는 이러한 요구를 수용할 수 없는 지경에 이르게 되었습니다.

Cryptocurrencies ▾ Exchanges ▾ Watchlist

#	Name	Market Cap	Price	Volume (24h)	Circulating Supply	Change (24h)	Price Graph (7d)
1	Bitcoin	$119,794,412,367	$6,968.52	$4,312,456,618	17,190,787 BTC	-6.66%	
4	Bitcoin Cash	$11,962,081,207	$692.44	$329,965,247	17,275,338 BCH	-5.05%	
31	Maker	$363,386,022	$543.81	$161,614	668,228 MKR *	-2.49%	
2	Ethereum	$40,906,815,357	$404.51	$1,485,430,360	101,125,776 ETH	-3.37%	
57	Mixin	$157,329,739	$356.86	$77,225	440,867 XIN *	-4.87%	
15	Dash	$1,655,421,508	$200.91	$159,129,490	8,239,567 DASH	-5.23%	
19	Zcash	$781,661,280	$173.49	$93,125,148	4,505,444 ZEC	-7.16%	
12	Monero	$1,823,253,840	$112.09	$20,681,805	16,266,706 XMR	-6.88%	
58	DigixDAO	$154,174,969	$77.09	$318,334	2,000,000 DGD *	-5.92%	
7	Litecoin	$4,204,551,962	$72.86	$264,326,109	57,706,807 LTC	-5.88%	

<그림 1. 2018 년 현재 상위 10 위 코인>

1.2 기존 블록 체인의 대표적인 문제점

① 블록체인 자체의 근원적인 문제점 - 데이터 크기와 속도의 비효율성

a 블록체인은 일렬로 늘어진 정보의 추가가 가능한 데이터베이스입니다. 원하는 모든 노드가 블록을 만들 수 있지만 채택되는 블록은 오로지 하나이며 다수의 합의에 의해 승인된 블록만이 최종 블록으로 체인에 연결될 수 있습니다. 새로운 데이터를 데이터베이스에 추가하기 위해 찾고, 검증하고, 싱크하는 번잡한 프로세스가 존재하므로 블록이 쌓이게 될수록 당연히 속도는 느려지게 되어있습니다.

b 또한, 추가만 가능한 데이터 구조로 인해 블록체인 자체의 사이즈는 계속 커지게 되고, 모든 노드들은 이러한 최종 데이터베이스의 사본을 동일하게 보관하고 있어야 하며 이때 발생하는 싱크 작업 자체에도 상당한 시간이 소요됩니다.

② 블록 생성에 사용되는 비효율적 알고리즘

a 비트코인의 PoW 합의 알고리즘의 경우 한 블록을 생성하는데 소모하는 전기량은 2017 년 기준 약 250 kW h 이며 이는 서울시 한 가족이 사용하는 월 전기량과 동등한 수준의 양 입니다. 이러한 양의 전기를 단순히 단 1 개의 블록을 쌓는 데에 사용되고 있습니다.

b 비트코인의 채굴 풀(Mining Pool)은 한때 전 세계 500 대의 슈퍼컴퓨터가 사용하는 연산 능력을 모두 합친 것보다 훨씬 많은 양의 연산을 수행한 적이 있으며 한 해 전세계 하위 160 개국이 1 년간 사용하는 전기량을 앞질렀던 적도 있었습니다. 이러한 엄청난 자원의 소비가 단순한 블록 쌓기에 동원되고 있는 것 입니다.

c 이를 개선하기 위해 수많은 종류의 합의 알고리즘들이 등장했지만 비트코인, 그리고 이와 유사한 알고리즘을 가진 이더리움은 여전히 부동의 세계 1, 2 위의 유통량을 자랑하는 암호화폐이며 아직도 70% 이상의 채굴 풀들이 이들을 위해 가동되고 있습니다.

1.3 블록체인 기술의 현재

① 블록체인은 완결된 기술이 아니다

a 블록체인에 사용된 암호 알고리즘이 향후 양자컴퓨팅에 저항성을 가질 수 있는지 회의를 제기하는 사람들이 많습니다. 비트코인이 사용하는 SHA-256 해시 알고리즘은 향후 양자컴퓨팅의 의해 해독될 가능성이 높다는 연구결과들이 이미 존재 및 발표되어 있습니다.

b 위험한 상황은 검증에 수년이 걸리는 암호화 알고리즘을 개발자가 '보장'한다는 이유만으로 믿고 써야 하는 상황으로 대표적인 케이스는 IOTA 가 자체 개발한 해시 알고리즘인 SHA-3(Keccak) 과 Curl 을 조합하여 변형한 P-Curl 에 대한 논쟁으로 검증이 되지 않은 암호화 알고리즘으로 아직도 이에 대한 검증 논쟁이 많이 발생하고 있습니다.

c 또한, 이더리움에서 사용하는 가상 머신의 단순 코딩 오류로 사용자들의 자산이 더 이상 사용할 수 없게 된다거나 2016 년 BitGo 지갑의 취약성으로 비트피넥스(Bitfinex) 거래소가 12 만 ETH 를 도난당한 사례들은 앞으로도 얼마든지 일어날 수 있는 결함들입니다.

d 블록체인의 최대의 적은 해킹이 아니라 미흡한 기술과 미흡한 어플리케이션 구조에 있다는 점을 입니다.

② 블록체인은 만능의 데이터베이스가 아니다

a 블록체인의 태생적인 구조상의 한계와 데이터가 추가되고 연결되는 과정의 문제로 인해 블록체인을 사용하면 더 큰 문제가 발생되는 경우도 있습니다. 어떠한 경우에는 기존의 관계형 데이터베이스와 분산된 네트워크 스토리지에 의존하는 전통적인 데이터베이스 방식이 훨씬 더 효율적일 수 있습니다.

b 특히, 프라이빗 블록체인의 경우 오히려 분권화(Decentralization)의 장점이 모호한 부분이 많기 때문에 개발 및 활용에 있어 많은 고민이 필요합니다. '공유'와 '강력한 조작 및 변경 저항성'이 필요한 경우가 아니면 블록체인의 사용 보다는 기존의 데이터베이스를 사용하는 것이 훨씬 나을 수 있기 때문 입니다.

c 블록체인은 '체인' 형태이기 때문에 블록 삽입이 직렬화되어야 하고 이 때문에 데이터의 업데이트 속도가 병렬적인 업데이트를 하는 전통적인 데이터베이스보다 느릴 수 밖에 없습니다. 검증되지 않은 불특정 다수의 사람이 참가할 수 있는 글로벌 규모의 네트워크에서는 이처럼 비싼 비용과 느린 속도가 용납이 될 수 있지만 참가자가 엄격히 통제되는 기업 환경에서 는 블록체인 기술에 그토록 많은 에너지와 시간을 쏟을 필요는 없을 수 있기 때문 입니다.

③ 스마트 계약(Smart Contract) 과연 만능인가?

a 자율 및 자체 이행을 하는 계약의 측면에서 스마트 계약은 별도의 이행 강제 주체가 필요 없다는 점에서 블록체인이 가진 가장 매력적인 기능 중 하나라고 할 수 있습니다. 기본적으로 이는 계약 당사자들이 동의한 거래 조건이 달성될 때 그 대가로 약속된 자산이 자동으로 계약 당사자에게 전달되는 시스템이다.

b 개념적으로 보면 멋진 아이디어임에도 이런 비즈니스 프로세스의 자동화를 코드로 옮긴다는 것은 생각보다 훨씬 어렵고 복잡한데 비해 현재까지 개발된 블록체인 개발용 스크립트 언어는 매우 초보적인 수준으로 단순한 분기와 조건 설정이 전부인 상태 입니다. 더불어 이런 스크립트 자체의 오류도 상당하여 때로는 이더리움 하드포크 사태와 같은 최악의 상황까지 불러일으키기도 합니다.

c 거래 준수에 대한 세부적인 운용방안, 분쟁 시 해결 방법 등, 실질적인 블록체인의 운영 방법을 구현해 넣기에는 아직 스마트 계약은 스마트 하지 못하다. 특히 비정상적 상황에서 계약을 정지 시키는 '킬 스위치' 등은 블록체인은 변경될 수 없다는 대원칙에 위배되는 자기 모순을 가지고 있고 이러한 형태의 많은 문제들의 해결 방안을 추가하면서 대원칙에 위배되지 않는 블록체인을 개발하는 일은 아직도 스마트 계약이 넘어야 할 산 입니다.

1.4 블록체인 기술의 미래

① 확장성과 신뢰성의 확보

a 블록체인 기술은 기본적으로 두 개인, 기업, 단체 간의 신뢰를 수학적 원리로 대체하려는 시도다. 이는 바꿔 말해 블록체인 기술의 수학적 원리에 대한 의존성이 커질수록 더 많은 노드(서버)가 필요하게 되며, 운영 환경은 더욱 연산 집약적이 되고, 그에 따라 비용도 증가하게 됨을 뜻합니다. 이는 필연적으로 블록체인의 확장성을 더욱 악화시키는 악순환을 발생시킵니다.

b 또한, 가장 널리 사용되는 블록체인 형태이기도 한 퍼블릭 블록체인은 불특정 다수에게 투명하게 열려 있기 때문에 누구나 거래 장부를 볼 수 있습니다. 비트코인과 이더리움이 이런 경우인데 이것을 상업적 환경에서 사용할 때에는 이 투명성이 반드시 좋은 것이라고 보기 어렵습니다. 예를 들어, 만약 블록체인 기술이 주식 거래 플랫폼상에서 즉각 합의(Instant Settlement) 메커니즘으로 사용된다면? 각 참여자가 다른 참여자의 모든 의도와 행동을 읽을 수 있게 되므로 결국 메커니즘 자체가 제대로 기능하지 못하게 될 가능성이 있는 것 입니다.

c 또 다른 예로, 만일 제조업에서 공급업체에 대한 공개 거래 장부로 블록체인 기술을 사용할 경우 어떤 한 계약자가 블록체인상의 다른 모든 거래 업자들의 거래 내역을 볼 수 있게 될 것 입니다. 이 상태에서는 반대로 기업은 이 거래 데이터를 어떻게 하면 비공개로 유지할 수 있을지에 대한 고민이 생길 수 밖에 없습니다. 현재의 기술

내에서는 이런 경우에 기존의 전통적인 방식의 데이터베이스를 사용하는 것이 더 나은 선택이 될 수 있습니다.

d 블록체인을 사용해야 하는 경우는 비 신뢰 기반의 '거래', 참여자 모두에게 열린 '공유' 시스템 및 '강력한 조작 및 변경 저항성'이 필요한 경우뿐 입니다.

② 암호 화폐의 미래

a 미래학자이자 저자인 토마스 프레이(Thomas Frey)는 "암호 화폐는 생활의 일부가 됐다. 암호 화폐가 2030 년까지 법정 화폐의 25%를 대체할 것"이라고 예측하며 암호 화폐를 훨씬 더 효율적인 시스템으로 평가했다.

b 글로벌미래연구소의 제임스 캔턴(James Canton)박사는 "암호 화폐가 지난 2 년간 새로운 자산으로서의 가능성을 보여줬다"고 말했다. 그는 "암호 화폐 투자가 기하 급수적으로 증가할 것"이라고 덧붙였다.

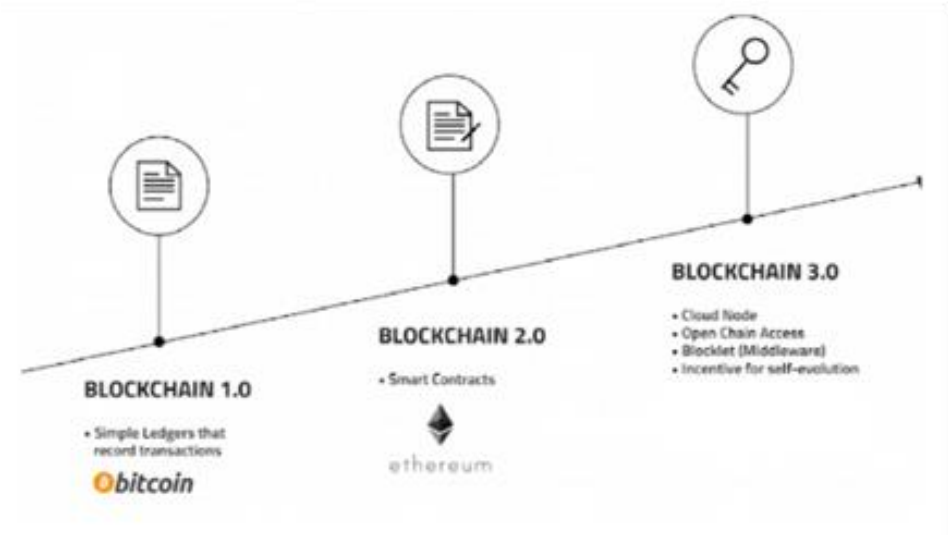
③ DAO 구조

<그림 2. Distributed Autonomous Organization; DAO 구조>

a 과거의 신뢰할 수 있는 제 3 자를 구성원들이 선출 또는 선택하여 역할을 부여하는 방식에서 구성원이 직접 모든 정책에 참여하고 의사결정을 하는 체계로 다양한 이해관계자들이 집단지성을 활용하여 의사결정을 할 수 있는 환경을 블록체인

네트워크가 제공하는 방식으로 변모하여 "분산된 자율조직(Distributed Autonomous Organization; DAO)"이 등장하고 모든 서비스를 블록체인 기반으로 운용하게 될 것입니다.

④ 블록체인 3.0



<그림 3. 블록체인 3.0>

a 블록체인 1.0, 블록체인 2.0 시대를 지나 초당 1,000 TPS 속도의 트랜잭션 처리와 양자 컴퓨팅 저항, 현실에 즉각 반영할 수 있는 기술인 블록체인 3.0 기술의 등장

2 DUBU4 블록체인 개요

우리는 DUBU4 는 블록체인의 발전과 밝은 미래를 믿습니다. 수학적 알고리즘이 보장하는 '비신뢰 기반(Trustless)' 거래 시스템이라는 개념은 인류가 생각할 수 있는 최고의 개념적 경제 혁명이라고 말할 수 있을 정도이며 때론 아름답기까지 합니다.

그러나 위에서 보았듯이 넘어야 할 요소가 분명히 존재합니다. 우리가 주목하는 부분은 '효율성'과 '사용성'의 향상에 있습니다. 실제 수요자가 현실에서 요구하는 다양한 거래 요구와 블록체인만이 가지는 고유한 특징을 쉽게 활용할 수 있게 만드는 것을 우리의 과제로 생각하고 있습니다.

우리의 DUBU4 Blockchain and Economic System 이 제공하고자 하는 서비스를 개념과 기술 두가지 측면에서 좀 더 상세히 설명하고자 합니다.

2.1 DUBU4 블록체인이 그리는 미래

① 전자 ID (Digital Identity)

a 위, 변조에서 자유롭고 본인인증 절차의 복잡한 등을 간소화하는 디지털 신원 시스템. 사용자는 본인 정보에 대한 모든 권한을 가지며 정보 활용도를 직접 결정합니다. 분산장부기술(Distributed Ledger Technology)로 위, 변조 및 도용 위험을 최소화 할 수 있으며 발급 절차의 간소화로 비용절감 및 정보 노출을 최소화 할 수 있습니다.

b 다양한 분야에서 개인의 정보를 사용해야 할 때 정보 제공자 자신이 허락한 노출 정도에 따라 쉽고 빠르게 데이터를 찾을 수 있으므로 관공서 및 특정 기업을 통해야만 하는 신원 확인 프로세스의 단계를 대폭 줄일 수 있습니다.

② 전자문서 원본 증명 (Digital Stamping)

a 블록체인이 제공하는 거래와 시간 정보의 합치성의 부분을 응용하면 발급된 전자문서의 해시와 거래 해시를 동시에 게재하여 제출된 전자문서의 원본 보증 및 검증이 손쉽게 이루어질 수 있습니다.

b 위탁기관, 인증기관, 등록기관 등으로 분류된 기존의 제 3 자 인증 위임 프로세스의 비효율성, 고비용 구조가 한 번에 개선되며 도용 방지 및 진위 여부 구분에 탁월한 보안 능력을 제공할 수 있습니다.

③ 출처 증명 (Provenance)

a 엔터프라이즈 유통채널의 경우 관리자 검증과 승인 기록을 DUBU4 블록체인으로 관리하여 전 과정을 한 번에 모니터링 할 수 있으며 관계자 사이의 인증 및 승인 절차를 간소화 할 수 있습니다.

b 이는 식품의 변질이나 가축의 질병 등으로 인해 문제가 발생했을 경우 역학조사 및 원산지, 제조자 출처를 즉각적으로 확인하여 효율적인 대응이 가능하도록 할 수 있으며 전체 단계를 담고있는 유통 정보로 인해 이 거래에 참여하는 다양한 사용자에게 무결성이 보장된 정보가 공유됨으로 번거로운 2 중, 3 중의 확인 과정없이 전체 프로세스를 빠르게 진행할 수 있습니다.

c 소비자에게는 출처 및 원산지 확인에 대한 불 투명성을 제거하여 단순히 제조 및 공급자의 약속을 믿는 것이 아니라 직접 빠르고 간편하게 원산지 및 유통 정보를 확인 할 수 있는 것도 가능합니다.

④ 물류 (Logistics)

a 물류는 하나의 거대한 순환 망으로 이는 제조에서 판매, 수요자, 중간에 공공기관까지 참여하는 거대한 블록체인으로 간주할 수 있습니다. 현재의 방식에서는 각 단계에 필요한 검증 및 확인 절차로 인해 실제 상품이나 서비스에 드는 이동 시간을 제외하고도 엄청난 비용이 소비됩니다.

b 이러한 문제점들은 관리 및 유통 비용을 상승시키고 결과적으로 전체적인 유통 비용을 상승시키는 결과를 가져옵니다. 특히 국제간 거래가 개입된 물류인 경우는 이 복잡도는 기하급수적으로 증가하게 됩니다.

c 만일, 여기에 DUBU4 의 효율적인 블록체인 시스템을 도입하한다면? 만일, 투명하고 위, 변조가 불가능한 블록체인에 이 물류의 전 과정에 걸친 내용들을 기록해 놓는다면 물류 망의 참여자 간 의사소통이 획기적으로 향상될 수 있으며 공공기관에서도 효율적인 모니터링을 실시하여 관세, 세금, 검역 등의 Compliance Process 들이 투명하고 빠르게 진행될 수 있을 것 입니다. 생산, 소비자들뿐만 아니라 물류 과정 전반에 걸치 참여자들에게 스마트한 물류 관리가 제공될 수 있는 기반이 되는 것 입니다.

d 사용자 입장에서도 기관이나 업자가 제공해주는 기존의 물류 정보 조회의 한계에서 벗어나 전 과정에 걸치 세세한 과정을 쉽게 모니터링 할 수 있어 구매자 단계의 오류를 크게 개선시킬 수 있습니다.

e 바야흐로 DUBU4 블록체인이 물류의 혁명적 변화의 토대가 되는 것 입니다.

⑤ Full Stack Governance 를 통한 균일하며, 자율적이고 효과적인 자산 거래 시스템 구현

a 디지털 자산의 효율적인 거래를 위해 다양한 시스템을 준비하고 제공합니다. 우리는 DUBU4 블록체인이 외부에 존재하는 현실 및 디지털 자산과 DUBU4 블록체인 안에 존재하는 자산의 원활한 교환을 위해 여러가지 기술적 단계를 제시할 계획입니다.

b DUBU4 는 사용자가 원할 경우 자산을 손쉽게 생성할 수 있습니다. 해당 자산은 유저가 발행한 Token 일 수도 있고 별도의 자산과 연결시킨 Bridge 자산일수도 있습니다. UCA (User Created Assets)을 풀 스케일로 지원하여 특별한 목적의 자산 거래 시스템을 쉽게 구현할 수 있도록 지원합니다. 주요한 목적을 위한 DUBU4 자체의 Core Assets 을

제공하며 이를 기반으로 유저들이 스스로 User Assets 을 확장시켜 나갈 수 있도록 합니다

c 우선 DUBU4 블록체인 외부의 디지털 자산 (비트코인, 이더리움 등)을 연결시키는 다양한 Gateway 및 Bridge 시스템을 제공할 예정입니다. 이로써 DUBU4 Coin Type-1 을 교환 매개로 하는 DEX 및 Atomic Swap 이 가능하도록 합니다. Fiat 화폐 (현물 화폐) 및 자원 등과 같은 현실 세계에 존재하는 유형의 자산들은 믿을 수 있는 제 3 자에게 전용 블록체인 노드를 제공하여 블록체인 내부의 자산과 연결시킬 수 있도록 합니다.

d 이는 Paypal 혹은 블록체인계에서는 Ripple 이나 Stellar 의 Anchors 와 비슷한 개념이며 신뢰할 수 있는 제 3 사업자의 개입으로 진행이 될 수 있습니다. 물론 Anchors 들 역시 모든 거래는 DUBU4 의 블록체인과 동일한 프로토콜 아래 진행됩니다. 이를 응용하면 이미 존재하는 현실 세계의 다양한 결제 시스템들(ATM, Credit Card 등)과 쉽게 접목이 가능합니다.

⑥ DUBU4 ICO 및 토큰 발행 시스템을 통한 다양한 거래 방법 지원

a 블록체인은 이제 DApp 을 제외하고는 논의할 수 없는 시대가 되었습니다. 사용자는 제공되는 '기본 시스템'만으로 생성된 거래보다 자신에게 맞는 새로운 거래 방법을 만들고 싶어하고 독특한 아이디어를 구현하고 싶어합니다. DUBU4 는 사용자가 자신의 기준 토큰 및 추가적인 새로운 토큰 발행, ICO 진행 시 외부 토큰과의 연결 등을 지원하여 자생적으로 발전하는 블록체인 생태계를 적극적으로 지원 할 예정입니다.

b 사용자 및 기업은 자유롭게 자산형, 지분형, 상품형 등 다양한 방향으로 스스로의 거래 아이디어를 보증된 가치 체계 아래서 설계해 나갈 수 있습니다.

c DUBU4 경제 생태계의 분산화 된 서비스 (Decentralization of Services) 토대 제공

d 유틸리티 토큰과 블록체인의 레버리지를 활용한 다양한 서비스 접목을 지원합니다.

e 통화정책과 무관한 유틸리티 토큰을 사용하므로 규제에서 비교적 자유로우며 서비스의 단위가 명확하여 사용 개념의 이해가 쉽고 사용자에게는 위, 변조가 불가능한 장부로 인해 높은 투명성을 담보할 수 있어 이를 접목하여 기존 구조를 개선하기 좋은 분야를 선제적으로 지원하고 그에 따른 기반 기술을 개발해 제공할 예정입니다.

2.2 DUBU4 블록체인 코인 레이어

① 코인 레이어 개요

a 총 4 종류의 코인으로 구성

i Coin Type-1

ii Coin Type-2

iii Coin Type-3

iv Coin Type-4

b 향후 3 년간 4 개의 코인을 개발 및 공개할 예정입니다. 곧 정식 명칭을 부여할 예정입니다.

② 코인 레이어 상세

a 현재 DUBU4 Blockchain 1 이 먼저 개발되어 시험 가동되며 이 블록체인 위에서 Coin Type-1 이 생성될 예정입니다. DUBU4 Blockchain 1 은 우리가 생각하는 블록체인의 근간이 되는 부분을 담고 이를 확장할 수 있도록 설계되어 각 기능별 정합성 및 효율성을 실제 확인하는 실험적 블록체인의 성격을 가지고 있습니다. Coin Type-1 은 총 10 억개 발행 예정이며 모두 선 발행되고 40% 가량의 외부 공개 물량을 제외하고는 모두 재단이 보유할 예정입니다.

b DUBU4 Blockchain 2 는 우리의 본격적인 블록체인으로 DUBU4 Blockchain 1 에서 검증된 내용 및 ICO 참여자들의 다양한 의견을 바탕으로 대대적인 보완 및 개선이 이루어져 공개될 예정이며 향후 Coin Type-2, 3, 4 를 위한 기반 블록체인의 역할을 할 것 입니다. 실제 릴리즈되는 DUBU4 블록체인이라고 할 수 있습니다.

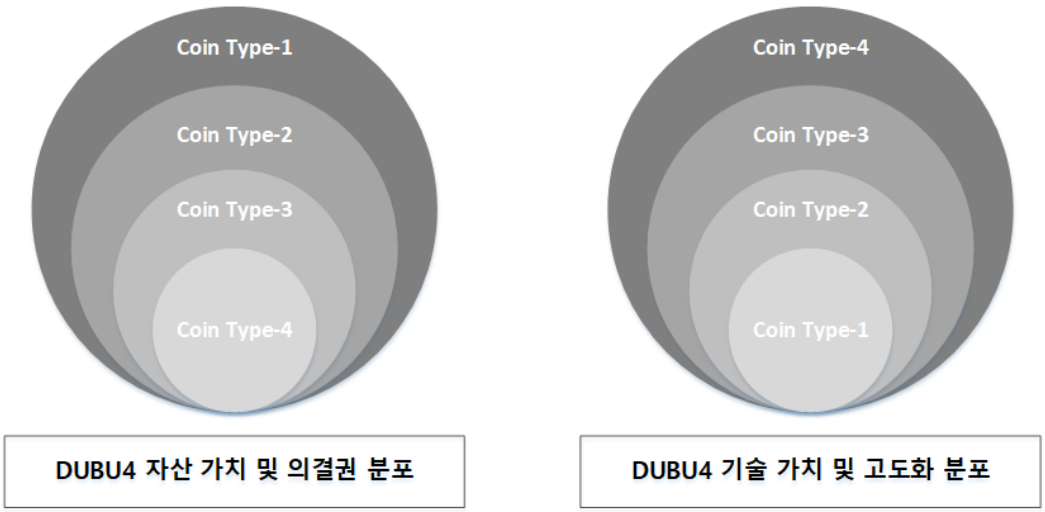
c DUBU4 Blockchain 1 은 지속적으로 존재하며 이에 따른 Coin Type-1 도 향후 추가 개발되는 블록체인 내의 코인들과 외부 코인들을 연결하는 Bridge 코인으로 그 역할을 할 예정입니다. 저희 DUBU4 에서 Coin Type-1 은 이러한 매우 중요한 의미를 지닙니다. 최초 프라이빗 세일 투자 진행자들 및 ICO 참여자들을 중요한 참여자로 모시겠다는 약속이기도 하며 실제적으로 추후 개발되는 모든 DUBU4 코인의 가치를 평가하는 기준 코인의 역할을 하게 될 예정입니다.

d 예를 들면, DUBU4 Blockchain 2 에 적용될 Witness 노드 운영은 Coin Type-1 을 Deposit 해야 가능하도록 설계할 예정이며 DUBU4 Blockchain 의 발전 방향에 대해

중요한 합의가 필요할 경우 투표권을 부여합니다. 저희가 제공할 예정인 DEX 에서도 이미 가치를 보유한 코인으로써 Atomic Swap 에 Base Resource 로 교환하거나 매각할 예정입니다. DUBU4 의 모든 주주권은 Coin Type-1 에서 나온다는 것이 저희가 Coin Type-1 의 소유자 분들에게 드리는 약속입니다.

e Coin Type-2 는 PoS/PoW Hybrid 로 설계될 예정이며 30% 선 발행(Pre-Mined), 70%는 채굴 유동(Mining Liquid)으로 남겨집니다. 선 발행된 코인 30%는 전량 Coin Type-1 의 가치로부터 파생되며 Coin Type-1 과 교환을 원하면 소각 후 해당 가치만큼 CoinType-2 코인을 지급합니다. Coin Type-2 의 자세한 기술적 스펙은 추후 백서 버전업과 Git-Hub 소스 공개를 통해 자세히 공개 할 예정입니다.

f Coin Type-3, 4 는 현재 PEG Coin 과 Bridge-Anchor 전용 코인으로 설계 및 개발됩니다. 이 부분에 대한 자세한 기술적 스펙은 추후 로드맵에 따라 순차적으로 공개될 예정입니다.



<그림 4. 4 Types Coin 가치 내재>

2.3 DUBU4 블록체인 지향점

① DAG 기반의 블록체인 코어 – 현존하는 가장 빠르고 경량화된 블록체인 시스템

a No Blocks with Unlimited Transactions

b Conditional Payment – 조건부 지불, 일반인도 이용 가능한 간이 스마트계약

c Witness Consensus Algorithm – 이중지출(Double Spending) 방지

d Quantum Resistance – 양자컴퓨팅 저항성

e D4VM – DUBU4 Virtual Machine, 다양한 언어의 가상머신 지원

② UAM - Unified Assets Manage System, 통합 자산관리 시스템 지원

a 송금, 인출

b P2P 거래소

c Trusted Data Oracle, 제 3 자 데이터 피딩 시스템

d Dual Transaction, 신뢰도 조절 기반의 Dark, White 거래 시스템

e Messenger Based 자산 관리, Textcoin Messaging

f Blockchain Schema & Deploy GUI Tool

g Transaction Search

③ DEX – 철저히 분권화된 거래소 시스템

2.4 DUBU4 블록체인 기술 핵심

DUBU4 블록체인 시스템의 빠른 트랜젝션과 모듈화 시스템 구성으로 메인 체인과 기타 파생 체인들 간의 연결성(Linkable)을 지향하고 있습니다. 메인 체인을 구성하는 근본 알고리즘은 DAG 성격의 알고리즘을 사용합니다. DUBU4 에 구현된 DAG 에 대해 좀 더 자세히 알아보겠습니다.

① DAG 란 무엇인가?

a DAG 는 간단하게 말하면 위상 정렬(Topological Ordering)을 사용하는 방향성 그래프 데이터 구조를 말합니다. 각 요소들은(Entity)의 순서는 일정한 방향으로 진행되고 역방향이 없도록 정렬되어 순환구조가 만들어지지 않습니다. 주로 데이터 처리, 스케줄링, 최적 경로 찾기나 데이터 압축 등에 활용되는 데이터 구조입니다.

b 비트코인은 PoW(Proof-of-Work) 시스템의 한계로 인해 효율성이 떨어지는 데이터 구조를 가지고 있습니다. 블록들은 동시에 생성될 수 없으며 전체 네트워크에서 유효한 연결 데이터 구조는 오직 하나만 존재합니다. 해당 노드 주변에 존재하는 비슷한 시간대에 발생한 모든 거래는 동일한 블록에 기록되고 채굴자(Miner)에 의해 블록의

정합성을 획득한 블록만이 포함된 거래들을 가지고 정식 블록으로 채택됩니다. 작업 증명(PoW)에 성공한 블록에 포함되지 못한 거래들은 'Unconfirmed' 상태로 전환되고 다시 다음 블록 생성으로 넘겨집니다. 이 모든 작업이 10 분 단위로 이루어집니다. 채택되지 못한 거래는 몇시간에서 몇일을 기다려야 할 수도 있습니다.

c 이런 비효율적인 구조를 개선하고 체인 형태의 데이터 구조를 DAG 형태로 재구성하고자 하는 시도는 NXT 에 의해 최초로 시도됩니다. 채굴 시간을 바꾸지 않고 동시에 네트워크상의 블록을 몇배로 확장할 수 있는 아이디어였습니다. 그럼에도 이 때까진 사이드 체인 (Side-chains)의 성격이었으며 아직 블록이라는 개념은 그대로 차용하고 있었습니다. 단지 다른 형태의 거래를 다른 체인에서 동시에 처리한다라는 접근 방법이었고 실제로 이 사이드 체인은 기존 블록체인 1.0, 2.0 기술들을 보완하는 기술로 현재 활용되고 있기도 합니다.

d 그러나 여전히 블록 생성 시간이라는 병목 현상은 해결되지 않고 있습니다. 비트코인은 매 10 분, 이더리움은 좀 더 나아졌지만 15-20 초가 필요합니다. 이들 블록체인 시스템은 많은 거래들을 블록 안으로 집어넣고 거래 순서들은 블록 간 해시들에 의해 연결, 유지되게끔 합니다. 그러나 '왜 블록을 만들어야 할까?' 라는 근본적인 의문에서 출발해서 해결책을 찾기 시작했습니다. 만일 '거래와 블록을 하나로 처리하면 어떨까?', '모든 거래들이 그 자체가 단일 블록으로서 전체 시퀀스를 유지하는데 직접적으로 사용하면 안되는 것인가?', '거래가 인증되면 채굴이라는 과정을 즉시 생략할 수 있으며 블록 자체가 필요 없는 단일 거래 베이스의 더욱 효율적인 시스템이 될 수 있지 않을까?'의 이 모든 의문에서 출발하여 탄생한 것이 DAG 입니다.

e 이것이 바로 블록이 없는 (Blockless) 블록체인 시스템, Blockless-DAG 입니다.

② DAG 의 기술적 이슈들

a 이중 지불(Double-Spending)이라는 근본적인 문제의 해결

i DAG 에서는 Hash 함수를 해결한 Miner 가 동시에 하나 이상일 수 있습니다. DAG 에서는 새로운 거래가 이전의 (최소)두개의 거래를 검증하는 시스템으로 되어 있습니다. 이때 서로 연결된 거래의 개수(혹은 '투표-Vote'라고도 합니다)로 동시에 발생한 거래(Transaction)의 정합성(Validation)을 검증합니다. 더 많은 거래와 연결된 거래가 채택이 되는 구조로 이중 지불 문제를 해결합니다.

b 네트워크 대역

i 각 거래가 검증이 될 때 기존 네트워크에 포함된 이전 거래들과 연결이 되어야 하는데 이때 훨씬 이전 거래 노드와 마구 링크가 된다면 검증해야 할 트랜잭션의 경로가 길어져 네트워크 자체가 너무 방대해 지는 문제가 생깁니다. 그래서 DAG 네트워크는 새로운 거래가 생길 때 검증을 위해 연결될 이전 노드를 비교적 최근에 발생한 노드와 연결시키려는 알고리즘을 가지고 있습니다. 이것의 목적은 네트워크를 특정 범위 안에 유지시키고 새로 발생한 거래를 빠르게 검증하기 위해서 입니다.

c 빠른 거래

i DAG 는 블록을 생성하지 않는 구조적 특성상 거래 자체가 네트워크에 바로 삽입되며 전체적으로 PoW, PoS 기반의 블록체인들보다 훨씬 빠르게 거래가 이루어집니다.

d 불필요한 채굴

i DAG 에는 Miner 가 없습니다. 거래 검증은 거래 자체에서 발생합니다. 사용자들에게 이것은 거의 즉각적인 거래 성사를 의미하기도 합니다.

e 마이크로 페이먼트(Micro-Payment)

i 소규모 즉시 결제에 유리합니다. DAG 의 구조적 특성상, 고기능의 함수를 최소의 수수료로 구동시킬 수 있습니다. 적은 비용을 결제할 때 수수료가 더 많이 나오는 비트코인이나 이더리움보다는 훨씬 유연하고 빠르며 합리적인 결제 시스템을 구축할 수 있으므로 실제로 사용할 수 있는 분야가 극적으로 확대될 수 있습니다.

f 어플리케이션 Scalability

i 초당 수천 건의 트랜잭션이 필요한 어플리케이션에도 DAG 는 유용하게 사용될 수 있습니다. CryptoKitties 라는 이더리움의 DApp 은 그 엄청난 인기에 힘입어 결과적으로 이더리움 전체 네트워크를 더 느리고 비싼 수수료의 네트워크로 만들어 버렸습니다. 이더리움에서는 Sharding 이라는 해결책을 제시했지만 벌써 5 년이 지났고 근본적인 해결이 되진 못했습니다. DApp 을 개발하는 데에도 DAG 는 새로운 대안이 될 수 있습니다.

3 DUBU4 블록체인 Technical Detail

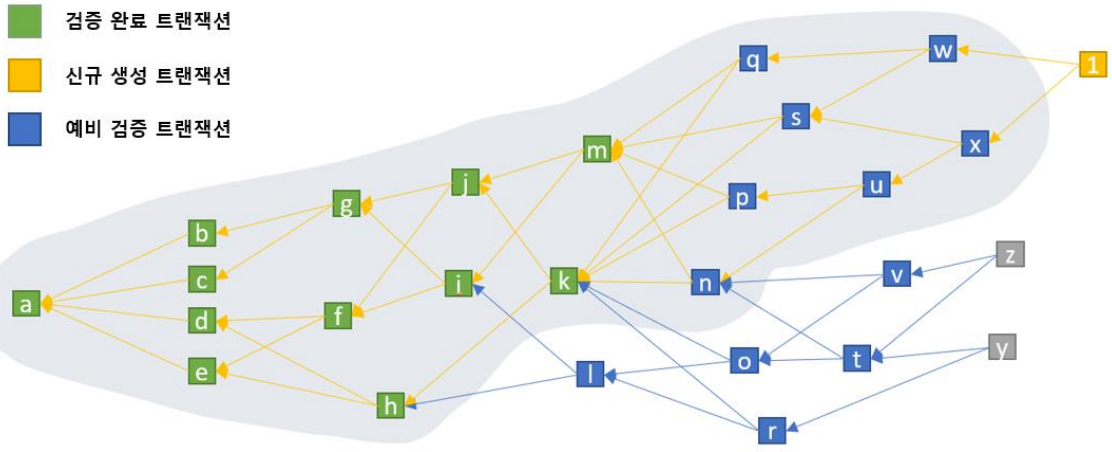
3.1 DUBU4 코어

① 유연성, 신속성, 통합성

a 현 시점에서 블록체인은 그 시작점에서의 기술 가능성에서 크게 벗어나지 못한 상태로 오래된 블록체인 플랫폼의 데이터 사이즈는 거대해지기만 해 단 하나의 트랜잭션을 처리하는데 드는 시간과 비용도 날로 증가하고 있습니다. 이러한 사유로 인해 현실의 비지니스 분야들과 지금까지의 블록체인 시스템이 결합되어 실질적인 비즈니스에 어떤 영향력을 끼치는 것이 어려웠던 것이 사실입니다. 여기에 DUBU4 블록체인의 개발 시작점이 있습니다. DUBU4 블록체인 코어는 단순히 블록체인 기술 위에 코인만 올려진 현 상태의 블록체인에서 벗어나 누구나 블록체인 기술을 기존 환경 및 비지니스와 결합하여 사용할 수 있도록 DAG 기술 기반으로 9.7K TPS 의 신속성으로 트랜잭션의 처리를 완료하고 그 데이터의 무결성을 보장합니다.

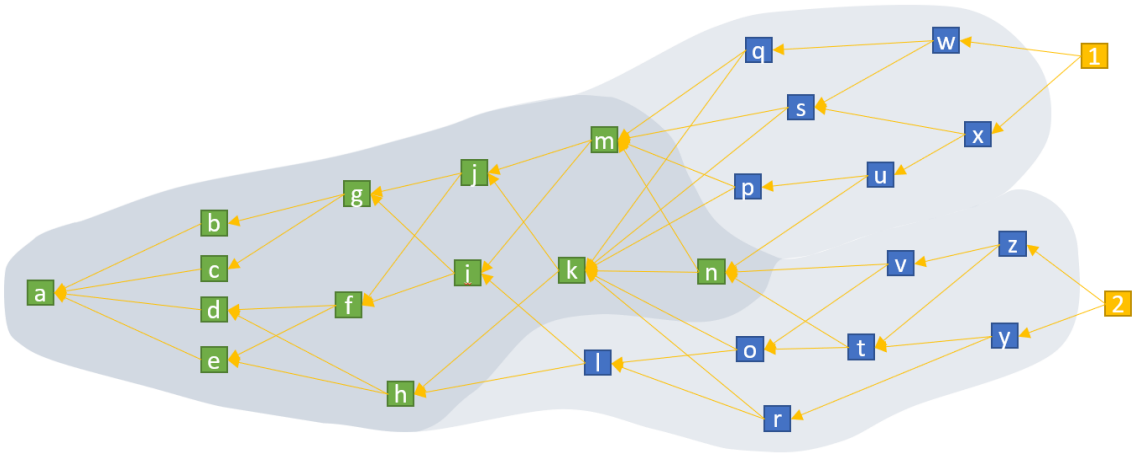
b DUBU4 블록체인은 설계 시작부터 사용자가 독립적인 자신의 네트워크를 만들기 쉽도록 설계되었고 이는 GUI 형태로 제공되는 개발 툴을 사용하여 누구나 자신이 원하는 기술적 내용을 포함한 코인과 네트워크를 구성하고 가질 수 있습니다.

c 또한, 이렇게 탄생된 모든 네트워크들은 독자적으로 변형 및 발전시킬 수 있으며 이들 모두가 DUBU4 메인 네트워크와 손쉽게 결합되어 DUBU4 엔터프라이즈가 완성될 수 있습니다.



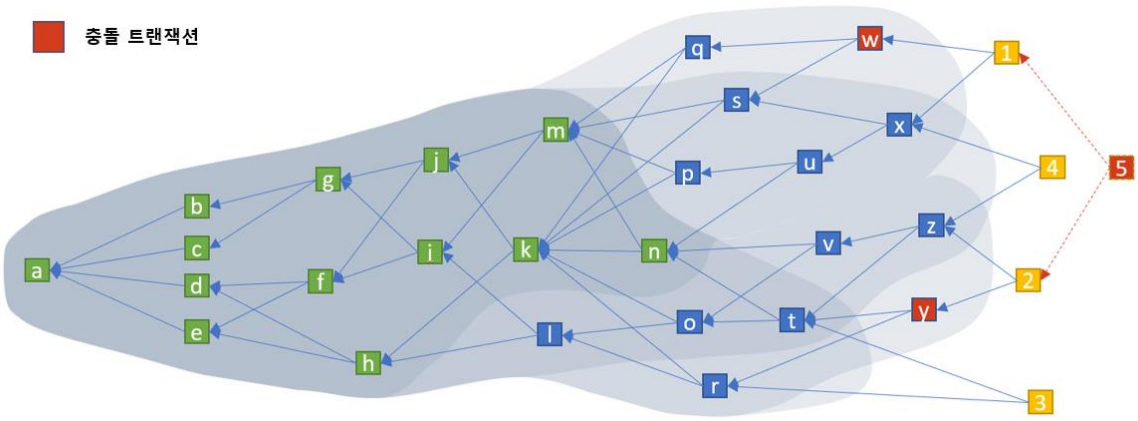
새 트랜잭션 1 을 추가하려면 w 과 x 두 개의 트랜잭션을 무작위로 선택하고, w 과 x 를 검증해야 합니다. 검증 작업과 함께 트랜잭션이 직간접적으로 참조하는 과거의 트랜잭션과 충돌이 없는지 확인을 거칩니다. 선택한 트랜잭션에 문제가 없다면 사용자는 그 두 개의 트랜잭션 w 와 x 를 참조하는 새 트랜잭션 1 을 전체 체인에 추가합니다.
선택된 두 개의 트랜잭션 w 와 x 에 의해 직간접적으로 참조되지 않는 트랜잭션(회색 경계 밖에 있는 l, o, r, t, v, y, z)은 1 이 트랜잭션에 추가되는 과정에서는 검증되지 않습니다. 이 트랜잭션들은 나중에 다른 트랜잭션이 추가될 때 검증됩니다.

<그림 5. DUBU4 DAG 트랜잭션 처리 과정>



1 과 2 에 의한 검증 경로를 겹쳐보면 위의 그림과 같습니다. 어떤 트랜잭션은 1 이나 2 중 한 쪽에 의해서만 검증되고, 어떤 트랜잭션은 1, 2 모두에 의해 검증됩니다. 현재 시점에 존재하는 트랜잭션 모두에 의해 검증되고 완료된 트랜잭션들은 '검증 완료'되었다고 합니다. 그래서 n 은 새롭게 전체 완료되어 DUBU4 의 더 깊숙한 곳으로 들어가고 검증 완료(초록색 네모)로 바뀌게 됩니다. 추가로 1 또는 2 에 추가되는 자손 트랜잭션들은 n 을 계속해서 재 검증하게 됩니다.

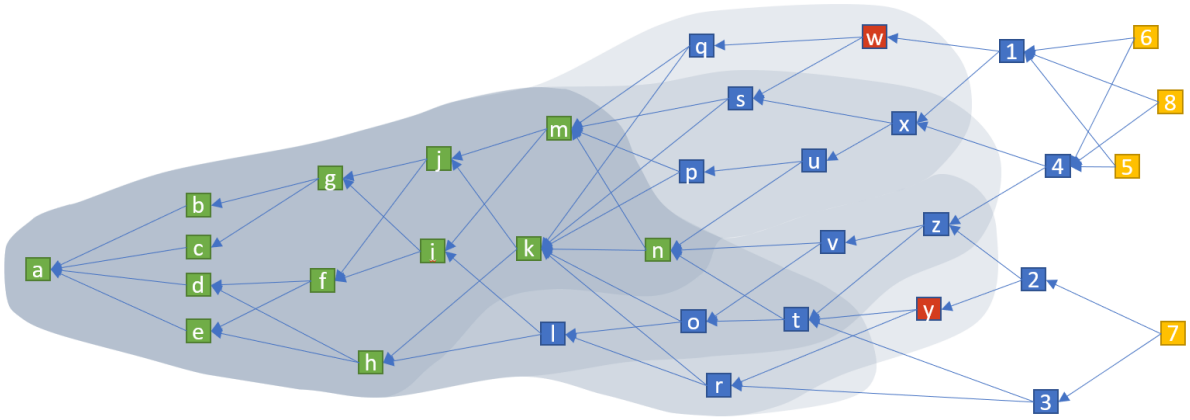
<그림 6. DUBU4 DAG 트랜잭션 검증 완료 과정>



사용자가 두 개의 충돌되는 트랜잭션 w 와 y 를 DUBU4 블록체인의 서로 다른 영역에서 발생시켰다고 가정합니다. 이후에 추가되는 트랜잭션은 검증 트랜잭션의 선정이나 전파 지연 때문에, 충돌되는 트랜잭션인 w 와 y 중 하나의 트랜잭션만 검증 경로에 포함 시킬 가능성이 있는 상태 입니다.
예를 들어 1 을 추가하는 사용자와 2 를 추가하는 사용자는 w 와 y 가 충돌된다는 사실을 알 수 없으며, 결과적으로 w 와 y 를 충돌 없는 유효한 트랜잭션으로 판별하게 됩니다.

하지만 그 충돌은 머지 않아 발견됩니다. 1 과 2 를 참조하는 5 가 추가되면 5 의 검증 경로에는 w 와 y 가 모두 포함되므로 5 는 충돌을 발견할 수 있습니다. 그래서 5 는 1 과 2 를 선택하지 않고 충돌이 없는 다른 두 개의 트랜잭션을 다시 선택할 겁니다. 그래야 5 자신이 나중에 추가되는 트랜잭션에 의해 유효한 트랜잭션으로서 검증 받을 수 있기 때문입니다.
검증 프로세스에서 충돌이 명백하게 발견되기 전에 w 와 y 중 하나만 검증 경로에 포함한 많은 사용자는 w 와 y 의 충돌을 발견할 수 없으므로, w 와 y 를 유효한 트랜잭션으로 인정할 가능성은 있습니다.
하지만 1 차적으로는 결국에는 사용자들이 새로운 트랜잭션을 w 를 검증 경로에 포함하는 트랜잭션과 y 를 검증 경로에 포함하는 트랜잭션 중 어느 쪽에 더 많이 추가 했는지에 따라 w 와 y 둘 중 하나만 확정되고 나머지 하나는 버려집니다. 버려진 쪽에 추가된 트랜잭션들은 충돌이 있었는지 몰랐지만 억울하게도 함께 버려집니다. 하지만 충돌을 모른 채 버려진 트랜잭션들이 DUBU4 블록체인에서 아예 사라지는 것은 아니고 다른 사용자에 의해 선택 되어 다시 추가되고 검증 받을 수 있는 기회를 항상 얻게 됩니다.
검증을 받으려면 이 프로세스가 다시 실행되겠지만 송금자로부터 거래 승인 정보를 다시 받아와야 할 필요는 없습니다.

<그림 7. DUBU4 DAG 트랜잭션 이중 지불>



앞의 이중 지불 문제에서 사용자는 5 를 1 과 2 에 추가하려고 했지만, w 와 y 가 충돌된다는 걸 발견하고는 다른 트랜잭션을 다시 선택해서 1 과 4 를 선택 했고, 1 과 4 에서는 충돌이 발견되지 않았으므로 5 는 1 과 4 에 의해 DUBU4 블록체인에 추가 되었습니다. 다른 사용자는(반드시 다른 사용자일 필요는 없습니다) 7 을 2 와 3 에 추가 했습니다.
이렇게 되면 w 를 포함하는 경로와 y 를 포함하는 두 가지 경로로 일종의 분기가 발생하지만, 앞의 이중 지불 단원에서 설명한대로 둘 중 하나는 버려지고 하나만 살아남게 됩니다. 트랜잭션의 누적 가중치를 감안한 무작위 선택 로직에 의해, 분기된 두 경로 중 한 쪽 경로에 더 많은 자손 트랜잭션이 추가될 것 입니다.
그리고 시간이 지나면 누적 가중치를 감안한 선택 알고리즘에 의해 한 쪽 경로에는 정상적인 방법으로 트랜잭션을 추가하는 것이 불가능해집니다. 앞의 그림에서 5, 6, 8 다음에는 새로운 트랜잭션이 계속 추가될 수 있지만, 7 다음에는 트랜잭션을 추가할 수 없게 됩니다. 그래서 y, 2, 3, 7 은 더 이상 검증을 받지 못하게 되고 검증 완료 상태가 될 수 없습니다.
이중 지불 문제에서 설명한 것처럼 버려지는 경로에 있던 y, 2, 3, 7 은 일단 DUBU4 블록체인 네트워크에서 떨어져나간 후에 다른 새로운 트랜잭션들에 의해 검증되면 다시 체인에 추가될 수 있습니다. y, 2, 3, 7 각각이 유효한 트랜잭션이라면 다른 정상적인 트랜잭션과 마찬가지로 결국에는 확정될 수 있습니다. 그래서 2, 3, 7 은 확정될 수 있지만 충돌 내용이 포함된 y 는 끝내 확정될 수 없습니다.

<그림 8. DUBU4 DAG 트랜잭션 이중 지불 문제 해결>

3.2 투명성과 익명성

① DUBU4 블록체인은 투명성과 익명성, 양쪽의 가치를 모두 지원합니다.

a DUBU4-Public

i DUBU4 퍼블릭 체인은 DAG 기반의 최소 노드 교차 검증 방식으로 체인에 연결된 노드들 중 새로운 트랜잭션과 교차되는 노드들 중 선택된 최소 단위의 노드들이 빠른 검증을 통해 트랜잭션을 완료하고 이 데이터는 다른 교집합의 노드들에 의해 확인됩니다. 이러한 방식을 통해 기존 POW, POS 기반의 블록체인에 비해 선택된 노드 사이에서는 100% 합의 방식으로 데이터의 투명성이 보장되면서 속도가 저하되지 않습니다.

b DUBU4-Private

i DUBU4 프라이빗 체인의 경우 어떤 공익적인 목적이나 모두에게 투명하고 열린 데이터가 우선시 되게 보다는 빠르고 정확한 서비스를 제공해야하는 실질 비즈니스와의 결합에 초점을 맞추고 설계되었습니다.

ii 프라이빗 체인에서는 기본적으로는 체인 관리자 홀로 관리할 수 있는 인증 서버를 통해 모든 트랜잭션을 관리, 검증하고 해당 트랜잭션을 완료할 수 있으며 이를 통해 기존의 방법들과 비교, 비즈니스 로직을 수행하는데 있어서 속도가 전혀 뒤쳐지지 않습니다.

iii 또한, 완료된 트랜잭션의 데이터는 프라이빗 체인 안에 있는 노드 구성원들에게 열람하게 할 수 있으며, 만약 필요한 경우 인증 서버 홀로 트랜잭션을 처리하지 않고 인증 서버가 검증한 트랜잭션을 현재 프라이빗 체인 내에 선출된 검증 위원 노드 혹은 매번 임의의 최소 노드가 선택되어 추가 검증하는 방법 등을 관리자가 간단하게 추가, 변경 가능합니다.

3.3 DUBU4 블록체인 – 합의

① Chain Scheme

a 우리의 DAG 는 특별한 DAG 입니다. 정상적으로 사용하는 경우 대부분 사람들은 새로운 단위를 DAG 가 한 방향으로 만 성장한다는 것을 의미합니다. 하나, 내부에 많은

비월선이 있는 두꺼운 코드로 그려 낼 수 있습니다. 이 속성, 우리는 자녀-부모 링크를 따라 하나의 사슬을 선택할 수 있다고 제안합니다. DAG 한 다음 모든 유닛들이 체인과 연결하십시오. 모든 부대는 우리가 주 체인이라고 부르는 이 사슬은 상대적으로 그래프의 가장자리를 따라 홉 수가 적다. 그것은 고속도로와 같습니다. 연결 도로, 주 체인을 만드는 한 가지 방법은 알고리즘을 개발하는 것입니다. 단위의 부모는 그 중 하나를 "최상의 부모"로 선택합니다. 선택 알고리즘 해당 단위가 사용할 수 있는 지식, 즉 데이터에 기반해야 합니다. 유닛 자체와 모든 조상에 포함되어 있습니다. 어떤 팁에서 시작 (아이가 없는 DAG 의 단위 (단위)로 설정하면 최상의 상위 링크를 따라 기록에서 뒤로 이동합니다. 이런 식으로 여행하면서, 우리는 주 체인을 만들고 결국 기원 부대에 도착합니다. 특정 유닛에서 시작하여 빌드 된 주 체인은 절대로 변경되지 않습니다. 새로운 유닛이 추가됩니다. 이것은 각 단계에서 우리가 아이부터 부모 및 기존 단위는 새로운 부모를 얻을 수 없습니다. 다른 팁에서 시작하면 또 다른 주 체인을 구축 할 것입니다. 여기서 주목할 것은 그 두 개의 주 체인이 역사상으로 돌아갈 때 교차한다면, 그들은 둘 다 교차점 이후에 같은 경로를 따라 간다. 최악의 경우, 주 사슬은 기원에서만 교차합니다. 단위의 과정 생산은 사용자간에 조정되지 않지만, 팁에서 너무 멀지 않게 수렴하는 주류 체인 입니다.

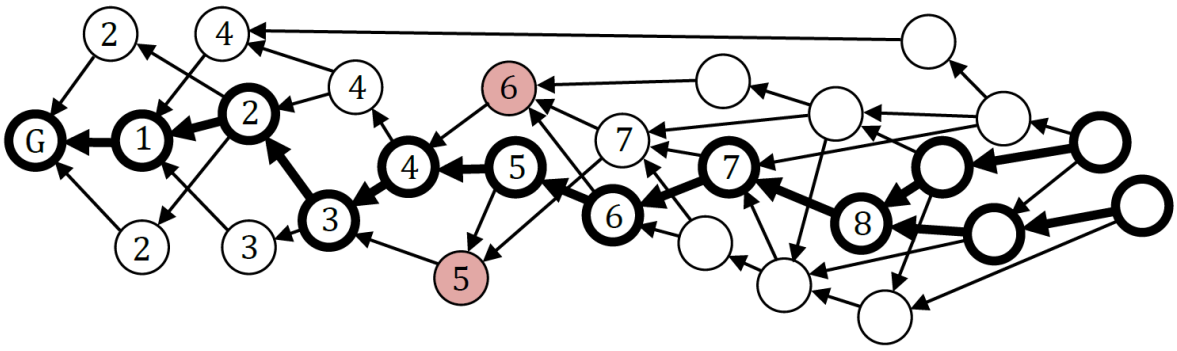


그림 n. 서로 다른 아이가 없는 단위로 만들어진 주요 체인이 교차하고 나서 이동합니다. 동일한 경로를 따라 2 배의 이중 지출 중, 더 낮은 주된 지출 체인 인덱스가 승리하고 다른 하나 (CSI = 6 인 경우)는 유효하지 않은 것으로 간주됩니다

b 우리가 주 체인 (CS)을 가지면, 충돌하는 두 개의 비 직렬 단위 (nonserial units) 사이에서 총 주문을 할 수 있습니다. 주 체인에 직접 놓여있는 유닛을 먼저 색인화합시다. 기원 유닛은 인덱스 0 을 가지며, 기원의 자식 인 다음 CS 유닛은 인덱스 1 을 가지며, 따라서 CS 를 따라 앞으로 나아가면서 우리는 CS 상에있는 유닛에 인덱스를 할당한다. CS 에 있지 않은 유닛의 경우, CS 색인은이 유닛이 처음 (직접 또는

간접적으로) 포함되는 곳에서 찾을 수 있습니다. 이렇게 MCI (MC Index)를 모든 유닛에 할당 할 수 있습니다. 그런 다음 두 개의 비 직렬 장치 중 MCI 가 낮은 장치는 이전에 유효하게 간주되고 다른 장치는 유효하지 않은 것으로 간주됩니다. 두 비 직렬 장치가 동일한 MCI 를 갖는 경우 base64 인코딩으로 표시된 낮은 해시 값을 가진 장치가 유효하다는 tiebreaker 규칙이 있습니다. 최종적으로 잃게 되는 것을 포함하여 우리는 이중 지출의 모든 버전을 유지합니다. DagCoin 은 충돌하는 모든 트랜잭션을 저장하고 어느 트랜잭션을 유효한 것으로 처리 할 것인지를 결정한 최초의 게시 된 작업입니다. 특정 단위에서 만들어진 MC 는이 단위의 저자가 과거 사건의 순서, 즉 역사에 대한 그의 관점에 대해 어떻게 생각하는지 알려줍니다. 순서는 위에 설명 된 대로 어떤 비 직렬 단위가 유효하다고 생각 하는지를 나타냅니다. 주어진 유닛의 모든 부모 중에서 가장 좋은 부모를 선택함으로써, 우리는 동시에 자신의 MC 중에서 선택을 합니다: 해당 유닛의 CS 는 가장 좋은 부모의 CS 가 하나의 링크로 앞으로 확장됩니다. 많은 (또는 심지어 모든) 상위 단위가 공격자에 의해 생성 될 수 있고 최상의 부모 선택이 본질적으로 역사 버전 중 선택이라는 것을 기억하면 우리는 최선의 부모 선택을 요구해야 합니다.

② Witness

a "현실성 테스트"를 원한다면 우리 네트워크의 참가자 중 일부는 평판이 좋지 않은 평판이 좋지 않은 사람들이나 명성이 오래가는 회사 일 수도 있고 네트워크를 건강하게 유지하려는 회사 일 수도 있습니다. 우리는 그 사람들을 증인이라고 부를 것입니다. 그들이 정직하게 행동 할 것으로 기대하는 것이 합리적이지만, 단 하나의 증인을 완전히 신뢰하는 것은 무리입니다. 여러 증인의 DUBU4 주소를 알고 충분히 자주 게시 할 것으로 예상되면 후보 CS 의 현실을 측정하기 위해 CS 가 시간을 거슬러 올라가 목격자 작성 단위를 세십시오 (동일한 증인이 두 번 이상 만난 경우 다시 계산되지 않습니다). 우리는 대다수의 증인이 여행하는 것을 멈출 것입니다. 그래프에서 가장 긴 경로의 길이를 우리가 시작점에서부터 시작점까지 측정합니다. 이 길이를 우리가 중단 한 단위의 수준과 CS 를 테스트하는 부모의 목격 한 수준이라고 합니다. 목격 된 수준이 더 높은 후보 CS 는보다 "실제"로 간주되며, 이 CS 가있는 부모는 최상의 부모로 선택됩니다. 최대 목격 수준을 가진 여러 경쟁자가 있는 경우, 우리는 자신의 레벨이 가장 낮은 상위를 선택할 것입니다. 동점이 지속되면 단위 해시가 가장 작은 부모를 Base64 인코딩으로 선택합니다 이 알고리즘은 증인이 저술 한 단위에 중력을 부여하는 MC 를 선택할 수 있게 하며, 증인은 현실을 대표하는 것으로 간주됩니다. 예를 들어, 공격자가 네트워크의 정직한 부분을 포크로 만들고 자신의 유닛 (섀도 체인)의 긴 체인을 비밀리에 구축하면 그 중 하나가 이중 소비를 포함하고 나중에 자신의 포크를

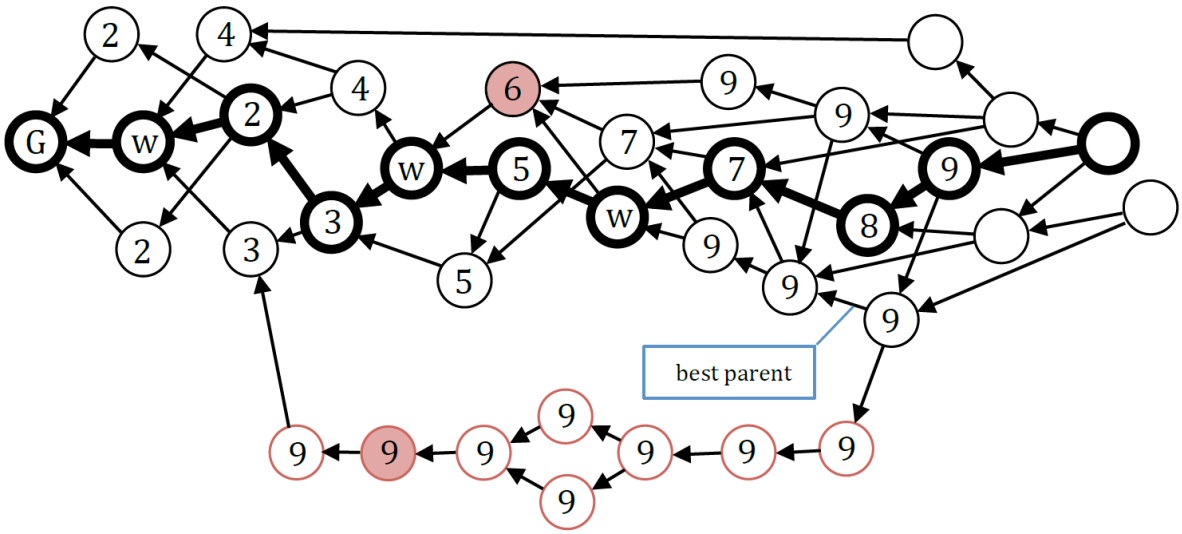
정직한 DAG 로 병합합니다. 합병 지점에서 최상의 부모 선택 알고리즘은 CS 를 정직한 DAG 로 유도하는 부모를 선택합니다. 목격자가 활성화 된 곳이기 때문입니다. 목격자들은 단순히 합병 전에는 목격하지 않았기 때문에 그림자 체인에 게시 할 수 없었습니다. 이 CS 의 선택은 증인과 그들을 임명 한 사용자가 보는 사건의 순서를 반영합니다. 공격이 끝나면 전체 섀도 체인이 CS 에 한 번에 착륙하게 되고 섀도우 체인에 포함 된 이중 소비는 해당 합법적 인 카운터 파트가 합병 시점보다 일찍 오기 때문에 유효하지 않은 것으로 간주됩니다. 이 예는 목격자 대다수가 순차적으로 게시하는 것으로 신뢰해야하는 이유를 보여줍니다. 대다수가 공격자와 공모하여 자신의 섀도우 체인에 게시해서는 안됩니다. 우리는 증인이 현실의 흔적 임을 믿고 어떤 쉐도우 체인에 비 직렬 단위를 게시하지 않을 것을 유의하십시오. 우리는 그들 중 누구에게도 네트워크 또는 그 일부를 제어 할 수 있는 권한을 부여하지 않습니다. 이 작은 의무에도 불구하고, 증인을 지명하는 사용자는 언제든지 자신의 결정을 변경할 수 있습니다. 알려진 실체를 현실의 신호로 보는 생각은 새로운 것이 아닙니다. 오랫동안 알려져 왔으며 일부 회사는 특정 날짜 이전에 일부 데이터가 존재했다는 것을 증명하기 위해 인쇄와 같이 일부 수정하기 어렵고 널리 목격 된 미디어에 데이터를 해시하고 해시를 게시 할 수 있음을 증명하기 위해 이러한 활동에 종사했습니다. 신문. DUBU4 의 증인은 신문과 같은 역할을 합니다. 신문과 마찬가지로, 그들은 잘 알려져 있으며 신뢰할 수 있습니다. 신뢰가 주어지는 데이터를 게시하는 것을 신뢰하는 것에 국한되는 신문의 경우, DUBU4 의 증인은 순차적으로 게시하는 것으로 신뢰할 수 있으며 그 이상은 아닙니다. 신문과 마찬가지로 목격자들은 목격하고있는 해시의 뒤에 무엇이 있는지를 모르고 돌보아 줄 몇 가지 이유가 있습니다. 신문은 수정하기가 어렵지만 (1984 년 가능), 증인이 제작 한 모든 것은 디지털 서명으로 보호되어 수정이 불가능합니다. 안정성을 위해, 우리는 하나뿐 아니라 여러 증인을 보유하고 있으며, 속도와 편리 성을 위해 온라인 상태입니다. 목격자 명부를 결정한 후, 우리는 부모와 그 현실에 대한 정의에 가장 적합한 해당 역사를 "이 증인이 사는 곳"으로 선택하는 것이 가장 좋습니다. 동시에 부모는 서로 다른 목격자 명단을 가질 수 있으며 결과적으로 현실의 정의가 다를 수 있습니다. 우리는 현실의 정의와 그로부터 따르는 역사가 공통점을 수렴하기를 원합니다. 이를 위해 다음과 같은 추가 프로토콜 규칙을 도입합니다. "거의 일치 규칙 (near-conformity rule)": 가장 좋은 부모는 증인 목록이 자녀의 증인 목록과 1 개 이상의 변이가 다른 부모들 중에서 만 선택되어야 합니다. 이 규칙은 CS 에있는 인접한 유닛의 목격자 목록이 충분히 유사하므로 그들의 역사가 대부분 서로 일치한다는 것을 보장합니다. 증인 목록이 0 또는 1 개의 돌연변이가 다른 부모는 직접 포함하는 단위와 호환이 가능하도록 호출되며, 나머지는 호환되지 않습니다. 호환되지 않는 부모는 여전히 허용되지만 부모가 될 기회는 없습니다. 아이가 없는 단위들 사이에

호환 가능한 부모가 없다면 (공격자는 근본적으로 다른 목격자 목록을 가진 자신의 단위로 네트워크를 넘칠 수 있습니다.), 이전 단위에서 부모를 선택해야합니다. 위의 의미는 각 단위가 증인을 열거 할 수 있도록 나열해야 함을 의미합니다. 목격자의 수는 정확하게 12 명이어야 합니다. 이 12 번을 선택했습니다. 사용자가 증인이 신빙성을 상실했다고 생각하거나 보다 나은 후보자가 있다고 생각할 경우 사용자는 증인을 다른 증인과 다를 수 있다는 것을 염두에 두고 증인을 그의 증인으로 교체 할 수 있습니다 단위를 두 개 이상의 위치로 나타냅니다. 이는 모든 변경 사항이 점차적으로 만 발생할 수 있으며 한 위치보다 큰 변경에 대해서는 일반적인 합의가 필요하다는 것을 의미합니다.

i 증인의 가끔 실패 (때때로 부정직하거나 해킹 당하거나 장기간 오프라인 상태가되거나 개인 키를 잃어 영원히 중단 될 수 있음)를 방지하기에 충분히 큰 크기입니다.

ii 사람이 누구인지를 알기 위해 모든 증인을 추적하고 필요한 경우 목록을 변경할 수 있을 만큼 충분히 작습니다.

iii 허용 된 돌연변이는 11 명의 변경되지 않은 증인에 비해 충분히 작습니다



<그림. n> 공격자가 그림자 DAG 를 조명 된 DAG 에 다시 참여 시키면 그의 유닛 그 선택이 그 길을 선호하는 것처럼 가장 좋은 부모가 되기 위해 경쟁을 잃어라. 증인이 더 많다. (w 로 표시)

③ 라이트 클라이언트

a 라이트 클라이언트는 전체 dubu4 데이터베이스를 저장하지 않습니다. 대신, 그들은 관심있는 데이터의 하위 집합 (예: 사용자의 주소는 지출 또는 자금 지원 중입니다. 라이트 클라이언트는 전체 노드에 연결하여 관심있는 유닛을 다운로드합니다. 광 클라이언트는 신뢰하는 증인 목록을 전체 노드에 알립니다 (반드시 필요하지는 않음). 새로운 유닛을 생성하는 데 사용하는 동일한 증인)과 자체 주소 목록. 전체 노드는 라이트 클라이언트가 관심있는 유닛을 검색하고 다음과 같은 방법으로 각 장치에 대한 견고한 체인을 만드십시오:

i 요청한 증인의 과반수까지 CS 를 따라 시간을 거슬러 올라감 충족되었습니다. 이 모든 CS 유닛을 모으십시오.

ii 이 세트의 마지막 유닛 (가장 빠른 시간이기도 함)에서 마지막 세트를 읽습니다.

iii 이 마지막 CS 에서 시작하여 블록이 될 때까지 CS 를 따라 시간을 거슬러 뒤로 걷습니다. skiplist 와 만난다. 이 모든 블록을 수집하십시오.

iv skiplist 를 사용하여 skiplist 에서 참조 된 이전 블록으로 이동합니다. 이 공에는 또한 skiplist 가 있고, 다시 뛰어 오르십시오. skiplist 에 여러 개의 블록이 있는 곳 배열, 항상 가능한 가장 큰 거리로 점프, 그래서 우리는 가속 먼저 10 개의 인덱스로 점프한 다음 100 으로 점프하고 1000 으로 점프합니다.

v 만일 skiplist 에 의한 다음 점프가 우리를 목표 공 뒤에 던지면, 더 작은 거리로 점프하여 감속하십시오. 궁극적으로 skiplist 를 떠나십시오. 부모 링크를 사용하여 한 번에 한 인덱스씩 CS 를 따라 걷습니다. 이 체인은 처음에는 증인 작성 단위를 가지고 있으므로 신뢰할 수 있습니다. 가벼운 클라이언트의 관점에서. 체인의 모든 요소는 상위 단위 링크 (증인 누적 중) 또는 마지막 블록 참조 또는 부모 블록 링크 또는 skiplist 링크로 표시됩니다. 체인의 끝에, 우리는 그 존재가 증명되어야 할 부대를 지니라.

④ Skiplist

a 블록의 일부는 증명의 더 빠른 생성을 가능하게하는 skiplist 배열을 포함합니다 라이트 클라이언트 용 (아래 참조). CS 에 직접 속한 블록과 CS 지수는 10 으로 나눌 수 있으며, skiplist 가 있습니다. skiplist 는 가장 가까운 이전 끝에 있는 숫자가 0 보다 작거나 같은 숫자의 CS 블록에 대한 예를 들어 CSI 190 의 블록에는 CSI 180 에서 블록을 참조하는

skiplist 가 있습니다. CSI 3000 의 블록에는 CSI 2990, 2900 에서 블록을 참조하는 skiplist 가 있습니다

⑤ 다자간 서명

a 단위는 여러 당사자가 서명 할 수 있습니다. 그러한 경우, 저자는 단위는 두 개 이상의 요소를 가집니다. 예를 들어 유용 할 수 있습니다. 둘 이상의 당사자가 계약서에 서명하기를 원할 경우 (평범한 옛날 바보 같은 계약이 아니라 현명한 계약). 그들은 모두 같은 유닛에 서명 할 것입니다. 텍스트 메시지 (app = 'text')가 들어 있습니다. 그들은 전체 텍스트를 저장할 필요가 없습니다. 공개 데이터베이스의 계약 및 지불 - 해시로 충분합니다. (payload_location = 'none'), 당사자는 텍스트를 개인적으로 저장할 수 있습니다. 다자간 서명의 또 다른 적용은 자산 교환입니다. 취하다 사용자 A 는 자산 Y 와 교환하여 자산 X 를 사용자 B 에게 보내려고 합니다 (기본 통화'바이트'는 기본 자산인 자산이기도 합니다. 그런 다음 그들은 두 개의 지불 메시지가 있습니다: 하나의 지불은 자산 X 를 A 에서 B 로 보내고, 다른 하나는 지불은 자산 Y 를 B 에서 A 로 보냅니다. 둘 다 이중 작성 단위에 서명하고 그것을 게시하십시오. 교환은 원자적입니다. 즉, 두 지불 모두에서 실행됩니다. 같은 시간 또는 둘 다 실패합니다. 지불액 중 이중 금액이 이중 지출로 나타난다면 전체 단위는 무효로 처리되고 다른 지불은 무효로 간주됩니다. 이 간단한 구성을 통해 사용자는 중앙 집중 식 거래소에 돈을 의존하고 있습니다.

⑥ 주소

a 주소로 식별되고 거래 출력이 주소로 전송되며, 비트 코인 (Bitcoin)과 마찬가지로 사용자가 여러 주소를 사용하는 것이 좋습니다. 재사용하지 마십시오. 그러나 일부 상황에서는 재사용이 정상입니다. 예를 들어, 증인은 동일한 주소에서 반복적으로 게시할 것으로 예상됩니다. 주소는 Bowl 표현식 인 정의를 나타냅니다 (원격으로 Bitcoin 스크립트와 유사). 사용자가 유닛에 서명을 하면, (일반적으로 ECDSA 서명) 정의에 적용될 때, 이 사용자가 이 서명을 할 권리가 있음을 증명하기 위해 그것을 사실로 평가해야 합니다. 단위. JSON 에서 정의를 작성합니다. 예를 들어, 이것은 다음에 대한 정의입니다. 서명 할 ECDSA 서명이 필요한 주소: ["sig", { "pubkey": "Ald9tkgiUZQQ1djpZgv2ez7xf1ZvYAsTLhudhvn0931w"}] 이 정의는 주소의 소유자가 공공 대응 물은 정의에 (base64 인코딩으로) 주어지며, 그는 이 개인 키를 가진 모든 유닛. 위의 정의는 해당 인증 자에 주어진 서명이 유효 하거나 그렇지 않은 경우 거짓. 그만큼 서명은 인증자를 제외한 장치의 모든 데이터에 대해 계산됩니다. 정의 객체가 주어지면 해당 주소는 초기 정의 객체와 체크섬 입력을 피하기 위해 체크섬이 추가됩니다. 오류. 그러나 일반적인 체크섬 디자인과 달리 체크섬 비트는 점검되지 않은 데이터의 끝 부분에 추가됩니다. 오히려, 그들은 안으로 삽입됩니다. 데이터 내의 여러 위치. 이

디자인은 긴 문자열을 삽입하기가 어렵습니다. 주소가 예상되는 필드의 불법 데이터 주소에서 쓰여진다. base32 인코딩. 위의 정의는 주소에 해당합니다. 예시: A2WWHN7755YZVMXCBLMFWRSLKSZJN3FU

b 주소가 자금을 지원 받으면 지불 한 사람이 알고 있고 지정합니다. 지불 출력의 주소 (정의의 체크섬 된 해시) 만. 정의는 밝혀지지 않았으며 소유자 이외에는 누구에게도 알려지지 않았습니다. 출력이 소비 될 때까지 사용자가 주소에서 첫 번째 단위를 보내면 사용자는 그 정의를 밝혀야합니다 (그래서 서명 확인을 가능하게하기 위해) 저자 배열에서:

```
unit: {
    …
    authors: [ {
            address: 'DJ6LV5GPCLMGRW7ZB55IVGJRPDJPOQU6',
            definition: [
                    "sig",
                    {"pubkey":"AsnvZ3w7N1lZGJ+P+bDZU0DgOwJcGJ51bjsWpEqfqBg6"}
            ],
            authentifiers: {
                    r:
                    '3eQPIFiPVLRwBwEzxUR5thqn+zIFfLXUrzAmgemAqOk35UvDpa4h79Fd6TbPbGfb8VMiJzqdNGHCKyAjl786mw=='
            }
        } ],
    …
}
```

c 사용자가 동일한 주소에서 두 번째 단위를 보내면 정의 (그것은 Byteball 에 이미 알려져 있다). 그는 두 번째 유닛을 그 정의는 안정적이된다. 즉 정의가 드러난 단위는 두 번째 유닛의 마지막 볼 유닛에 포함됩니다. 사용자는 이전 주소를 유지하면서 주소의 정의를 업데이트 할 수 있습니다. 주소. 예를 들어 주소에 연결된 개인 키를 회전하려면 사용자 다음과 같은 메시지가 포함 된 광고 단위를 게시해야 합니다.

```
unit: {
    …
    messages: [
            …
            {
                    app: "address_definition_change",
                    definition_chash: "I4Z7KFNIYTPHPJ5CA5OFC273JQFSZPOX"
            },
            …
    ],
    …
}
```

d 여기서 definition_chash 는 새 주소의 체크섬 된 해시를 나타냅니다. 정의 (나중에까지 밝혀지지 않음)이며, 유닛 자체에 의해 서명되어야 합니다. 오래된 개인 키. 이 주소의 다음 단위는 다음과 같아야 합니다. 그것의 현재 정의. 오히려, 그것의 checksummed 해시와 동일 남아 초기 정의. 정의 변경은 사용자가 키를 변경하고자 할 때 유용합니다 (예를 들어 이전 주소를 유지하면서 새 기기로 이전 할 때 이 주소라면 이미 다른 수명이 긴 정의에 참여하고 있습니다 (아래 참조).

i 마지막 unit 에이 address_definition_change 단위를 포함하십시오. 즉, 이미 안정되어 있다.

ii 저자 배열과 동일한 방식으로 저자 배열에 새로운 정의를 표시합니다. 주소의 첫 번째 메시지 25 명 변경 후 주소는 더 이상 체크섬이 없는 해시와 동일하지 않습니다.

⑦ 다른 주소로 위임

a 주소는 다른 주소에 대한 참조를 포함 할 수 있습니다.

b 다른 주소로 서명을 위임하고 공유를 구축하는 데 유용합니다. 제어 주소 (여러 사용자가 제어하는 주소). 이 구문은 사용자가 자신의 구성 요소 주소의 정의를 변경할 수 있는 유연성 언제든지 다른 사용자를 괴롭히지 않아도 됩니다

```
["and", [
            ["address", "ADDRESS 1 IN BASE32"],
        ["address", "ADDRESS 2 IN BASE32"]
]]
```

3.4 Supported Development with Multi Language

① 블록체인 개발을 위한 새로운 언어의 습득 불필요

a 많은 수의 블록체인 기술이 단 한 종류의 개발 언어만을 지원합니다. 하지만 DUBU4 블록체인의 경우 코어를 포함한 모든 라이브러리를 아래 리스트의 언어로 지원합니다.

i Go

ii Python

iii Java

iv Node.js

v 향후 지속적 추가

3.5 GUI Development Tool

① 누구나 사용할 수 있는 블록체인

a 대부분의 블록체인 기술은 코드 베이스로 되어 있어 프로그래밍 언어를 공부하지 않은 일반인이 어떠한 아이디어만 갖고 있을 때 이것을 실체화 해보기 어려운 상태였습니다.

b DUBU4 블록체인은 코어를 포함한 모든 라이브러리를 GUI 개발 툴 안에 집약시켜 이러한 일반 사용자들이 단 한 줄의 코드도 직접 타이핑 하지 않고 자신만의 코인과 네트워크를 생성 및 보유할 수 있도록 하였습니다.

② GUI 개발 툴 기능 리스트

a 메인넷 생성

i 프라이빗 네트워크

ii 퍼블릭 네트워크

b 코인 생성

i 코인 네이밍 부여

ii 코인 수량 확정

iii 코인 생성 후 코인 증/감 여부 등

c 합의 알고리즘 선택 및 부여

i Based on DUBU4 DAG core

ii POW

iii POS

iv DPOS

v POI

vi CA 서버 검증 등

3.6 Atomic Swap P2P Exchange

① 수수료 없는 거래

a DUBU4 블록체인에서는 주요 코인 30 종 이상을 지원하는 Atomic Swap 거래소를 오픈합니다. DUBU4 의 4 종류 코인을 보유한 모든 사용자들과 DUBU4 Atomic Exchange 에서 지원하는 주요 30 종류의 코인들을 보유한 모든 사용자들은 우리의 거래소를 통해서 어떠한 수수료도 지불하지 않고 사용자들간의 교환 비율 합의만 이루어질 경우 코인의 직접 교환이 가능합니다.

② Based on Hashed Time-locked Contract

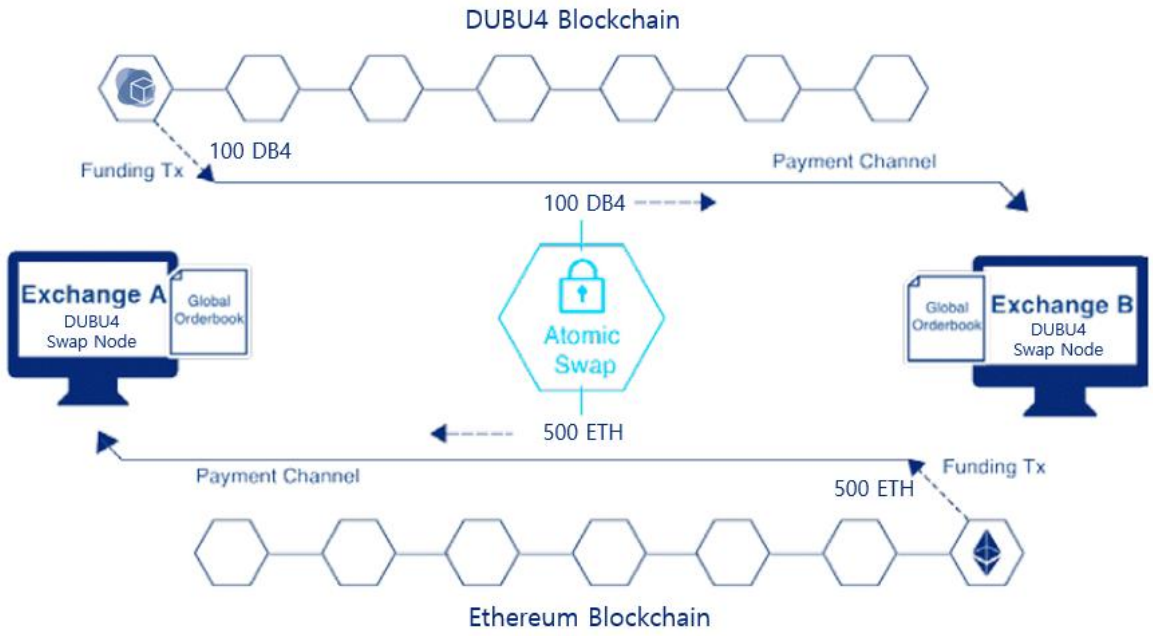
a DUBU4 Atomic Swap 은 현존하는 대부분의 암호화 기능에 사용되는 스크립트 언어의 일부인 Hashed Time-locked Contract 를 기반으로 거래의 수령인이 정해진 기간 내에 지불을 받은 사실에 대해 인정할 것을 요구하며, 수령인이 사실을 인정하는 모든 절차와 과정에 대하여 암호화 증명을 하는 것 입니다.

③ Hashed Time-locked Contract 기술 설명

a 만약 A'코인을 소유한 A 라는 거래자와 B'코인을 소유한 B 라는 거래자가 있을 경우 A 는 A'코인을 전송한다는 내용을 A'코인 블록체인 상에 작성하고, B 는 B'코인을 전송한다는 내용을 B'코인 블록체인 상에 작성을 하게 됩니다. 이 때, 거래 시에 A 혹은 B 의 거래 채널이 Time-locked 기술을 사용하고 있다면, HTLC 로 인하여 Hash-locks 기능을 사용하게 됩니다. 이 상태에서 거래의 수령인이 자신의 해시 값을 공개하고 자신이 교환 받은 코인의 공개된 해시 값을 입력하여 찾게 됩니다. 이러한 과정을 통해서, 거래자들이 2 개 이상의 다른 블록체인 상에 있음에도 불구하고, 거래자 간의 코인 거래가 가능해지는 것이다. 부가적으로, 코인 거래 시 HTLC 를 사용하게 되면 거래 수령자에 의해 수령 보류 상태에 놓인 거래에 대해서는 수령인에게 해당 거래의 코인을 수령 받을 권리를 몰수할 수 있게 한다. 그로 인해 수령인이 수령 보류 중인 거래에 대해서 보다 효과적으로 돈을 돌려 받을 수 있습니다.

④ 코인 리스트

a DUBU4 Atomic Exchange 에서 직접 거래가 가능한 코인의 리스트는 추후 백서 업데이트에 따라서 공개됩니다.



<그림 9. DUBU4 Atomic Swap Exchange 구조>

4 로드맵

<그림 10. DUBU4 로드맵 2017-2018>

4.1 4Q, 2017 – 완료

① DUBU4 재단 설립

② DUBU4 블록체인 핵심가치 정립 및 설계

③ 블록체인 기반 기술 R&D

4.2 1Q, 2018 – 완료

① Coin Type-1 공개

② DUBU4 메인넷(프로토타입) 테스트

4.3 2Q, 2018 – 완료

① 프라이빗 세일 진행(Coin Type-1)

② DUBU4 코어 디테일 R&D

4.4 3Q, 2018 – 진행 중

① Coin Type-2 공개

② Whitepaper 공개

③ ICO 1st 진행

④ 독일/유럽 지역 Meetup 진행

4.5 4Q, 2018 – 예정

① Coin Type-3 공개

② ICO 2nd 진행

③ GUI dev tool 공개(Version 1)

5 참고 문헌

5.1 용어 설명

① 암호화폐 (Cryptocurrency): 피투피(P2P: Peer-to-Peer) 네트워크에서 안전한 거래를 위해 암호화 기술(cryptography)을 사용하는 전자 화폐.

② DAG (Directed Acyclic Graph): 방향성이 있는 비순환 그래프라는 뜻으로 일반적인 블록체인처럼 직렬로 데이터가 검증, 결합되는 것이 아닌 방향성 있는 비순환 그래프의 형태로 교차, 병렬로 데이터가 검증, 결합 되는 것을 말한다.

③ 트랜잭션 (Transaction): 블록체인 안에서 발생되는 단일 거래.

④ POW (Proof-Of-Work, 작업증명): 피투피(P2P: Peer-to-Peer) 네트워크에서 일정 시간 또는 비용을 들여 수행된 컴퓨터 연산 작업을 신뢰하기 위해 참여 당사자 간에 간단히 검증하는 방식. 또는 블록체인(blockchain)에서 정보를 랜덤한 논스(nonce)값과 해시(hash) 알고리즘을 적용시켜 설정된 크기의 해시보다 작은 값을 도출하는 과정으로, 새로운 블록을 블록체인에 추가하는 작업을 완료했음을 증명하는 것.

⑤ POS (Proof-Of-Stake, 지분증명): 어떠한 자산의 지분을 갖고 있다는 증명으로 블록체인 기술 컨텍스트에서는, POW 를 POS 로 대체하는 방법의 대표적 사례로 사용.

5.2 기술 참조

[1] Colin LeMahieu, Nano, "Nano's white paper", https://raiblocks.net/media/RaiBlocks_Whitepaper__English.pdf, 2017

[2] Anton Churyumov, "Byteball's white paper", https://byteball.org/byteball.pdf, 2016

[3] Ethereum Foundation, "Ethereum's white paper", https://github.com/ethereum/wiki/wiki/White-Paper, 2014

[4] Satochi Nakamoto, "Bitcoin: A peer-to-peer electronic cash system", http://bitcoin.org/bitcoin.pdf, 2008

[5] Serguei Popov, "The tangle," 2016