

GİRİŞ

Blockchain & Sui 101

Dijital dünyanın yeni güven ve şeffaflık altyapısı.



Özgür Yazılım ve
Açık Kaynak
Topluluğu

 BuilderMare
 Sui

Blockchain neyi çözer?

Birbirini tanımayan taraflar, güvenilir bir aracı
olmadan durum hakkında nasıl anlaşabilir?

Klasik yaklaşım:

- Merkezi otorite (banka, sunucu)
- Problem: tek hata noktası, sansür,
parazitçilik

Ya sistem kendi kendini doğrularsa?

İÇİNDEKİLER

Blockchain

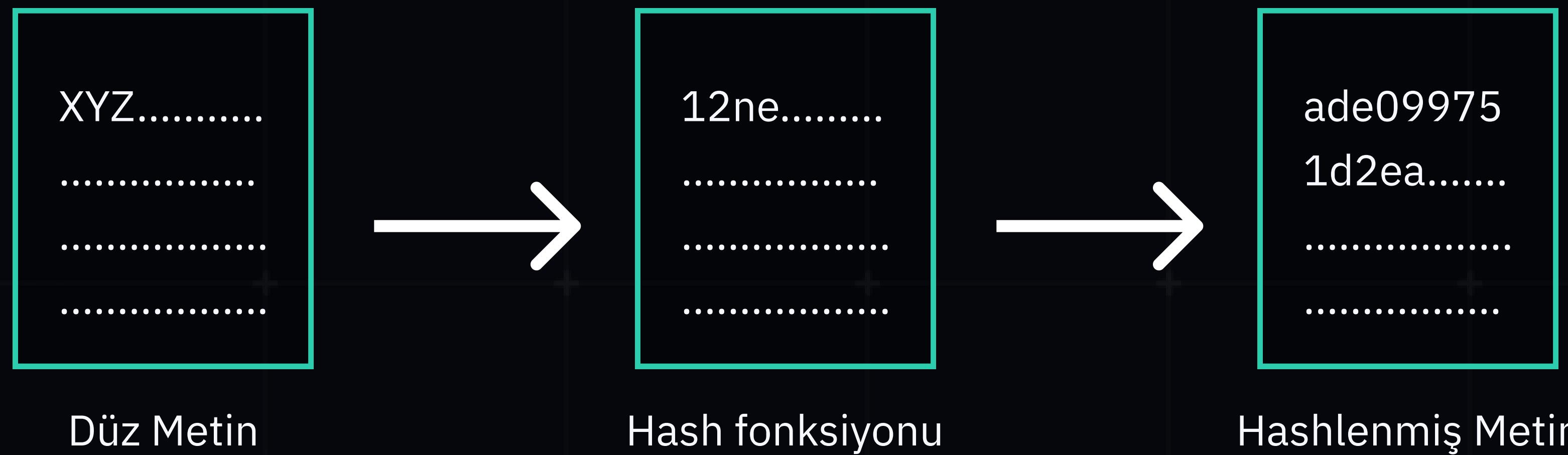
1. Temel yapı taşıları (Hash, bloklar)
2. Her şey nasıl senkronize kalır?
3. Önemli kırılım noktaları
4. Sui'ye bir bakış
5. Siz nasıl dahil olabilirsiniz?

Terminoloji

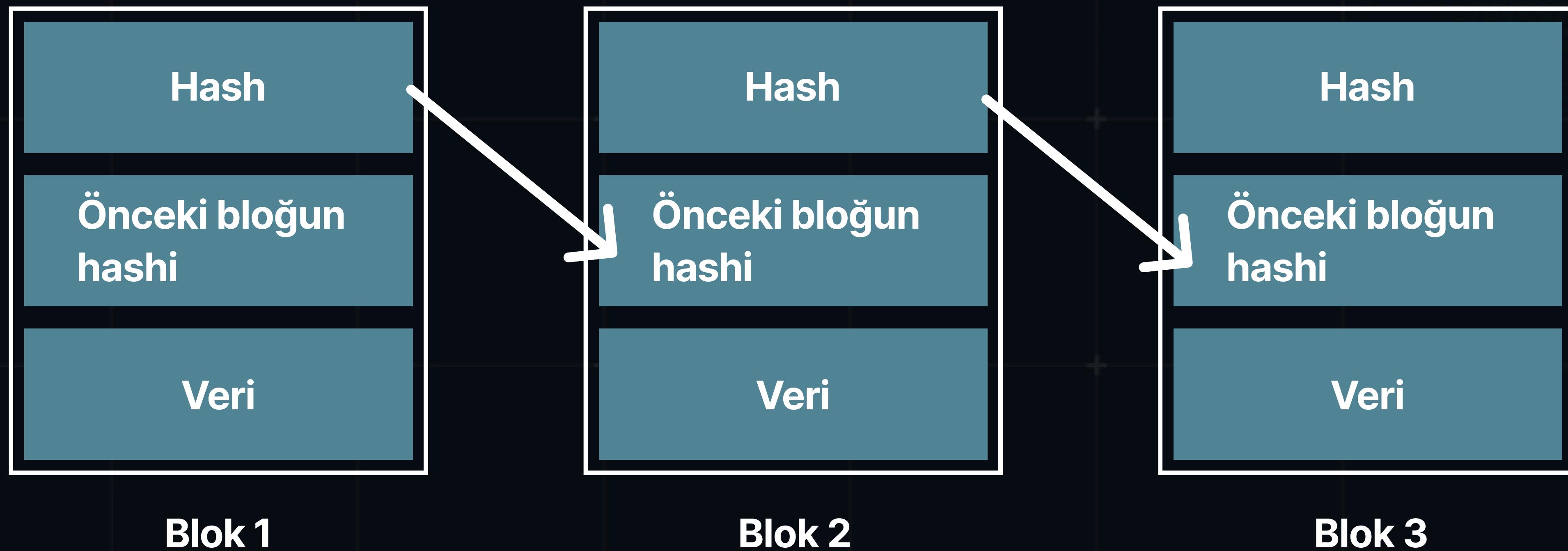
- **Cüzdan (Wallet) nedir?**
Dijital varlıklarınızı (coin, NFT) saklamانızı ve işlem yapmanızı sağlayan program veya cihaz.
- **Akıllı Sözleşme (Smart Contract) nedir?**
Belirli koşullar sağlandığında otomatik olarak çalışan, blockchain üzerinde yaşayan kod parçaları. (Ethereum'un en büyük yeniliği buydu).
- **dApp (Merkeziyetsiz Uygulama) nedir?**
Geleneksel bir sunucu yerine merkeziyetsiz bir blockchain ağı üzerinde çalışan uygulamalar.
- **Gas Ücreti (Gas Fee) nedir?**
Blockchain üzerinde bir işlem (para gönderme, sözleşme çalıştırma) yapmak için ödenen ağ işlem ücreti.
- **TPS (Saniye Başına İşlem) nedir?**
Bir ağın bir saniyede kaç işlemi onaylayabildiğini gösteren hız ölçüsü. (Solana ve Sui'nin öne çıktığı yer burasıdır).

Hash Algoritmaları

- Girdiyi sabit uzunlukta, benzersiz bir özet değere dönüştürür.
- Özgün veriye geri dönüştürülemez.
- Küçük değişiklikler tamamen farklı bir çıktı üretir.



Blockchain nedir?



MERKEZİYETSİZLİK

GELENEKSEL SİSTEM:

- **Nasıl Çalışır:** İşlemleriniz tek bir merkezi otorite (banka, sunucu) üzerinden geçer ve "tek bir kayıt defterine" yazılır.
- Kullanıcı ↔ Merkezi Otorite ↔ Tek Kayıt Defteri

BLOCKCHAIN SİSTEMİ:

- **Nasıl Çalışır:** İşlemleriniz dünya çapında "dağıtık bir ağa" yayınlanır ve kayıt defterinin bir kopyası her katılımcıda (düğümde) tutulur.
- Kullanıcı ↔ Dağıtık Ağ ↔ Kopyalanmış Kayıt Defteri (Her Düğümde)

Sonuçları (Güven ve Dayanıklılık):

- **Sansüre Dayanıklı:** Hiçbir merkezi otorite işlemleri tek başına engelleyemez veya geri alamaz.
- **Şeffaf:** Ağdaki herkes (açık blockchain ise) işlemleri görebilir ve doğrulayabilir.
- **Dayanıklı:** Tek bir sunucunun çökmesi sistemi durdurmaz; "tek hata noktası" yoktur.

Dezavantajı (Verimlilik):

- Her işlemin binlerce bilgisayar tarafından onaylanması gereği için, merkezi sistemlere göre daha yavaş ve daha fazla ek yüze (maliyete) sahiptir.

Nasıl anlaşıyoruz: Konsensüs

- **Problem: Bizans Generalleri Problemi**

Bu, bir ağdaki "kötü aktörlere" rağmen tüm dürüst katılımcıların nasıl ortak bir karara varabileceği sorunudur.

- **Çözüm: Byzantine Fault Tolerance (BFT)**

Bir ağın, katılımcıların bir kısmı hatalı veya kötü niyetli olsa bile çalışmaya devam etmesini sağlayan genel çözüm ailesidir.

- **Sektördeki Temel Yaklaşımlar:**

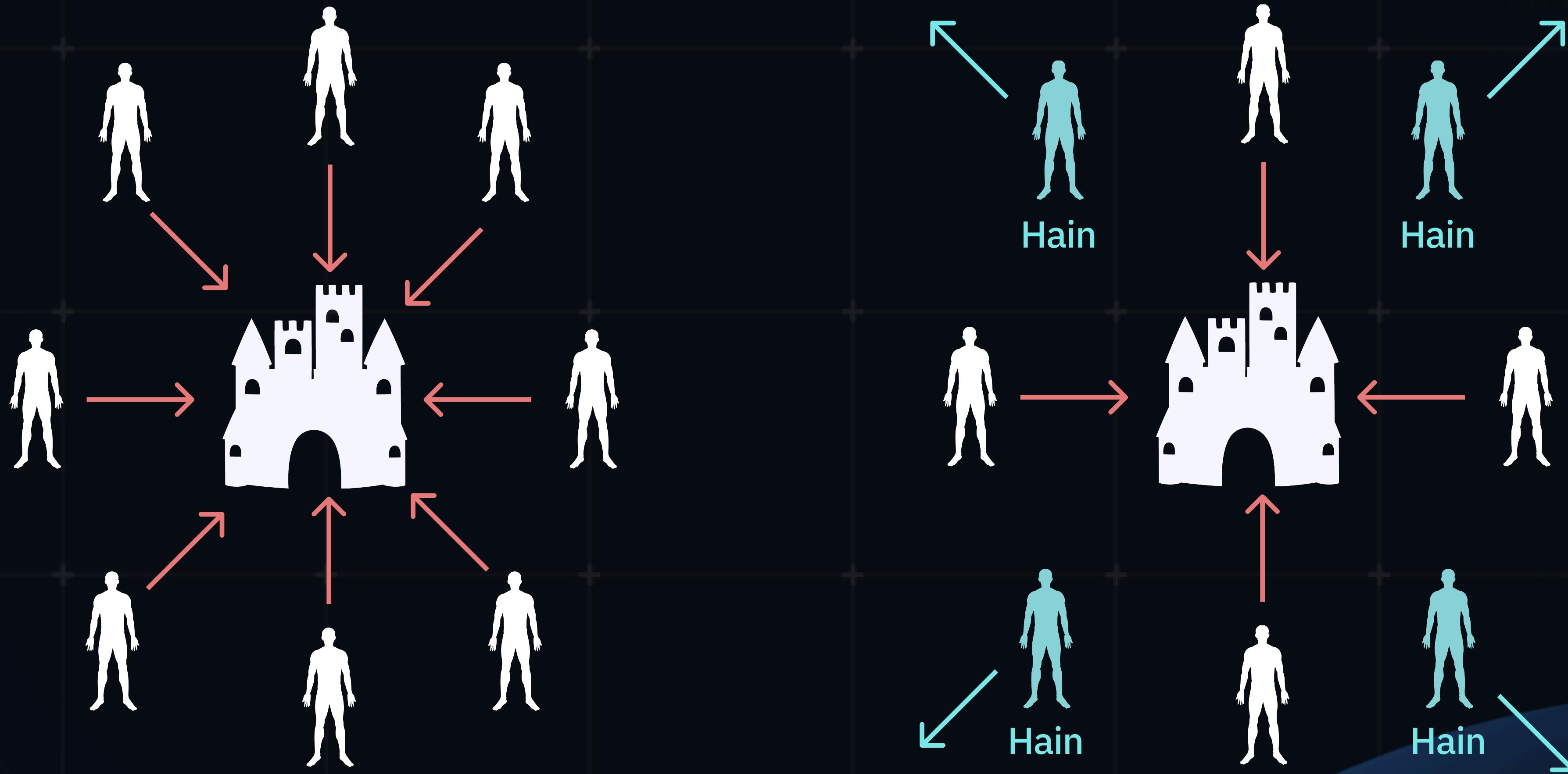
Paxos (ilk temel algoritma) ve Raft (Paxos'un daha uygulanabilir bir alternatifi) gibi algoritmalar, bu güven problemini çözen kanıtlanmış yöntemlerdir.

- **Sui'nin Yaklaşımı:**

Sui, bu anlaşmayı çok yüksek hızlarda ve verimlilikte sağlamak için Mysticeti adında modern ve gelişmiş bir konsensüs algoritması kullanır.

Bizans Generalleri Problemi

Problem: Birbirine güvenmeyen ve bazıları 'hain' (arızalı/kötü niyetli) olabilen bir grup generalin, ortak bir karar (Saldırı/Geri Çekilme) alması gereklidir. Bu, 'hainlere' rağmen tüm 'sadık' katılımcıların nasıl oybirliğine varacağı sorunudur.



Kim Blok Ekleyebilir? (Sybil Koruması)

Bir ağa herkesin katılabildiği bir sistemde (Açık Blockchainler), tek bir kişinin binlerce sahte kimlik oluşturup ağı manipüle etmesini nasıl engelleriz? Bu soruna Sybil Saldırısı denir. Çözüm, katılmanın "maliyetli" olmasını sağlamaktır. Bu "maliyet" bir kanıt gerektirir.

- **Yöntem 1: İş Kanıtı (Proof-of-Work - PoW)**

Nasıl Çalışır: Ağı güvenceye almak ve blok ekleme hakkı kazanmak için hesaplama gücü (enerji) harcarsınız.

Kim Kullanır: Bitcoin

- **Yöntem 2: Hisse Kanıtı (Proof-of-Stake - PoS)**

Nasıl Çalışır: Ağı güvenceye almak için kendi varlıklarınızı (coin) kilitlersiniz. Dürüst davranışlarınız ödül alır, hile yaparsanız kilitlediğiniz varlıklar kaybedersiniz.

Kim Kullanır: Ethereum

- **Peki ya diğerleri?**

- **Sui:** DPoS (Delegated Proof-of-Stake) kullanır. Bu, PoS'un özel bir türündür; hisselerinizi (stake) güvendiğiniz bir doğrulayıcıya (validator) devrederek sürece katılırsınız.
- **Solana:** PoH (Proof-of-History) kullanır. Bu, işlemleri sıralamak için PoS ile birlikte çalışan hibrit bir zaman damgası yöntemidir.

PROOF-OF-WORK (POW)

Mekanizma: Bir Hesaplama Yarışı

- **Madenciler (Miners):** Ağdaki katılımcılar, karmaşık bir matematiksel bulmacayı çözmek için sürekli olarak hesaplama gücü (enerji) harcarlar.
- **Ödül:** Bu bulmacayı ilk çözen, bir sonraki bloğu zincire ekleme hakkını kazanır ve bunun için bir ödül (örn: Bitcoin) alır.
- **Dengeleme:** Ağ, bu bulmacanın zorluğunu otomatik olarak ayarlar; böylece blok üretim süresi (örn: Bitcoin'de ~10 dakika) sabit kalır.

Güvenlik Modeli: Saldırının Maliyeti

- **"51% Saldırısı":** Ağı kandırmak veya sahte işlem yapmak için, bir saldırganın tüm ağdaki toplam hesaplama gücünün yarısından fazmasını (%51) tek başına kontrol etmesi gereklidir.
- **Ekonomik Güvence:** Bu, hem donanım (binlerce özel cihaz) hem de sürekli enerji tüketimi açısından muazzam derecede pahalıdır.
- **Sonuç:** Bu ekonomik model, dürüst bir madenci olarak ödül kazanmayı, ağı aldatmaya çalışmaktan çok daha kârlı hale getirir.

Konsensüs Mekanizmaları (2/2)

PROOF-OF-STAKE (POS)

Mekanizma: Bir Güven Teminatı

- **Validatörler (Doğrulayıcılar):** Katılımcılar, dürüst davranışlarına dair bir "güven teminatı" olarak kendi varlıklarını (sermaye/coin) ağa kilitlerler. Buna "stake" denir.
- **Seçim:** Bir sonraki bloğu kimin ekleyeceği, ağa kilitlediği sermaye miktarına (ve diğer faktörlere) göre rastgele olarak seçilir.
- **Yaptırım (Slashing):** Eğer bir doğrulayıcı hile yapmaya (sahtekarlık) çalışır veya ağa zarar verirse, sisteme kilitlediği sermayenin tamamını veya bir kısmını ceza olarak kaybeder.

Güvenlik Modeli: Saldırının Maliyeti

- **Ekonomik Güvence:** Güvenlik, kilitli olan bu sermaye ile sağlanır. Saldırgan, hile yapmaya çalıştığı anda teminatını kaybeder.
- **Saldırı Maliyeti:** Bir saldırının ağı manipüle etmesi için, sistemdeki kilitli sermayenin çok büyük bir kısmına sahip olması ve bu sermayeyi kaybetmeyi göze alması gereklidir.

Artıları ve Eksileri

- **Artısı:** PoW'dan (İş Kanıtı) çok daha enerji verimlidir.
- **Eksisi:** Servet yoğunlaşırsa (çok az kişide toplanırsa) merkezileşme riski taşır.

Blockchain Türleri

1. Açık Blockchainler

2. Özel veya Kapalı Blockchainler

3. İzne tabi Blockchainler

Açık Blockchainler

- Herkesin tüm işlemleri gördüğü blockchainler
- Bitcoin, Ethereum, Solana ve Sui bunlardan bir kaçı

Özel veya Kapalı Blockchainler

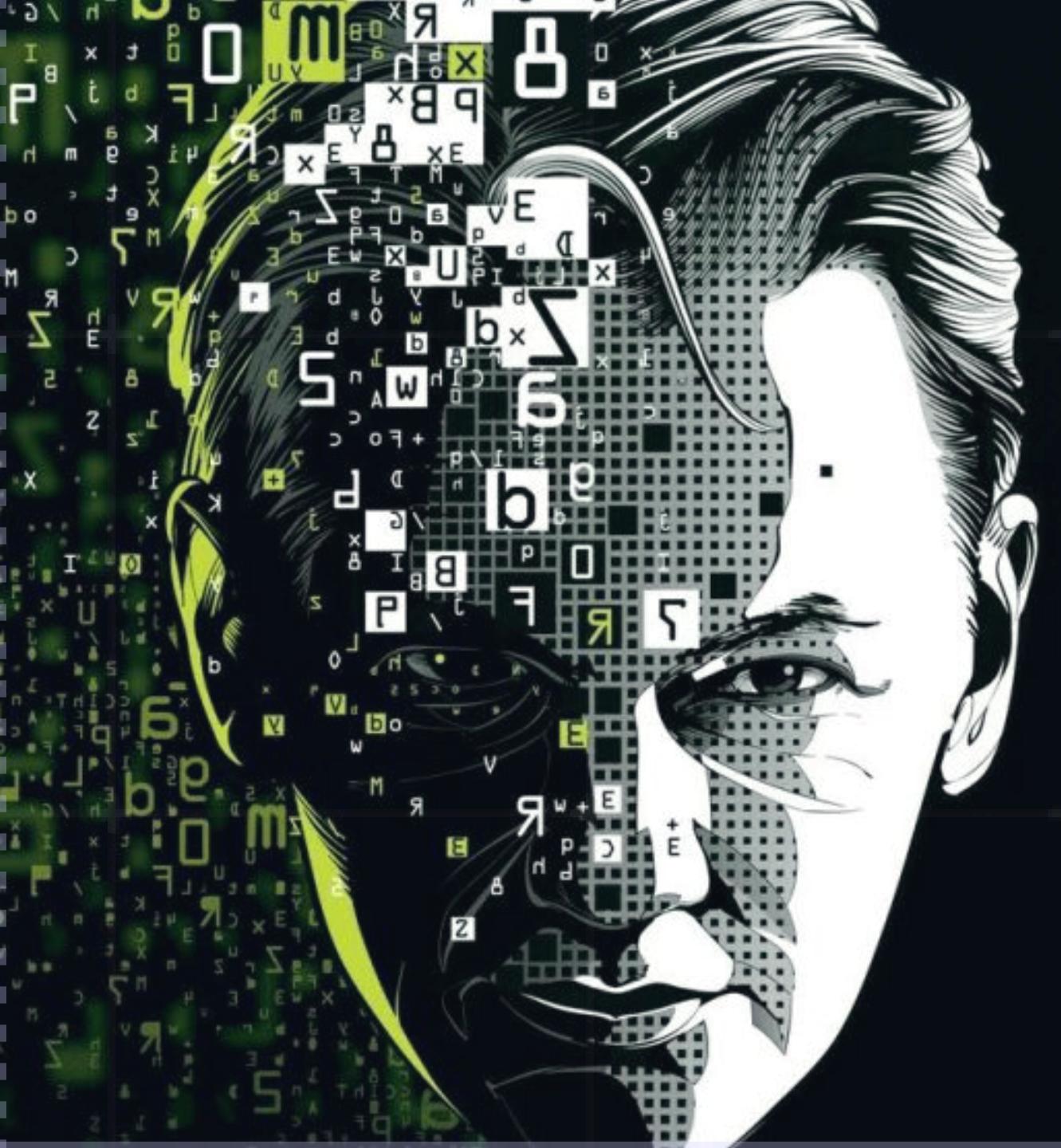
- Katılımı sadece kurum veya kurumlara tahsis edilmiş blockchainler
- Hyperledger ürünleri bunlara örnektir.

İzne tabi Blockchainler

- Kapalı veya Özel blockchainler gibi çalışır.
- İzin işlemi üzerinden yürürlü.
- Bu izne sahipler sisteme dahil olur veya çıkabilir.

Cypherpunk Manifestosu

- Adam Back, Nick Szabo, David Chaum, Hal Finney gibi isimlerin içerisinde bulunduğu bir hareket
- 1992 yılında Timothy May “the Crypto Anarchist Manifesto” makalesi yayınılanıyor.
- Eric Hughes, Tim May ve John Gilmore tarafından gizlilik konuları 1992 yılında tartışılmaya başlıyor.
- 1993 yılında ilk hali online hale geliyor.



The Cypherpunk Manifesto

9 March 1993

“

Privacy is necessary for an open society in the electronic age. Privacy is not secrecy. A private matter is something one doesn't want the whole world to know, but a secret matter is something one doesn't want anybody to know.

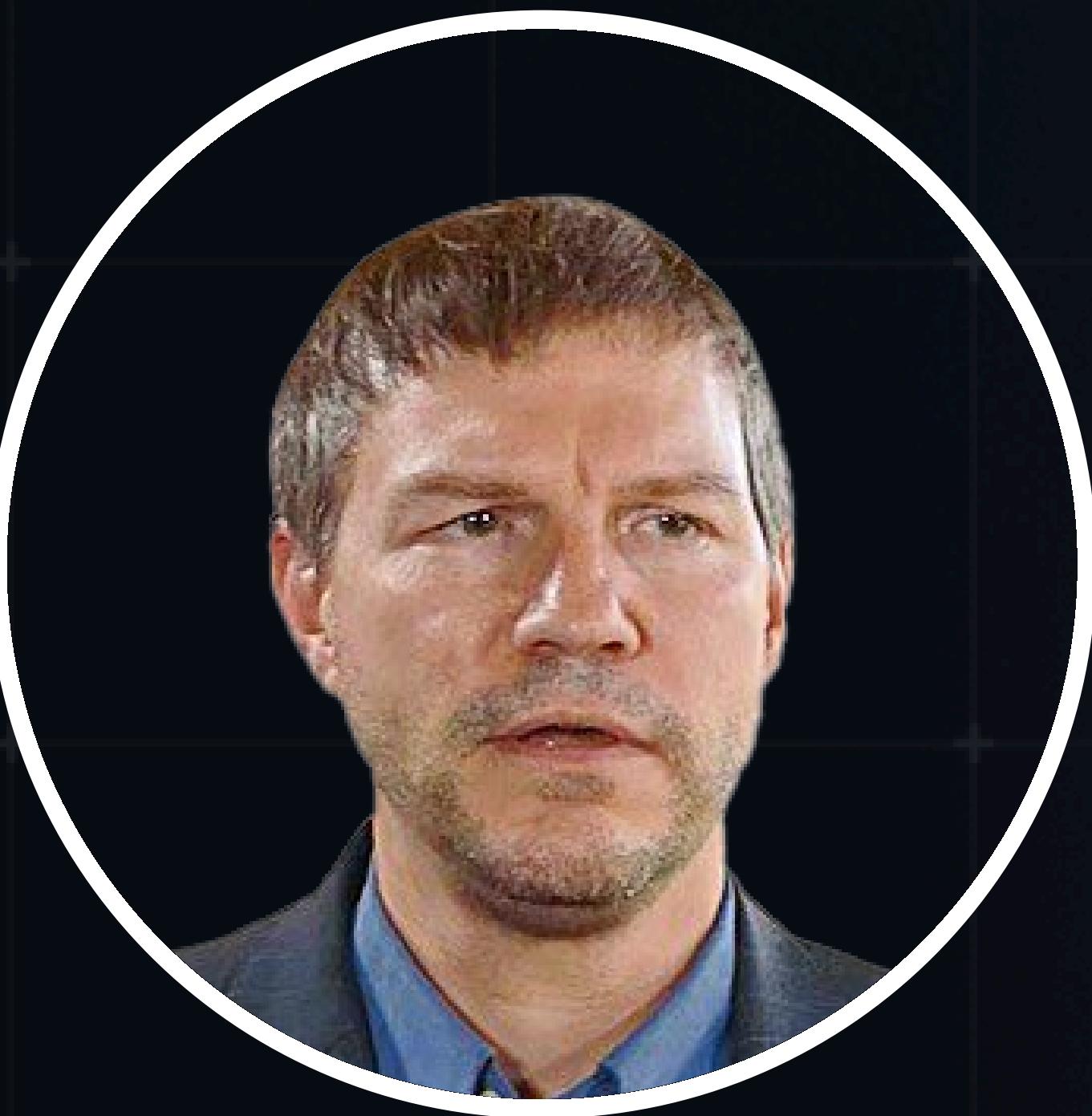
— ERIC HUGHES, A CYPHERPUNK MANIFESTO

Kriptografik Gelişmeler

- 3 Şubat 1989 yılında Leslie Lamport ve Nancy Lynch “Chapter on Distributed Computing” makalesini yayıyor.
- David Chaum ve Stefan Brands eCash protokolü geliştiriyor.
- 1992 yılında Cynthia Dwork ve Moni Naor ilk POW fikri
- 31 ekim 1996 NSA tarafından “Nasıl Darphane Yapılır: Anonim Elektronik Nakitin Kriptografisi” makalesi yayınlandı
- 1997 yılında Adam Back hashcash algoritmasını geliştirdi.
- Hal Finney Yeniden Kullanılabilir İş Kanıtı(RPOW) algoritmasını geliştirdi.

İlk Dağıtık Dijital Paralar

- Nick Szabo, Wei Dai, Adam Back tarafından fikirleri gerçekleştirildi.
- 1998 Yılında ilk Dağıtık Dijital Para fikri
- BFT (Byzantine Fault Tolerance) temeli
- Çifte Harcamayı önleme



NICK SZABO
being Satoshi?

Bitcoin

- 18 Ağustos 2008 bitcoin.org adresinin kayıt edilmesi
- 31 Ekim 2008 tarihinde bitcoinin yazılı olarak hedeflerinin yazılı olduğu makalenin online olması
- 12 Ocak 2009 ilk Bitcoin Transferi. Satoshi Nakamoto ile Hal Finney arasında. 170. blok ile yapılması
- 17 Kasım 2009 yılında Bitcointalk faaliyete geçiyor
- 17 Mart 2010 yılında ilk Bitcoin borsası BitcoinMarket
- 22 Mayıs 2010 yılında pizza alışverişleri ile viral olma



Satoshi **NAKAMATO**
being Nick?

Bitcoin Sonrası

Bitcoin'in başarısından sonra, farklı sorunları çözmeyi amaçlayan birçok yeni protokol ortaya çıkmaya başladı.

- **2011:** Litecoin projesi başladı.
- **2013 (Aralık):** Dogecoin (bir Bitcoin forku) başladı.
- **2013:** Ripple projesi başlatıldı.
- **2014:** RealCoin duyuruldu (daha sonra Tether oldu).

Yeni fikirler:

- **Fikir 1 "Daha hızlı bitcoin":** Litecoin, Bitcoin'in "dijital gümüşü" olarak, daha hızlı blok süreleriyle günlük kullanım için pratik olmayı hedefledi.
- **Fikir 2 "Topluluk & Meme":** Dogecoin, bir şaka olarak başlasa da, bir paranın değerinin güçlü bir topluluk ve internet kültüründen de gelebileceğini gösterdi.
- **Fikir 3 "Finansal Köprü":** Ripple, bankalar arası yavaş ve pahalı uluslararası transferleri saniyelere indirmeyi hedefleyen kurumsal bir çözüm sundu.
- **Fikir 4 "Fiyatı Sabit Para": (Stablecoin)** Tether, kripto piyasasının en büyük sorunu olan aşırı fiyat dalgalanmasını çözmek için 1 coin 1 ABD Doları'na eşitleme fikrini hayata geçirdi.

Zero-Knowledge Proof

Gizliliğin Maksimuma Çıkarılması

- Gizlilik odaklı algoritmaların geliştirilmesi
- Kanıt için sadece istenilen verilerin sunulması

- **Ana İlke:** Veri bütünlüğü, verinin kendisi açıklanmadan, kriptografik kanıtlarla gösterilir.
- **Kimlik ve Yetki:**
 - Kimlik doğrulaması, minimum veri paylaşımı sağlanarak yapılır.
 - Yetki kanıtı gerektiren senaryolarda esnek doğrulama imkanı sunar.
- **Finansal Mahremiyet:**
 - Finansal transferlerde işlem detayları gizli tutulur.
 - Süreçler ve hesap bakiyeleri doğrulanırken mahremiyet korunur.
- **Genel Amaç:**
 - Merkeziyetsiz uygulamalarda (dApps) kullanıcıların gizliliği ön planda tutulur.

Ethereum ve Akıllı Sözleşmeler

- **2013:** Projeye başlanması ve whitepaper'ın yayınlanması.
- **2015:** Ağın ilk versiyonunun ("Frontier") kullanıma açılması.
- **2016:** "The DAO Hack" olayı yaşandı. Bu büyük hack sonrası topluluk, ağı "Ethereum (ETH)" ve "Ethereum Classic (ETC)" olarak ikiye böldü
- **Ethereum nedir?** Bitcoin'in "dijital para" fikrinin bir adım ötesidir. Ethereum, "programlanabilir para" ve "merkeziyetsiz internet" altyapısıdır.
- **Akıllı Sözleşmeler:** Ethereum, "belirli koşullar sağlanınca otomatik çalışan program" mantığını ilk kez popüler ve çalışır hale getirdi.
- **EVM (Ethereum Sanal Makinesi):** Bu akıllı sözleşmeleri çalıştıran "küresel bilgisayar" motorudur. Birçok yeni blockchain projesinin de (forklar) temelini oluşturmuştur.

Solana ve Hızlı Çözümler

Solana: Hız ihtiyacı

- Bitcoin "güvenli para", Ethereum "programlanabilir para" ise, Solana "yüksek performanslı internet" altyapısı olmayı hedefledi.
- Temel felsefesi, merkeziyetsizlikten bir miktar ödün vererek hızı ve düşük maliyeti önceliklendirmekti.
- Bu performans, son kullanıcıya "anlık" ve "ücretsiz" hissi veren bir kullanıcı deneyimi sunmayı amaçladı.

Avantajları Neler?

- **Yüksek Hız (TPS):** Ethereum saniyede ~10-20 işlem yaparken, Solana saniyede 1500+ işlem seviyelerini mümkün kılarak aradaki farkı devasa boyutlara taşıdı.
- **Çok Düşük İşlem Ücreti (Gas):** Bu yüksek hız, ağdaki sıkışıklığı ortadan kaldırarak "Gas Fee" dediğimiz işlem ücretlerini neredeyse sıfıra indirdi.
- **Geliştirici Güvenliği (Rust):** Akıllı sözleşmeler için Rust dilini kullanır. Bu dil, güvenlik ve bellek yönetimi konusunda çok daha sıkı kurallara sahiptir ve daha güvenli sözleşmeler yazmaya olanak tanır.

Sui ile gelen yenilikler

Sui, önceki blockchain'lerin tıkandığı **hız ve kullanıcı deneyimi** sorunlarını temelden çözmek için tasarlandı.

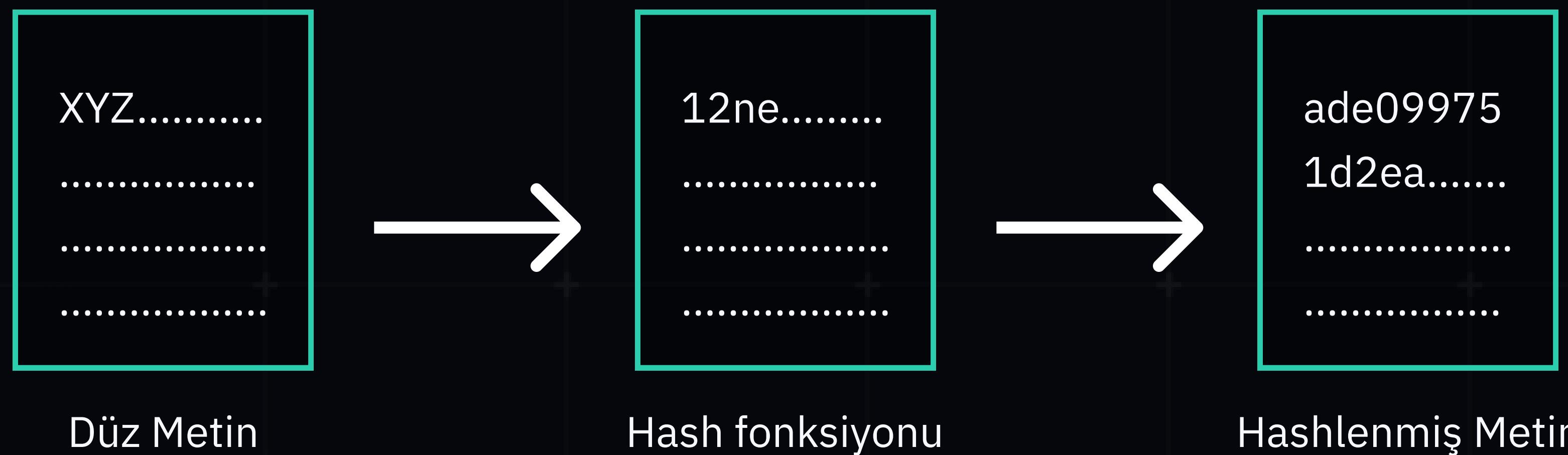
- 1. Muazzam Hız (400.000 TPS):** Diğer ağların aksine **Sui**, birbiriyle alakasız işlemleri (örneğin iki farklı kişinin NFT transferi) sıralamaya sokmaz, paralel olarak işler. Bu, ağın tıkanmasını engeller ve teorik olarak saniyede yüz binlerce işleme (Max ~400k TPS) olanak tanır.
- 2. Kriptosuz Giriş (zkLogin):** Blockchain'in kitlesel benimsenmesindeki en büyük engel olan "**cüzdan kurulumu**" ve "**12 kelime**" (seed phrase) zorunluluğunu ortadan kaldırır. Kullanıcıların doğrudan Google, Mail veya Twitch hesaplarıyla uygulamalara giriş yapabilmesini sağlar.
- 3. Geliştiriciler için Üst Düzey Güvenlik (Move Dili):** **Sui**, özellikle dijital varlıkların (coin, NFT) kopyalanamaması, kaybolamaması veya yanlışlıkla yok edilememesi için tasarlanmış "**Move**" programlama dilini kullanır. Bu, geliştirme aşamasında daha güvenli akıllı sözleşmelerin yazılmasını sağlar.
- 4. Nesne Tabanlı Mimari:** Sui'nin "**özgün çözümlerinin**" temelidir. Her şeyi bir "**hesap bakiyesi**" olarak değil, doğrudan sahip olunabilen "**nesneler**" (objects) olarak ele alır. Bu, geliştiricilere varlıkları programlama konusunda çok daha fazla esneklik ve güç verir.

Şifreleme Algoritmaları

1. Simetrik Şifreleme Algoritmaları
2. Asimetrik Şifreleme Algoritmaları
3. Hash Algoritmaları

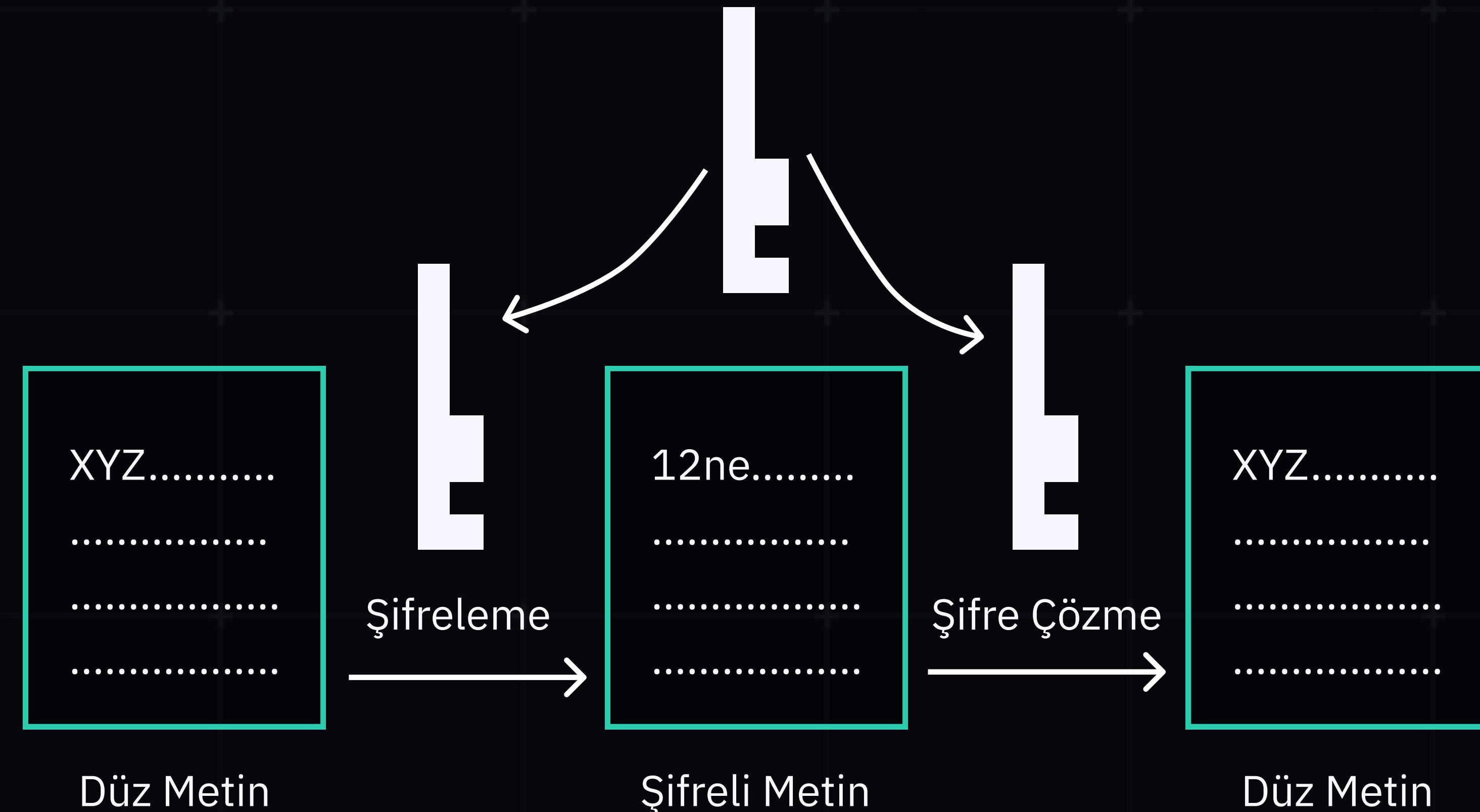
Hash Algoritmaları

- Girdiyi sabit uzunlukta, benzersiz bir özet değere dönüştürür.
- Özgün veriye geri dönüştürülemez.
- Küçük değişiklikler tamamen farklı bir çıktı üretir.



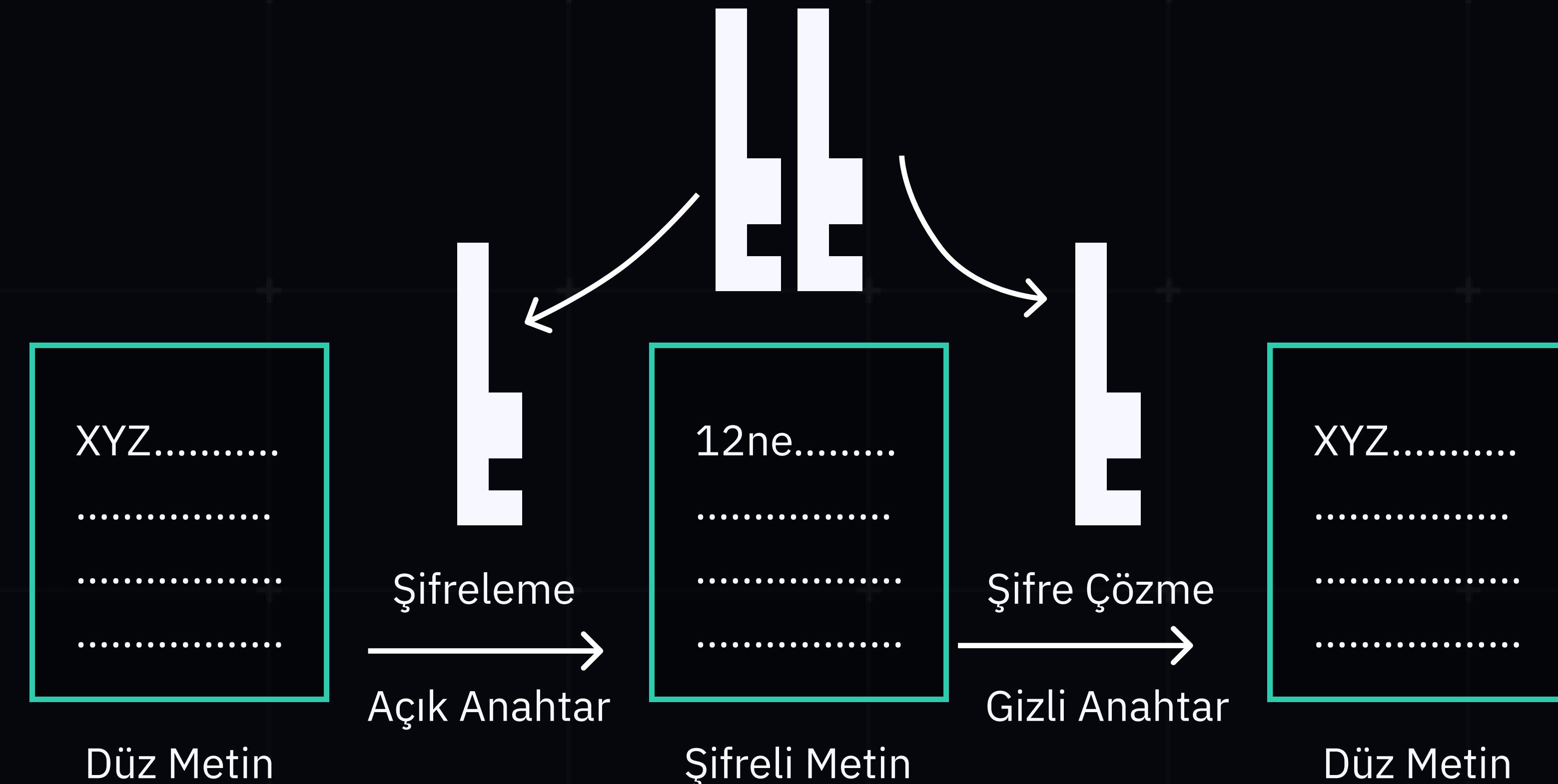
Simetrik Şifreleme Algoritmaları

- Şifreleme ve şifre çözmede aynı gizli anahtar kullanılır



Asimetrik Şifreleme Algoritmaları

- Açık ve gizli Anahtar üzerinden çalışır.
- Bir tanesiyle şifrelenip diğeriyile çözme işlemi gerçekleştirilir.



Sui nedir?

Yüksek performanslı, inovatif blockchain
altyapısı



Özgür Yazılım ve
Açık Kaynak
Topluluğu

 BuilderMare
 Sui

Tarihçe

Köken: Ekip, daha önce Meta'nın (Facebook) Libra/Diem projesinde çalışan ve Move programlama dilini geliştiren çekirdek araştırmacılarından oluşmaktadır.

- **2022:** Ağın halka açık "Testnet" süreci başladı.
- **3 Mayıs 2023:** Mainnet Lansmanı yapılarak ağ resmi olarak açıldı.
- **15 Eylül 2023:** zkLogin duyuruldu; bu özellik, kullanıcıların Google, Mail gibi hesaplarla cüzdan olmadan sisteme giriş yapabilmesini sağladı.
- **2025 Ürün Lansmanları:**
 - **Walrus:** (Merkeziyetsiz veri depolama çözümü)
 - **Seal:** (Şifreli zincir üstü iletişim servisi)
 - **SuiStack:** (Geliştiriciler için mesajlaşma SDK'sı)

SUI EKİBİ

Geliştirici Ekip



EVAN CHENG

Mysten Labs CEO
Eski Facebook Blockchain
Ar-Ge Lideri



SAM BLACKSHEAR

Mysten Labs CTO
Move Dilinin Yaratıcısı

SUI EKİBİ

Geliştirici Ekip



ADENIYI ABIODUN

Mysten Labs CPO



KOSTAS CHALKIAS

Mysten Labs Baş Criptografi
Eski Facebook Criptografi Lideri

Mysten Labs & Vizyon

- **Köken:** Eski Meta (Facebook) **Diem/Novi** projesinin çekirdek ekibi tarafından kuruldu.
- **Amaç:** Milyonlarca kullanıcı için **hızlı, güvenli ve esnek** bir blockchain altyapısı oluşturmak.

Sui'nin Temel Özellikleri

- 1. Nesne Tabanlı Model (Temel Mimari):** Diğer blockchain'leri (Ethereum gibi) dev bir "hesap tablosu" (kimde ne kadar var?) olarak düşünün. Sui ise, sahip olduğunuz her bir varlığı (bir miktar coin, bir NFT) doğrudan size ait, bağımsız bir "Dijital Nesne" olarak ele alır.
- 2. Paralel İşlem Yürütmeye (Yöntem):** Bu nesne tabanlı mimari sayesinde Sui, birbiriyle alakasız işlemleri aynı anda işleyebilir.
 - Eski Yöntem (Sıralı):** A -> B'ye ve C -> D'ye para gönderdiğinde, Ethereum bunları tek bir şeritte sırayla işler (işlem 1, sonra işlem 2).
 - Sui Yöntemi (Paralel):** Bu iki işlem alakasız "nesneler" kullandığı için, Sui bunları aynı anda (çok şeritli bir otoyol gibi) onaylar.
- 3. Yüksek TPS + Düşük Gecikme (Sonuç):** Bu paralel işleme yeteneği, ağıın tıkanmasını engeller. Sonuç? Muazzam derecede yüksek işlem hızı (TPS) ve banka POS cihazı gibi anlık onay (düşük gecikme).
- 4. Move Dili (Güvenlik Katmanı):** Tüm bu "nesneler", dijital varlıklarını korumak için özel olarak tasarlanmış Move dili ile güvence altındadır. Move'un katı kuralları, geliştiricilerin varlıkları yanlışlıkla kopyalamasını veya yok etmesini engelleyerek daha güvenli programlar (akıllı sözleşmeler) yazmasını sağlar.

Bitcoin / Ethereum / Solana Karşılaştırması

Zincir	Model Mimari	Güçlü Yarı	Zayıf Yarı	Sui'nin çözümü
Bitcoin	Global sequential log	Güvenli, basit değer transferi	Çok sınırlı kullanım alanı, yavaş	Sui → akıllı kontratlar, düşük ücret, çok geniş kullanım alanı
Ethereum	Global State Machine	Akıllı kontratlar, en büyük ekosistem	Yavaş, pahalı (gas wars), congestion	Sui → paralel execution + Move dili → daha hızlı, daha güvenli
Solana	High-performance state machine	Hızlı, ucuz	Network outage, tek sıra işlemler → tikanıklık	Sui → object-centric + paralel işlem → gerçek ölçeklenebilirlik
Sui	Object-centric parallel execution	Ölçeklenebilir, kullanıcı dostu, güvenli	Ekosistem yeni	Web3 için kitlelere uygun altyapı

Barındırdığı Projeler ve Ekosistemi

Memecoin	DeFi	NFT		
Blub	Momentum	Popkins		
Miu	Cetus	Prime Machin		
Hippo	Turbos	Karrier		
SuiTrump	SpringSui	Fuddies		
Fud	Bucket			
Lofi				
Axol				
Kappa				
Squirt	Deep	Tallys		
Toilet	Blue	Doonies		
Summon	Typus	Beelievers		
Pugwif	Haedel	Pawtato		
	Aftermath			
	Magma	SuiNS		
	Full sail	Mystic Yetis		
	DoubleUp	Ika		
	SuiMarket	Rootlets		
	Steamm			
AI Agent	Wallet	Lending	GameFi	Other
Vram	Slush	Suilend	SuiPlay	TradePort
Sui AI	Phantom	Navi	Xociety	7K
Inflectiv	Suiet	AlphaLend	Super-B	Sudo Finance
Nodo	Backpack	Scallop	Panzerdogs	Walrus
Circuit Layer	Nightly	Kai Finance	Isekai Blade	DripMarket
aiSUI	OKX wallet	Volo	Cosmocadia	GiveRep

Graphic researched and designed by our team!

Sui Community  @Community_Sui

Hedeflediği Çalışma Alanları

- 1. Kitlelere Erişim (Hesap Soyutlama):** Web3 dışındaki sıradan kullanıcıların, zkLogin (Google/Mail ile giriş) gibi teknolojilerle sistemlere cüzdan karmaşası olmadan, rahatça erişimini sağlamak.
- 2. Fiziksel Dünya Entegrasyonu (IoT & Robotik):** Robot programlama ve Nesnelerin İnterneti (IoT) projeleri için blockchain tabanlı, güvenli ve doğrulanabilir kullanım senaryoları geliştirmek. (Örnek: Dubai'deki Sui entegreli kahve otomati).
- 3. Geleceğin Veri Çözümleri (AI & RWA):** Yapay Zeka (AI) için yüksek hızlı, merkeziyetsiz veri depolama ve Gerçek Dünya Varlıklarının (RWA - örn: gayrimenkul) güvenle saklanması. (Walrus protokolü bu alanlara odaklanır).

Altyapı ve Ürünler

- **Finans ve Ticaret Altyapısı:**

- DeepBook: Hızlı ve merkeziyetsiz likidite (borsa altyapısı) sağlamak.
- Kiosk: Dijital varlıkların (NFT vb.) kiralanması ve alım-satımı için standartlaştırılmış bir yapı sunmak.

- **Veri ve İletişim Altyapısı:**

- Walrus: Programlanabilir, yüksek hızlı veri depolama.
- Seal: Zincir üstü, şifreli iletişim altyapısı.
- Messaging SDK: Geliştiriciler için programlanabilir haberleşme kiti.

- **Oyun ve Kullanıcı Deneyimi:**

- SuiPlay0x1: Web3 oyunları için tasarlanmış özel el konsolu.
- Enoki Platformu: Kullanıcıların "gas ücreti" (işlem ücreti) ödemeden işlem yapabilmesini sağlayan servis.

Blockchain Geliştiriciliği

Blockchain Yazılım Dilleri

1. Domain Spesifik Diller
2. Genel Kullanılan Diller

Domain Spesifik Diller

- Akıllı Sözleşme geliştirmek için veya Zero-Knowledge algoritmaları geliştirmek için kullanılırlar.
 - EVM dünyasında Solidity, Huff, Viper gibi diller yer almaktadır.
 - Facebook'tan çıkışmış bir dil olan Move dili ise Sui, Aptos gibi projelerde kullanılmaktadır.
 - Circum dili ile Zero-Knowledge Proof algoritmalarını implementasyonunu gerçekleştirebiliriz.
-

Genel Kullanılan Diller

- Rust: Hem akıllı sözleşme için hem de blockchainin kendisinde kullanılır. Örnek Sui ağı Rust ile kodlanmaktadır.
- Go: Hem akıllı sözleşme için hem de blockchainin kendisinde kullanılır. Akıllı Sözleşme için Hyperledger Fabric, ağ içinde Geth olarak adlandırılan Ethereum Client gösterilebilir.
- Java: Bazı Hyperledger ürünlerinde kullanılır.
- Javascript/TypeScript: Bazı Hyperledger ürünlerinde ve daha çok dApp geliştirmede kullanılır.

Blockchain Geliştiriciliği

- Akıllı Sözleşme Geliştirme
 - dApp Geliştirme (Merkeziyetsiz Uygulama)
 - Core Blockchain Geliştirme (Çekirdek Ağ)
 - Protokol geliştirme (örn: zkTLS)
-

Temel Araçlar

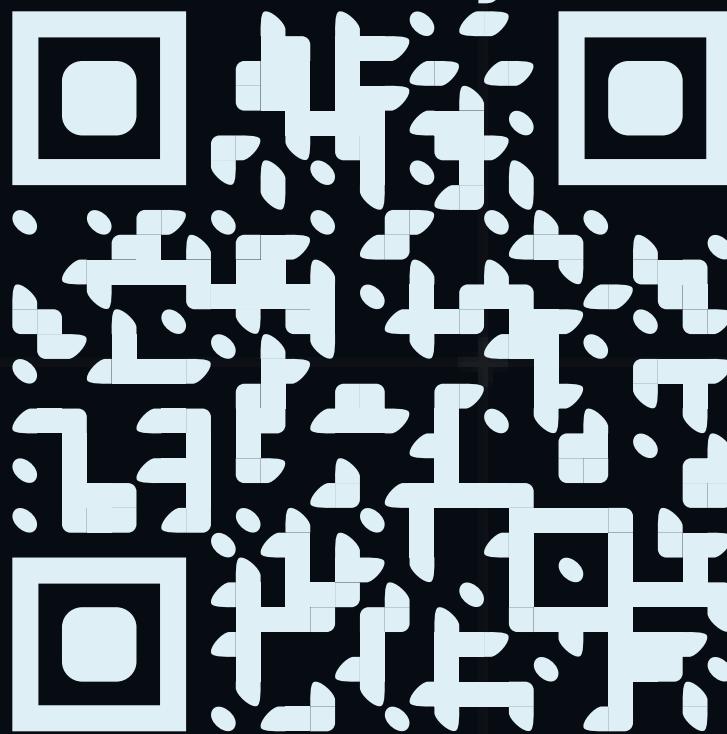
- Kod Editörü: VS Code
- Eklentiler: Sui Move Analyzer, Move Formatter, vb.
- Komut Satırı (CLI): Sui CLI, Hardhat, Foundry, vb.

SON

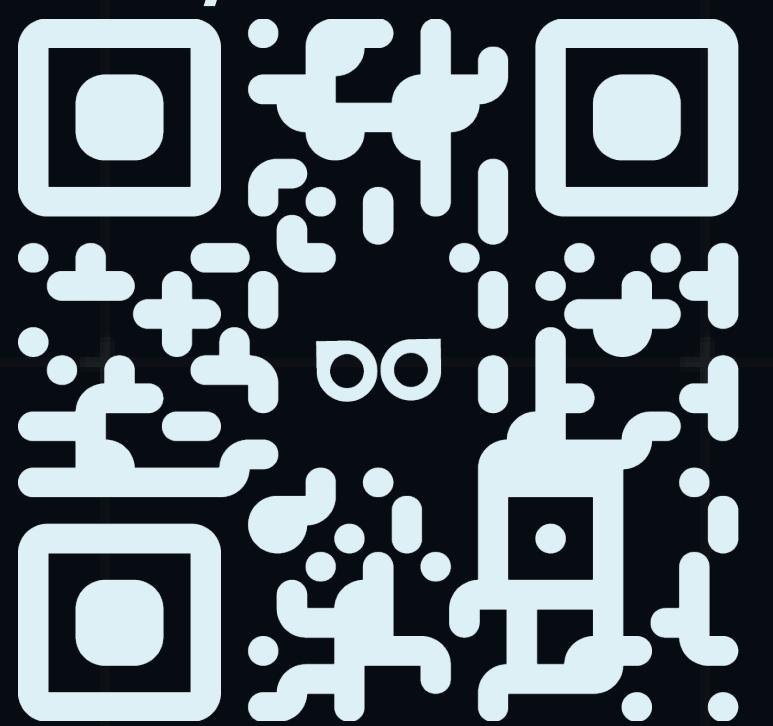
Soru&Cevap

Dinlediğiniz için teşekkürler!

@omuoyak



x.com/buildermare



Özgür Yazılım ve
Açık Kaynak
Topluluğu

 BuilderMare
 Sui