



image: Freepik.com, Web vector created by iurimotov

Blockchain, Criptomoedas & Tecnologias Descentralizadas

Blockchain sem o hype: motivação p/ blockchains

**Prof. Dr. Marcos A. Simplicio Jr. – mjunior@larc.usp.br
Escola Politécnica, Universidade de São Paulo**

Objetivos

- Entender o cenário que motivou a criação do que hoje é conhecido como “blockchain”
 - E, portanto, o tipo de problema para os quais blockchains são particularmente relevantes



Preâmbulo: “sem o hype”


- **Blockchain**: mecanismo distribuído **muito** interessante
- Mas ainda há uma **boa dose de exagero** na literatura sobre sua aplicabilidade e benefícios
 - Como se “descentralização” e “blockchain” fossem sinônimos, e usar blockchain fosse condição essencial p/ ter integridade e transparência
 - Infelizmente, ~90% desse *hype* beira a desinformação...
- A realidade: um blockchain “completo” é
 - **Muito útil** em um cenário: **ordenação de eventos** em sistema **altamente distribuído** e **sem confiança** entre as partes, mas há **incentivo** para sua participação no sistema
 - **Pouco (ou nada) interessante** quando no cenário alvo:
 - Há **entidades totalmente confiáveis** no sistema; ou
 - É **desnecessário ordenar** eventos (e.g., basta sua existência); ou
 - Ordem relativa de eventos não basta (e.g., requer **instantes exatos de tempo**); ou
 - Ataques envolvem **eventos que podem não ser registrados** no blockchain (e.g., atacante age no mundo real, sem benefício/exigência de divulgar ações)

Motivação: ordenação de eventos

- Problema:
 - Em uma rede distribuída, como determinar a ordem em que diversos eventos ocorreram?
 - Ex.: **envio de mensagens**, para determinar a sequência de uma conversa via WhatsApp (e.g., auditoria de mensagens)

A: Vem pra casa hoje?

B: Umas 19

A: Andou flertando com alguém no trabalho? 


B: Lógico que não

A: 



A: Vem pra casa hoje?

B: Lógico que não

A: Andou flertando com alguém no trabalho? 

B: Umas 19

A: 

Motivação: ordenação de eventos



- Problema:
 - Em uma rede distribuída, como determinar a ordem em que diversos eventos ocorreram?
 - Ex.: **transações financeiras**, para saber se usuário tem saldo suficiente no momento de uma compra

Saldo: A,B,C = \$0, D = \$400

A ← D: \$400

A → B: \$100

A → C: \$300

B:  → A C:  → A



Saldo: A,D = \$0,
B=\$100, C=\$300

Saldo: A,B,C = \$0, D = \$400

A → B: \$100

A → C: \$300

A ← D: \$400

B:  → A C:  → A

Saldo: A = \$400, B,C,D = \$0

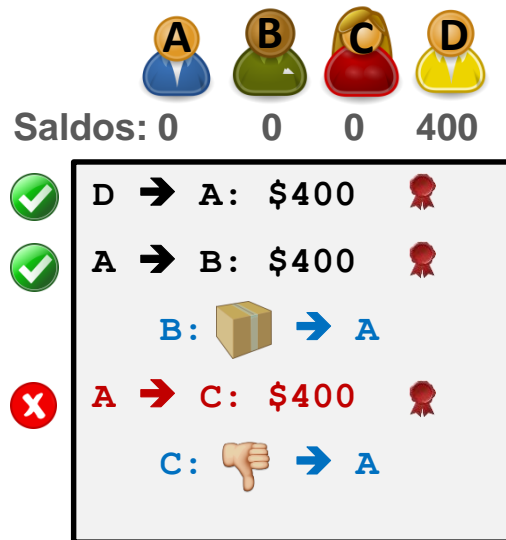
Motivação: ordenação de eventos

- **Nota:** tradicionalmente, problemas desse tipo são resolvidos usando algum grau de **centralização**
- Ex. (saldos): **Confiança** em banco, que controla saldos de usuários... e cobra taxas por isso...
 - Detém grande poder: tecnicamente, poderia “criar dinheiro do nada”, desfazer transações a seu bel prazer, bloquear saldo, ...



Motivação: ordenação de eventos

- Ex. (saldos): **Confiança** em banco, que controla saldos de usuários... e cobra taxas por isso...
 - **Uma tentativa de descentralização**: sabendo (1) o saldo inicial e todas as transações realizadas e (2) a ordem das transações → podemos **verificar o saldo** nós mesmos, e usar **assinaturas digitais** para validar as transações



← **Assinatura digitais** garantem integridade, autenticidade, e irretratabilidade das transações
Ex.: D assina “D → A : 400”


(confiança é **distribuída**)

Motivação: ordenação de eventos

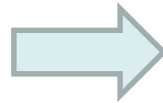
- **Problema:**
 - Em uma rede distribuída, como determinar a ordem que diversos eventos ocorreram?
- **Solução 1:** usuários têm um relógio sincronizado e incluem um carimbo de tempo nas mensagens

18:15 A: Vem pra casa hoje?

18:19 B: Lógico que não

18:18 A: Andou flertando com alguém no trabalho? 


18:16 B: Umas 19



**Analisando
os carimbos**

18:15 A: Vem pra casa hoje?

18:16 B: Umas 19

18:18 A: Andou flertando com alguém no trabalho? 

18:19 B: Lógico que não

- Funciona...?
 - Na verdade, temos 2 perguntas aqui...

Motivação: ordenação de eventos

- **Pergunta 1:** conseguimos sincronizar os relógios de vários usuários em uma rede distribuída?
- **Resposta:** sim, existem algumas soluções!
 - Network Time Protocol (ou similares)
 - NTP v4: RFC 5905 (2010)
 - Consegue atingir acurácia de dezenas de milissegundos entre máquinas na Internet
 - Máquinas com relógios de elevada precisão (e.g., baseados em GPS)
- **Portanto:** podemos sincronizar diferentes máquinas na Internet!
 - O que leva à segunda pergunta...



Motivação: ordenação de eventos

- **Pergunta 2:** podemos **confiar nos carimbos** de tempo embutidos nas mensagens?
- **Resposta:** em várias situações (relevantes!), não...
 - O problema é a **confiabilidade dos usuários**: pode haver benefícios em falsificar carimbos de tempo!
 - Conclusão: Solução 1 (relógio sincronizado) é insuficiente...

Saldo: A,B,C = \$0, D = \$400

09:00 A ← D: \$400

10:05 A → B: \$400

10:06 B: 📦 → A

"10:02" A → C: \$400

"10:03" C: 📦 → A



Saldo: A=\$0, B=\$0, C=\$400 ??

15:10 B → Z: \$400

15:10 Z: 🚔 / 👉 → B

15:11 B: 🤔

15:12 B: 😡 ... 🚫📦 → A!!!

15:13 A: tarde demais 😈








15:13 C: 🙌

Motivação: ordenação de eventos










- **Problema:**
 - Em uma rede distribuída, como determinar a ordem que diversos eventos ocorreram?
- **Solução 2:** Autoridade Certificadora de Tempo (ACT, ou *Timestamp Authority* – TSA) assina eventos



Saldos: A,B,C = \$0, D = \$400

	<u>09:00</u>	A ← D: \$400
	<u>10:05</u>	A → B: \$400
	<u>10:06</u>	B:  → A
	<u>"10:02"</u>	A → C: \$400
	<u>"10:03"</u>	C:  → A

Saldos: A,C=\$0, B=\$400

	<u>10:10</u>	B → Z: \$400
	<u>10:11</u>	Z:  → B
	<u>10:15</u>	C → Z: \$400
	<u>10:16</u>	Z:  /  → C
	<u>10:17</u>	C: 

Motivação: ordenação de eventos

- **Problema:**
 - Em uma rede distribuída, como determinar a ordem que diversos eventos ocorreram?
- **ACT: simples e efetivo. Mas e se não for possível ou desejável usar uma ACT no sistema?**
 - Sistema **totalmente distribuído**, sem entidades confiáveis
 - Que tal uma **ACT distribuída** (e.g., um **blockchain**)?



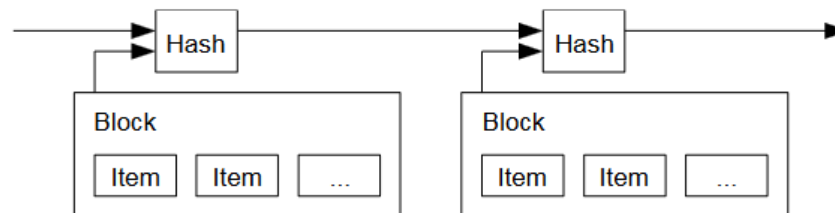
This is blockchain!

Blockchain = ACT distribuída

- FAQ: “Como assim, Blockchain é ‘só’ uma ACT distribuída?! Quem disse isso?!”
- Resposta: um tal de Satoshi Nakamoto
 - Obs.: inventor do Bitcoin, ao descrever o que hoje é chamado de “Blockchain” -- <https://bitcoin.org/bitcoin.pdf>

3. Timestamp Server

The solution we propose begins with a timestamp server. A timestamp server works by taking a hash of a block of items to be timestamped and widely publishing the hash, such as in a newspaper or Usenet post [2-5]. The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash. Each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it.



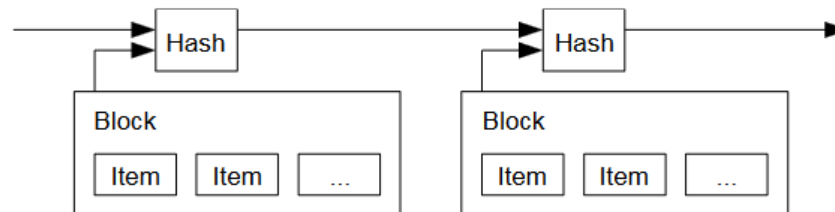
Começando com
cenário centralizado...

Blockchain = ACT distribuída

- FAQ: “Como assim, Blockchain é ‘só’ uma ACT distribuída?! Quem disse isso?!”
- Resposta: um tal de Satoshi Nakamoto
 - Obs.: inventor do Bitcoin, ao descrever o que hoje é chamado de “Blockchain” -- <https://bitcoin.org/bitcoin.pdf>

3. Servidor de carimbo de tempo

The solution we propose begins with a timestamp server. Um servidor de carimbo de tempo calcula o hash de um bloco de itens a serem carimbados e publica amplamente esse hash in a newspaper or Usenet post [2-5]. The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash. Cada carimbo de tempo inclui o carimbo anterior em seu hash, formando uma cadeia h additional timestamp reinforcing the ones before it.



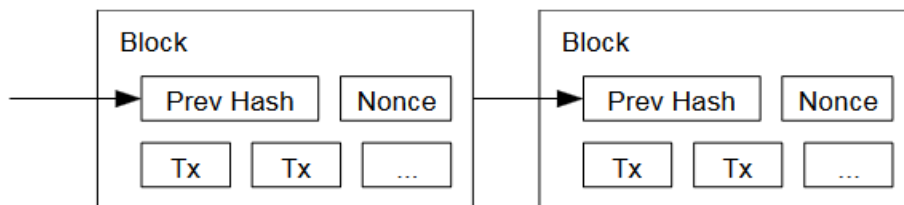
Começando com cenário centralizado...

Blockchain = ACT distribuída

- FAQ: “Como assim, Blockchain é ‘só’ uma ACT distribuída?! Quem disse isso?!”
- Resposta: um tal de Satoshi Nakamoto
 - Obs.: inventor do Bitcoin, ao descrever o que hoje é chamado de “Blockchain” -- <https://bitcoin.org/bitcoin.pdf>

4. Proof-of-Work

To implement a distributed timestamp server on a peer-to-peer basis, we will need to use a proof-of-work system similar to Adam Back's Hashcash [6], rather than newspaper or Usenet posts. The proof-of-work involves scanning for a value that when hashed, such as with SHA-256, the hash begins with a number of zero bits. The average work required is exponential in the number of zero bits required and can be verified by executing a single hash.



... e explicando como distribuir (mecanismo de consenso)

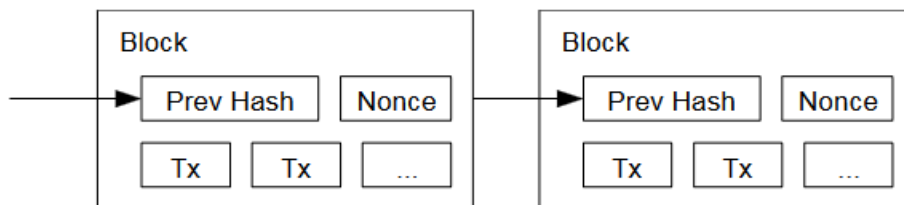
Blockchain = ACT distribuída

- FAQ: “Como assim, Blockchain é ‘só’ uma ACT distribuída?! Quem disse isso?!”
- Resposta: um tal de Satoshi Nakamoto
 - Obs.: inventor do Bitcoin, ao descrever o que hoje é chamado de “Blockchain” -- <https://bitcoin.org/bitcoin.pdf>

4. Prova-de-trabalho

Para implementar um servidor de carimbo de tempo distribuído, em um cenário peer-to-peer precisaremos usar um esquema de prova de trabalho

rather than newspaper or Usenet posts. The proof-of-work involves scanning for a value that when hashed, such as with SHA-256, the hash begins with a number of zero bits. The average work required is exponential in the number of zero bits required and can be verified by executing a single hash.



... e explicando como distribuir (mecanismo de consenso)

Mas então como um blockchain permite construir uma ACT distribuída?



Não perca os próximos episódios!



image: Freepik.com, Web vector created by iurimotov

Blockchain, Criptomoedas & Tecnologias Descentralizadas

Blockchain sem o hype: motivação p/ blockchains

Prof. Dr. Marcos A. Simplicio Jr. – mjunior@larc.usp.br
Escola Politécnica, Universidade de São Paulo

Referências

- S. Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System". Whitepaper, 2008. URL: <https://bitcoin.org/bitcoin.pdf>. Veja também (tradução paara português): <https://cointimes.com.br/whitepaper-do-bitcoin-traduzido/>
- A. Narayanan, J. Bonneau, E. Felten. "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction". Princeton University Press, 2016. ISBN: 0691171696. Available: https://d28rh4a8wq0iu5.cloudfront.net/bitcointech/readings/princeton_bitcoin_book.pdf?a=1
- L. Lantz and D. Cawrey. "Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications". O'Reilly Media, 2020. ISBN: 1492054704
- Stallings, W.; Brown, L. "Computer Security: Principles and Practice" (3rd/4th Ed.), Pearson (2014/2017). ISBN: 9780134794105