UNJAMMING LIGHTNING

Clara Shikhelman
Chaincode Labs

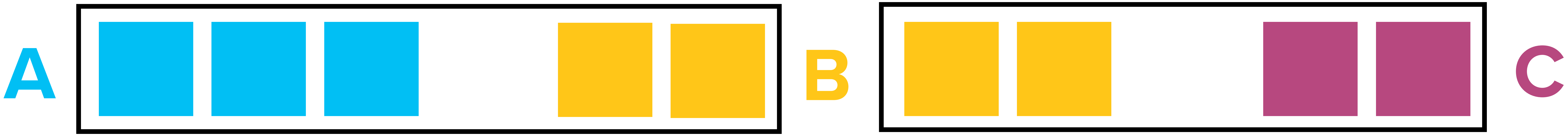
LIGHTNING NETWORK ROUTING

LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments

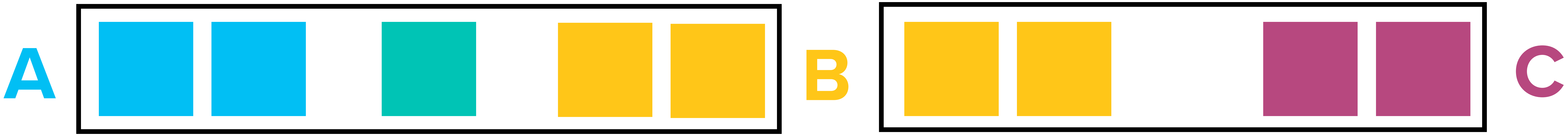
LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



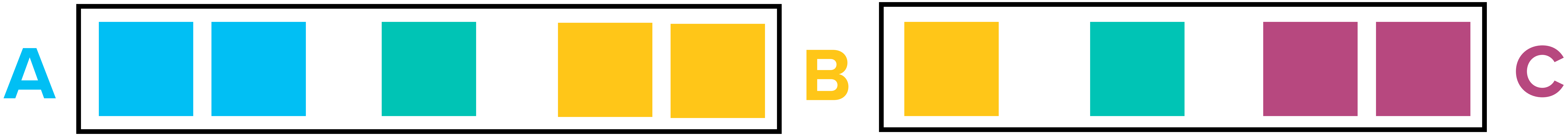
LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



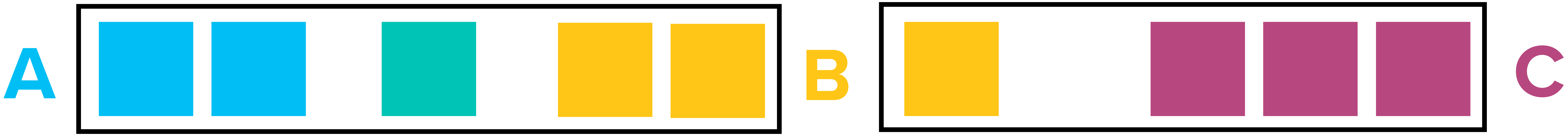
LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



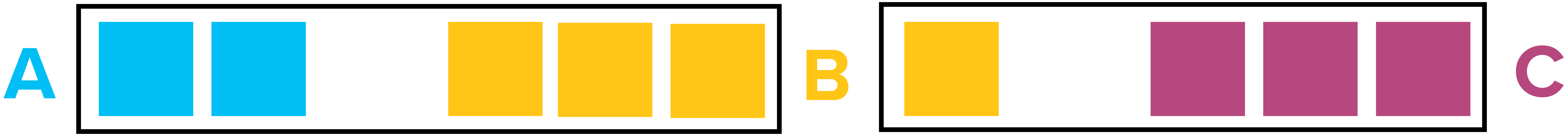
LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



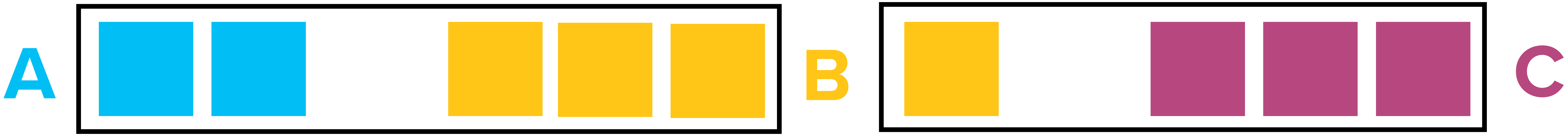
LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



LIGHTNING NETWORK ROUTING

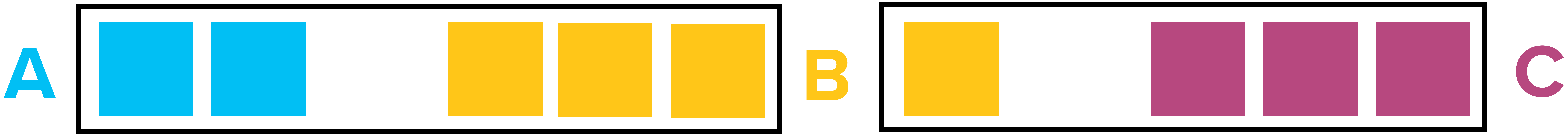
- Alice can route via Bob to Charlie, HTLC for atomic payments



- Bob charges a fee in case of *success*

LIGHTNING NETWORK ROUTING

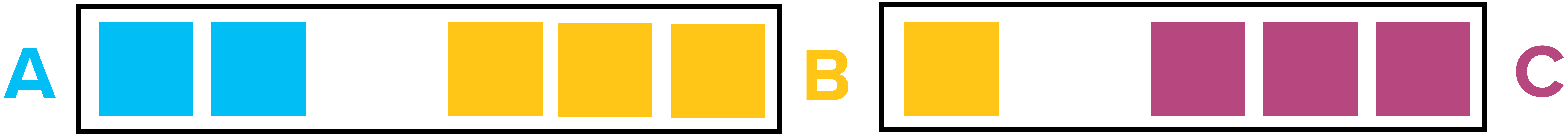
- Alice can route via Bob to Charlie, HTLC for atomic payments



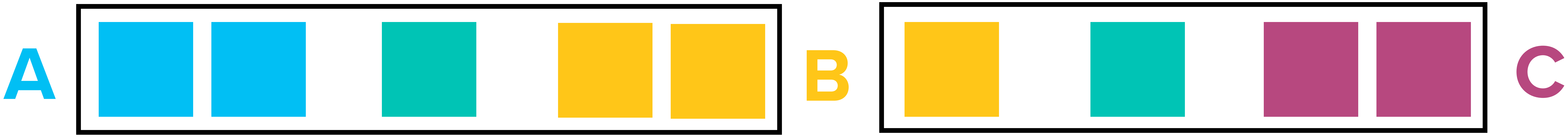
- Bob charges a fee in case of *success*
- But what if Charlie doesn’t give the secret?

LIGHTNING NETWORK ROUTING

- Alice can route via Bob to Charlie, HTLC for atomic payments



- Bob charges a fee in case of *success*
- But what if Charlie doesn't give the secret?

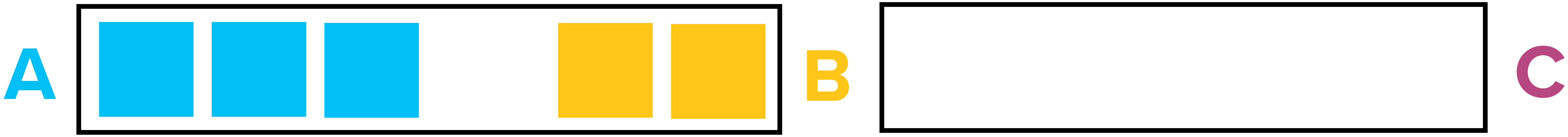


INTRO TO JAMMING

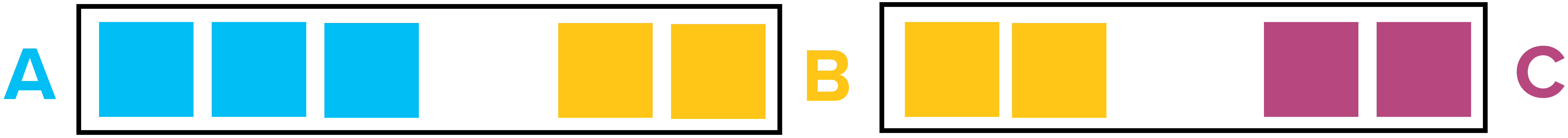
INTRO TO JAMMING

A B C

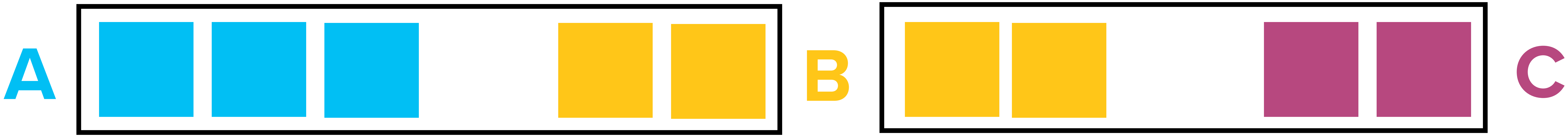
INTRO TO JAMMING



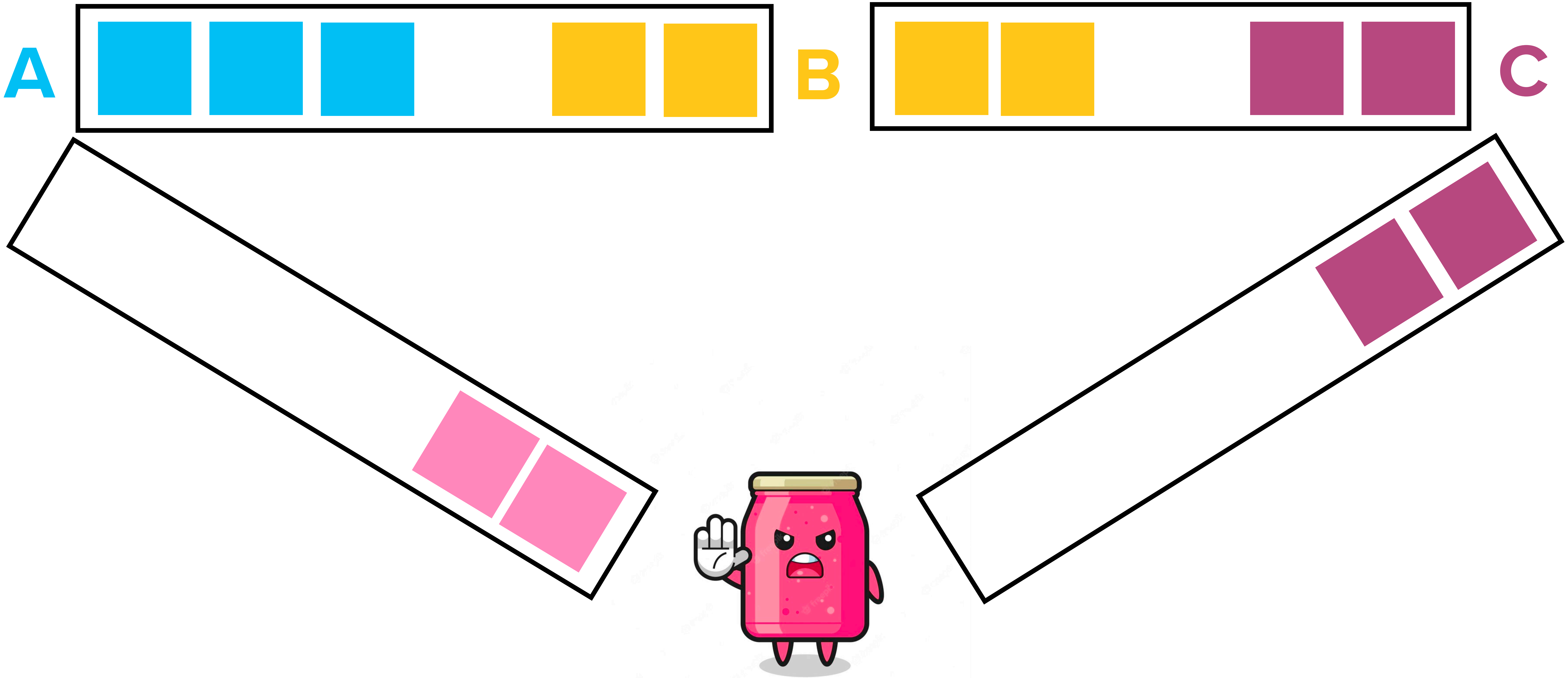
INTRO TO JAMMING



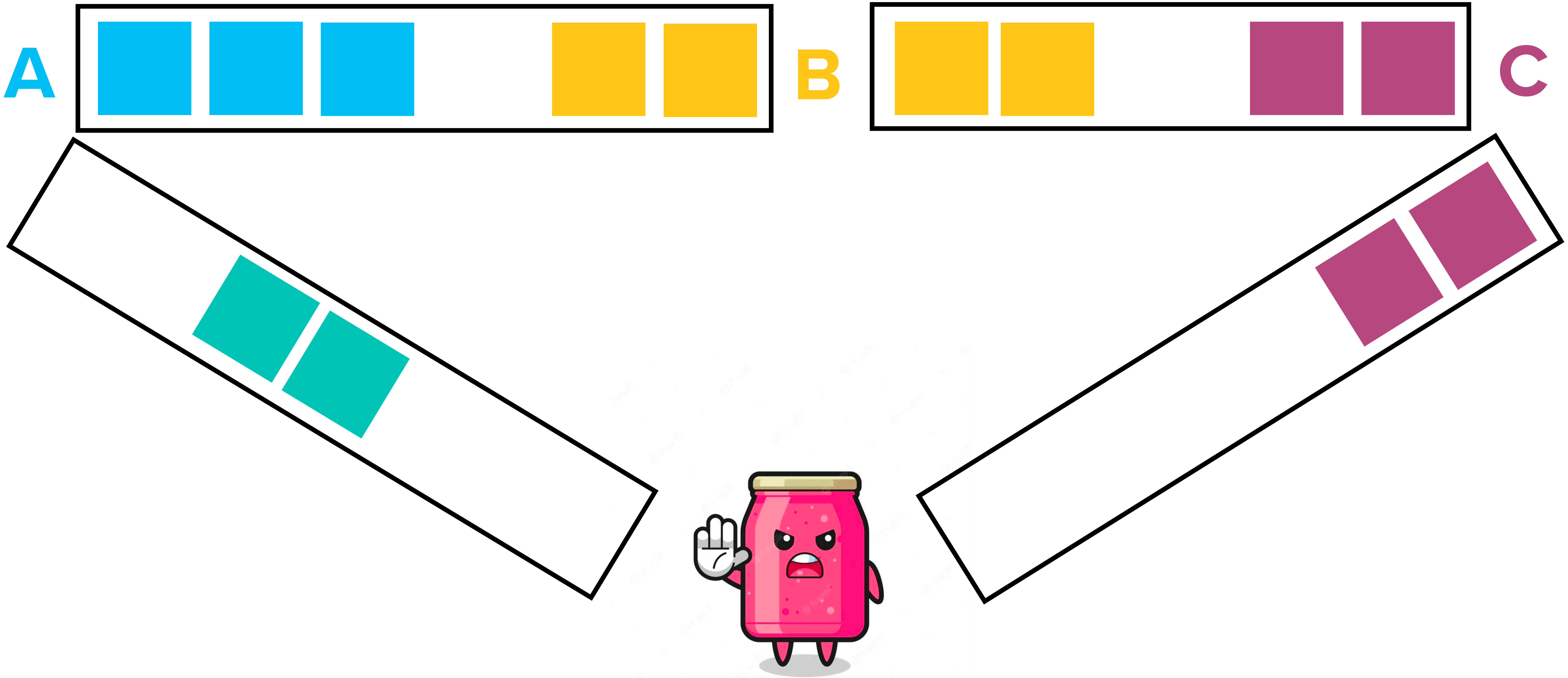
INTRO TO JAMMING



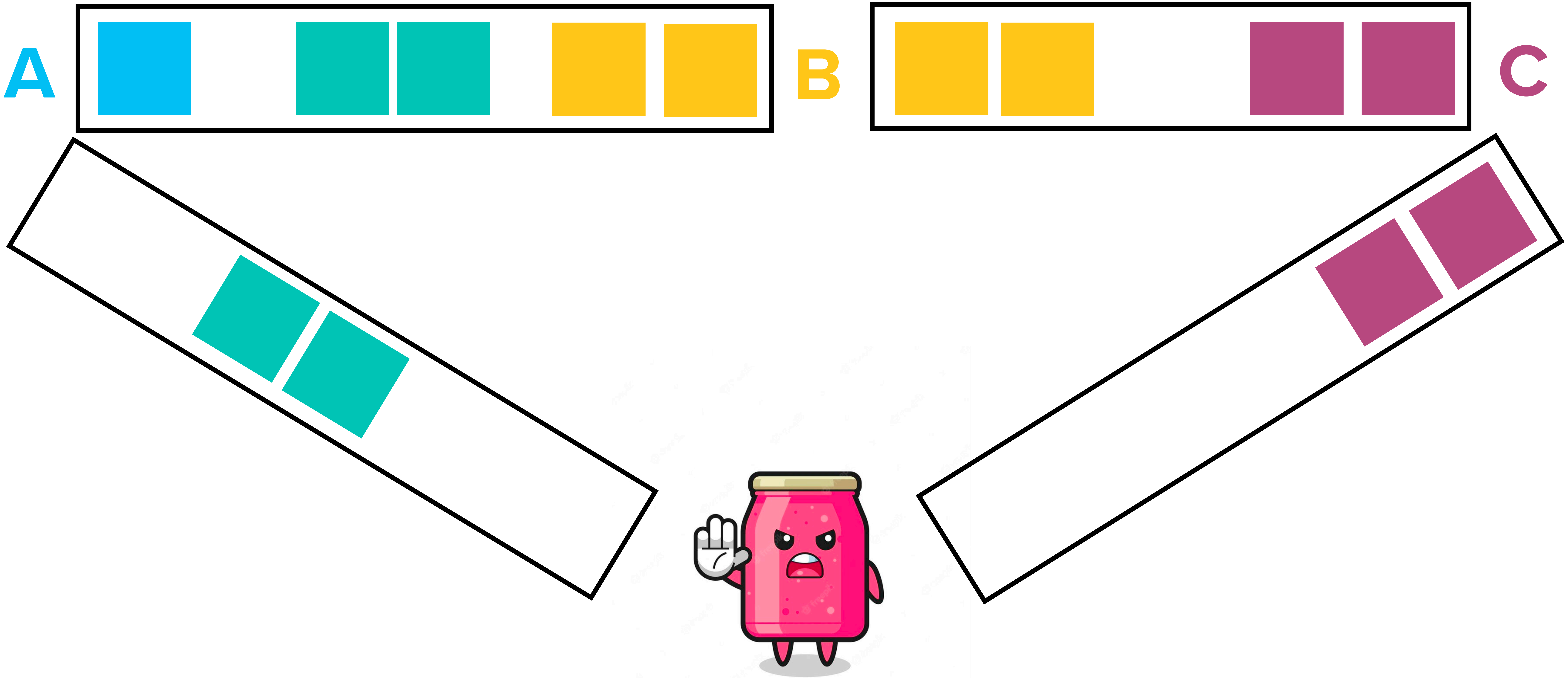
INTRO TO JAMMING



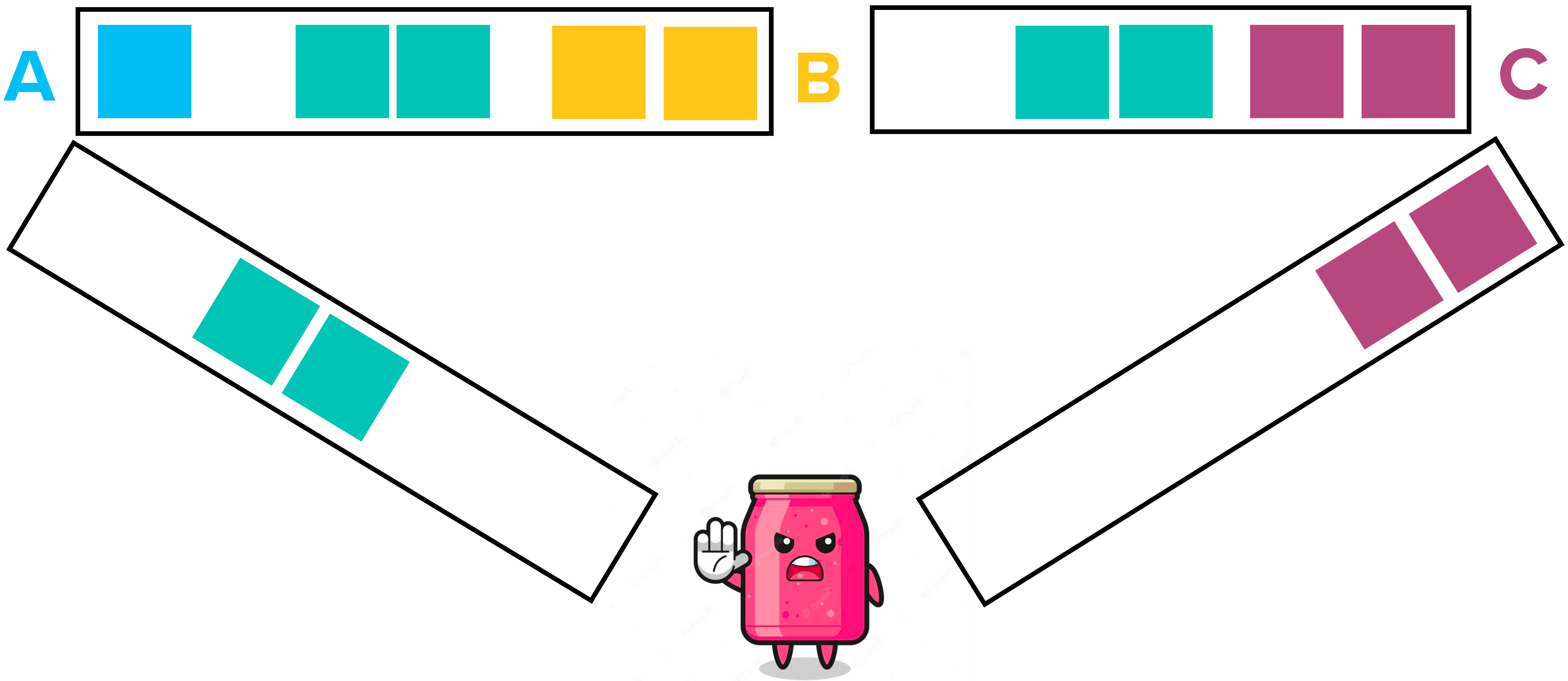
INTRO TO JAMMING



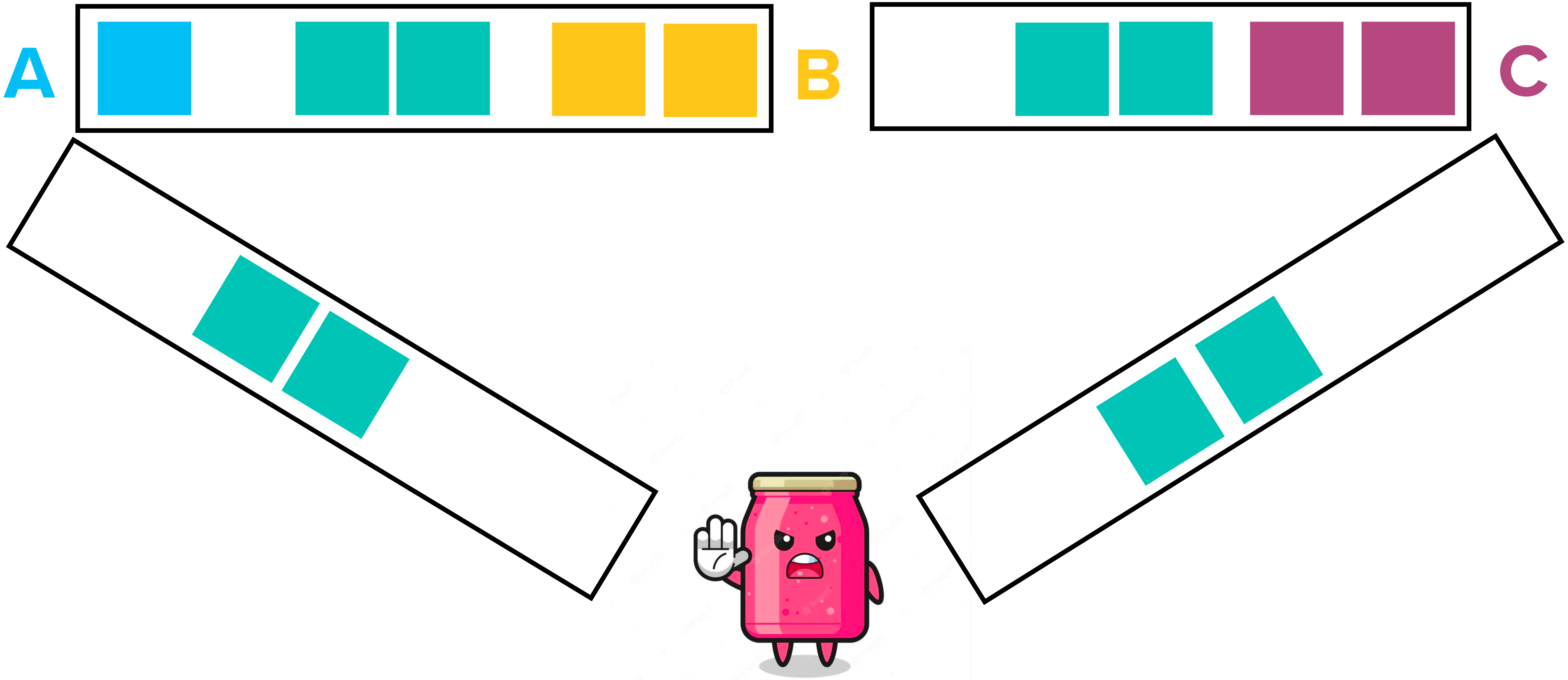
INTRO TO JAMMING



INTRO TO JAMMING



INTRO TO JAMMING



INTRO TO JAMMING

INTRO TO JAMMING

- Channels have two types of scarce resources

INTRO TO JAMMING

- Channels have two types of scarce resources
  - Liquidity (≤channel capacity) - satoshis are locked until resolved

INTRO TO JAMMING

- Channels have two types of scarce resources
    - Liquidity ($\leq$channel capacity) - satoshis are locked until resolved
    - Slots ($\sim$483) - a payment takes a slot until resolved

INTRO TO JAMMING

- Channels have two types of scarce resources
  - Liquidity ($\leq$channel capacity) - satoshis are locked until resolved
  - Slots ($\sim$483) - a payment takes a slot until resolved
- A Jammer locks *all* of the liquidity or *all* of the slots

MOTIVATION

MOTIVATION

MOTIVATION

- Attacking a business competitor

MOTIVATION

- Attacking a business competitor
  - Routing node

MOTIVATION

- Attacking a business competitor
  - Routing node
  - Service provider

MOTIVATION

- Attacking a business competitor
  - Routing node
  - Service provider
- Network-level attacks

MOTIVATION

- Attacking a business competitor
  - Routing node
  - Service provider
- Network-level attacks
  - Disconnecting nodes

MOTIVATION

- Attacking a business competitor
    - Routing node
    - Service provider
- Network-level attacks
    - Disconnecting nodes
    - Pushing the flow towards a specific node

TWO JAM FLAVORS

TWO JAM FLAVORS

TWO JAM FLAVORS

	SLOW

TWO JAM FLAVORS

	SLOW	QUICK

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve		

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?		

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?	Easy	

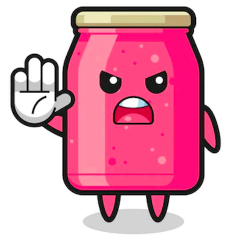
TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?	Easy	Hard

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?	Easy	Hard
Solution		

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?	Easy	Hard
Solution	Reputation	

TWO JAM FLAVORS

	SLOW	QUICK
Time to resolve	Hours/days	Seconds
Detectable?	Easy	Hard
Solution	Reputation	Fees

SOLUTION OVERVIEW

SOLUTION OVERVIEW

- Local Peer Reputation

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors
  - Reputation gives access liquidity and slot

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors
  - Reputation gives access liquidity and slot
- Unconditional Fee

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors
  - Reputation gives access liquidity and slot
- Unconditional Fee
  - Mitigates quick jamming

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors
  - Reputation gives access liquidity and slot
- Unconditional Fee
  - Mitigates quick jamming
  - Paid even if the payment fails

SOLUTION OVERVIEW

- Local Peer Reputation
  - Mitigates slow jams
  - Each node assigns reputation to its neighbors
  - Reputation gives access liquidity and slot
- Unconditional Fee
  - Mitigates quick jamming
  - Paid even if the payment fails
  - Compensates jammed node

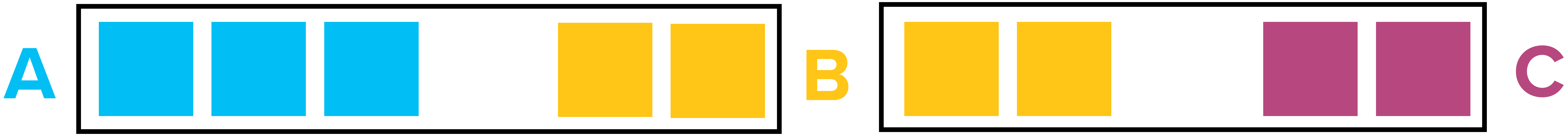
REPUTATION OVERVIEW

REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource

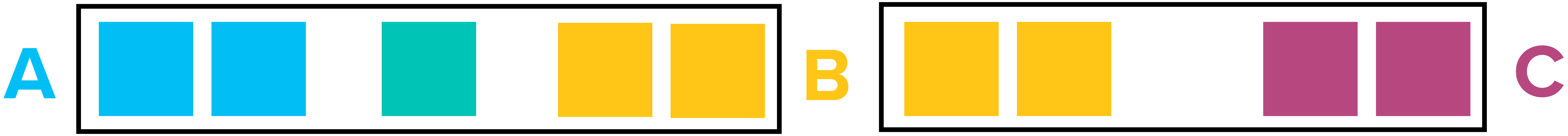
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



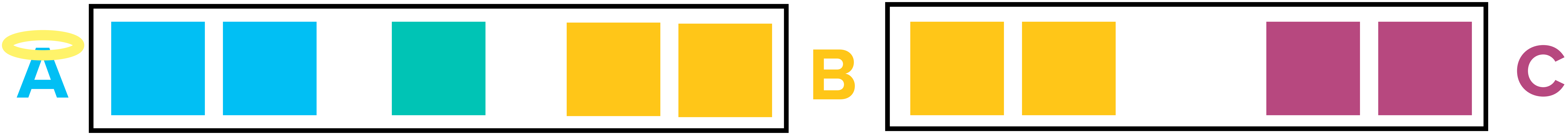
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



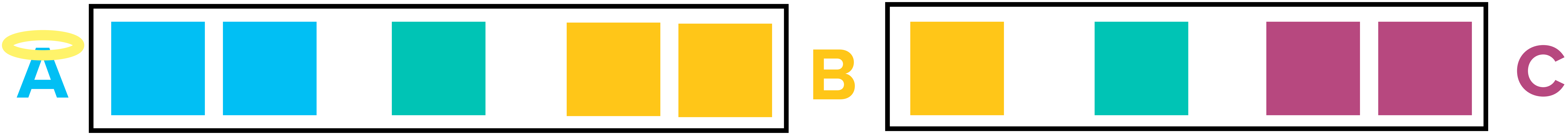
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



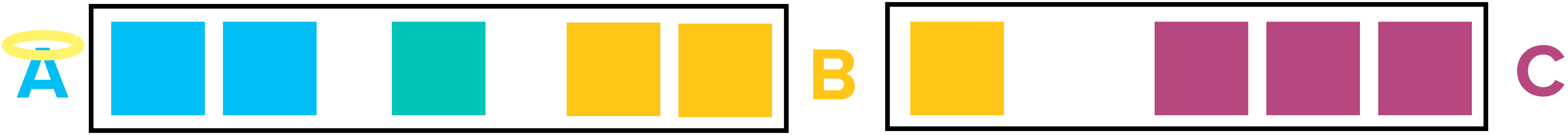
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



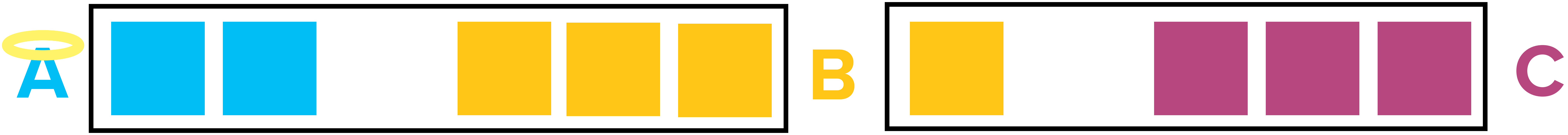
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



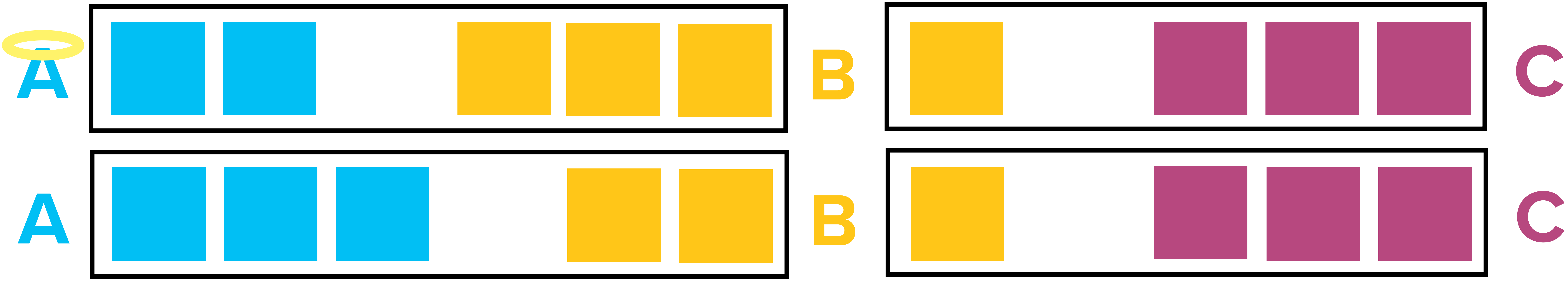
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



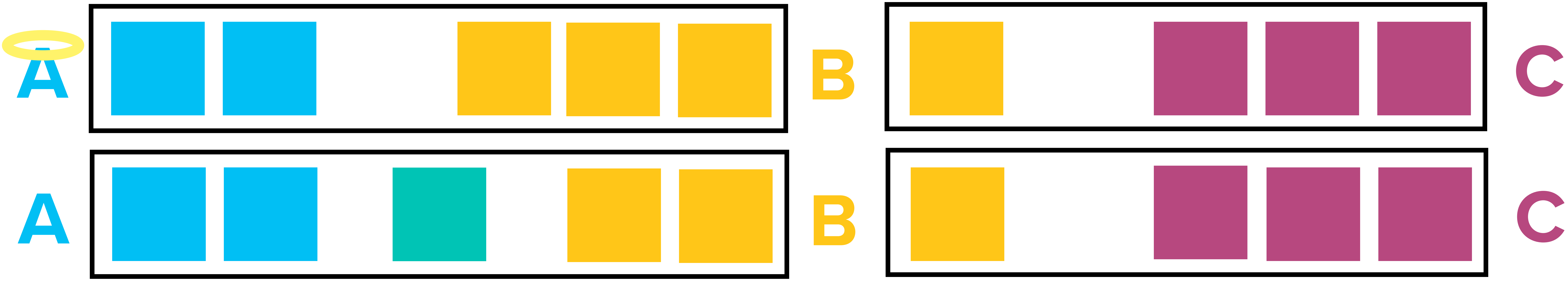
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



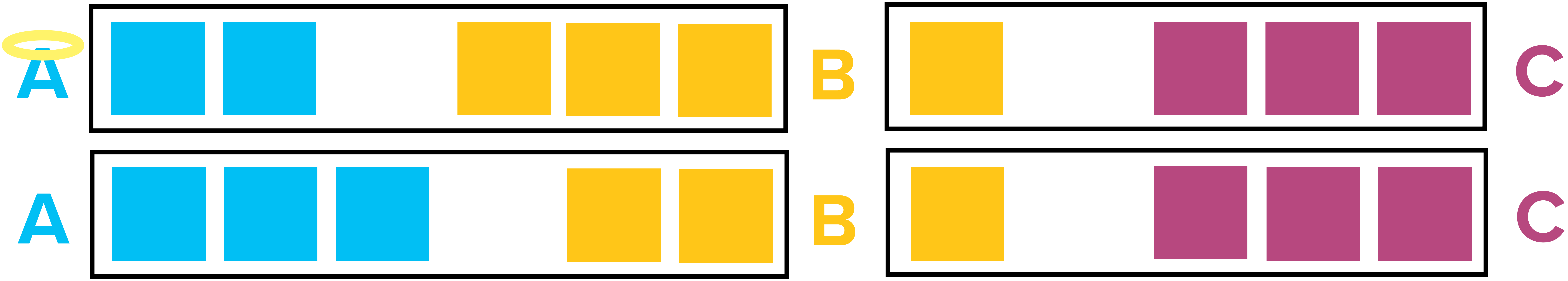
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



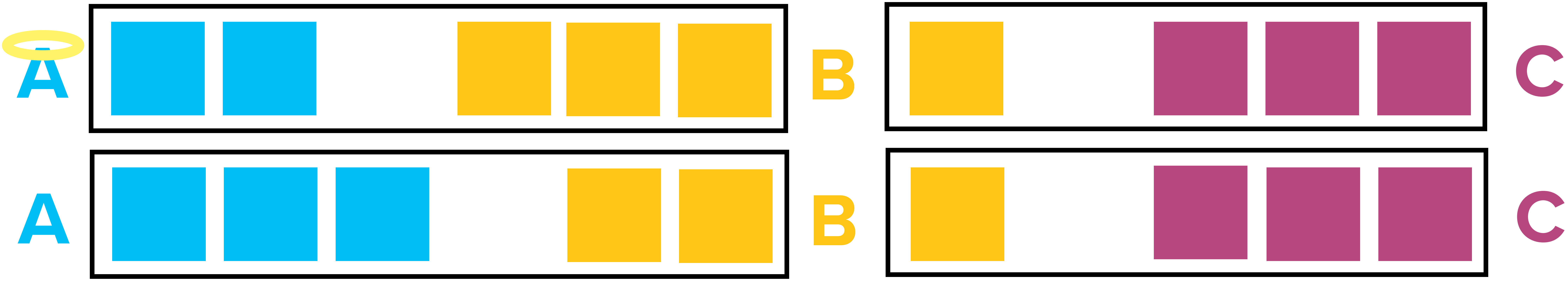
REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



REPUTATION OVERVIEW

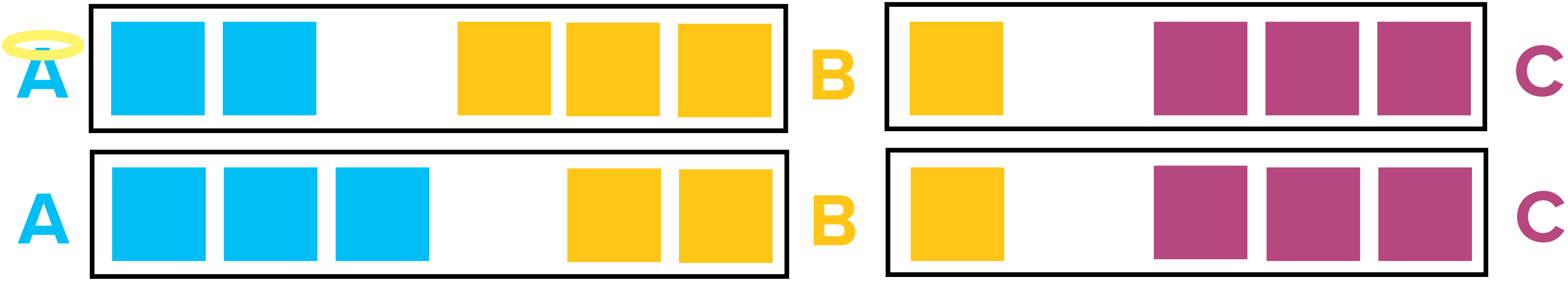
- Reputation is used to determine if to allocate resource



- Reputation is local

REPUTATION OVERVIEW

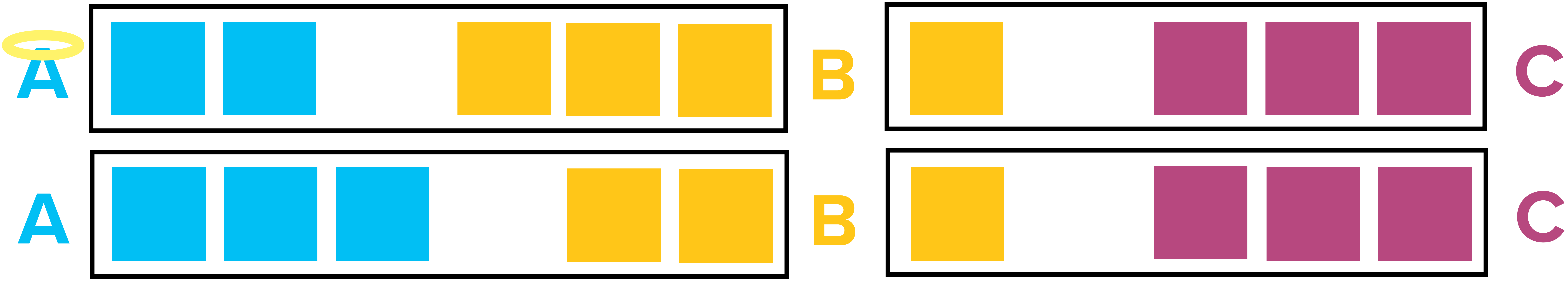
- Reputation is used to determine if to allocate resource



- Reputation is local
  - A node keeps track of the reputation of its direct neighbors

REPUTATION OVERVIEW

- Reputation is used to determine if to allocate resource



- Reputation is local
  - A node keeps track of the reputation of its direct neighbors
  - Alice and Charlie don’t need to agree on Bob’s reputation

BINARY LOCAL PEER REPUTATION

BINARY LOCAL PEER REPUTATION

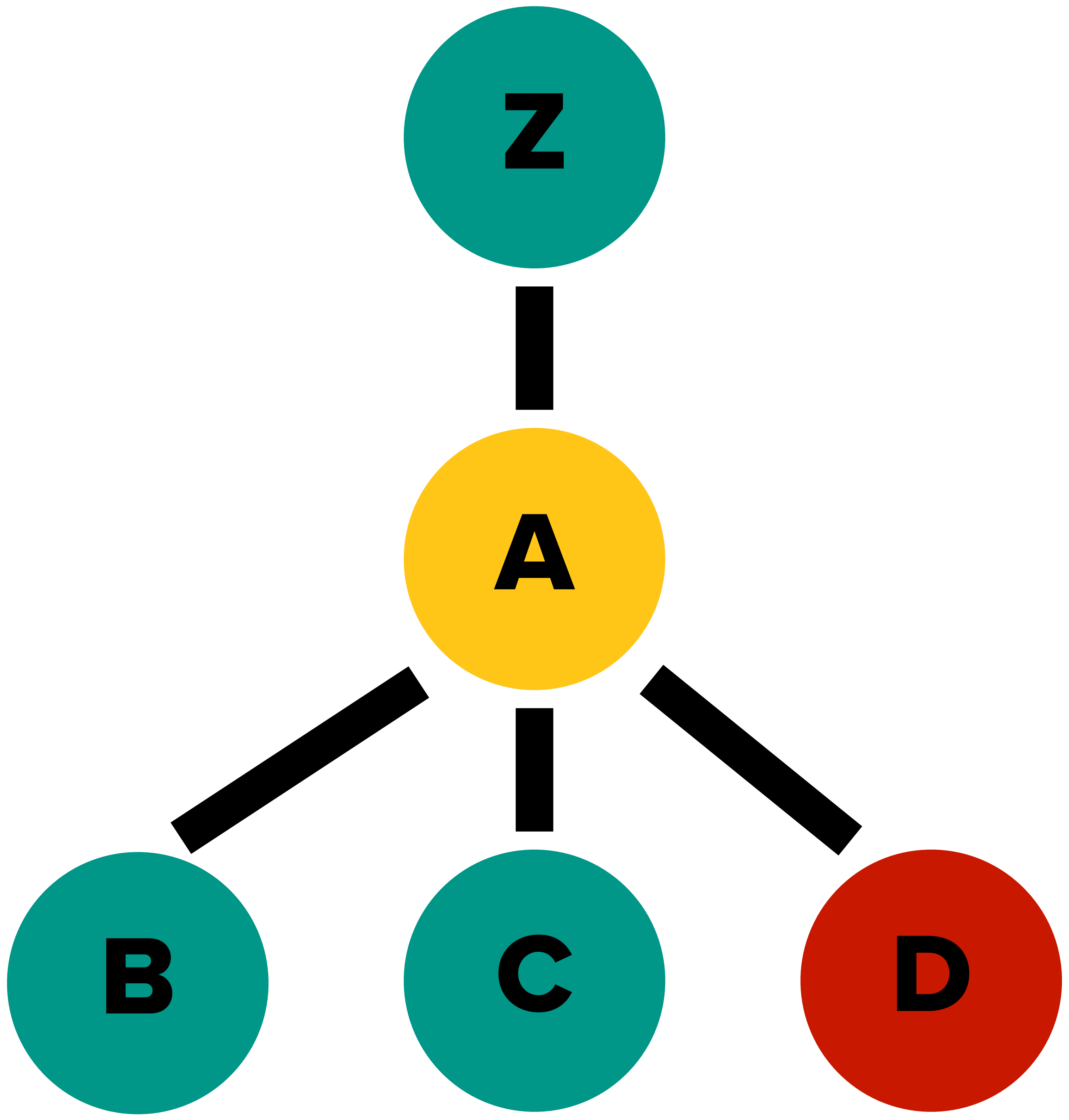
- Each node assigns a reputation to its neighbors

BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward

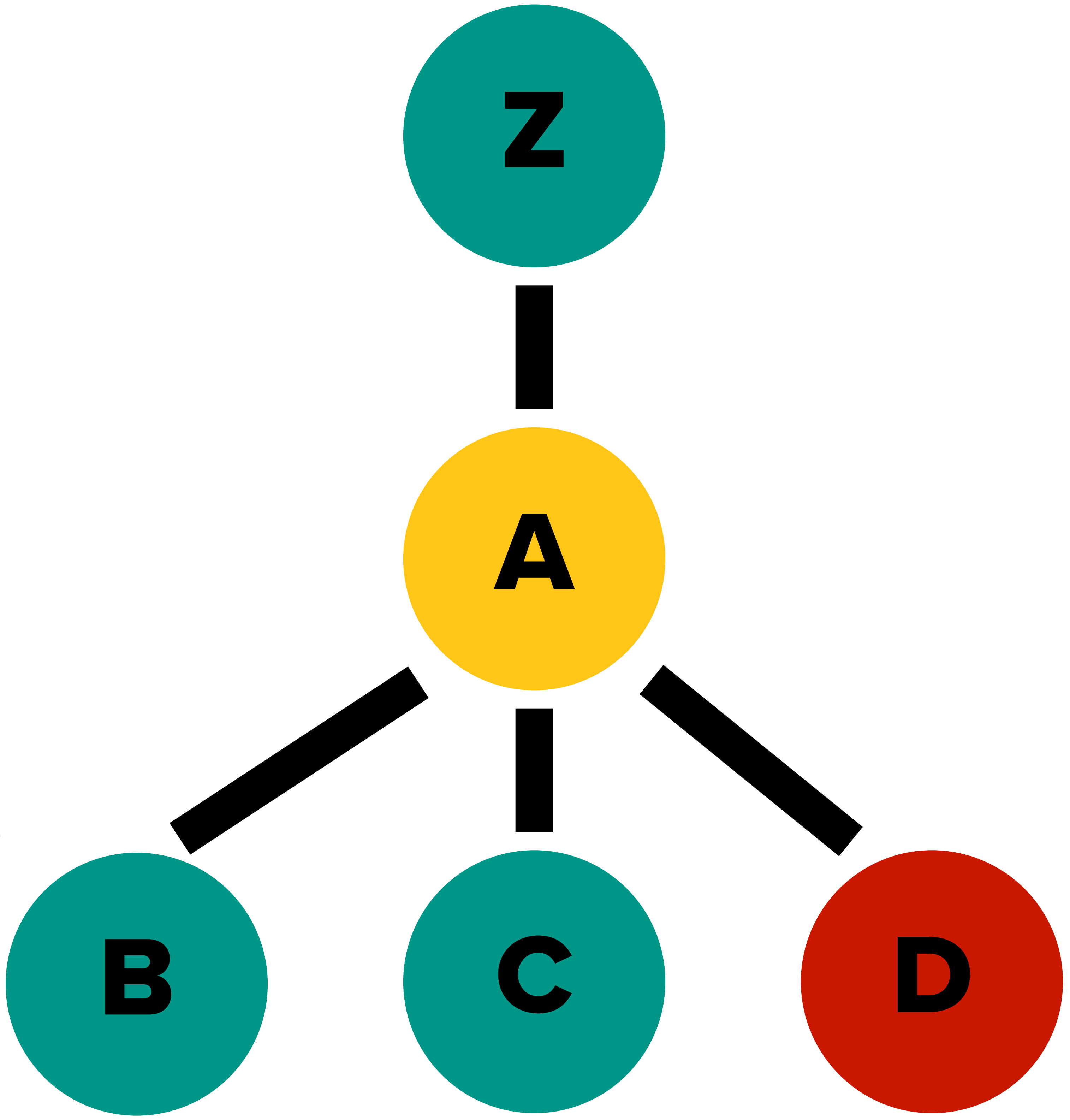
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward



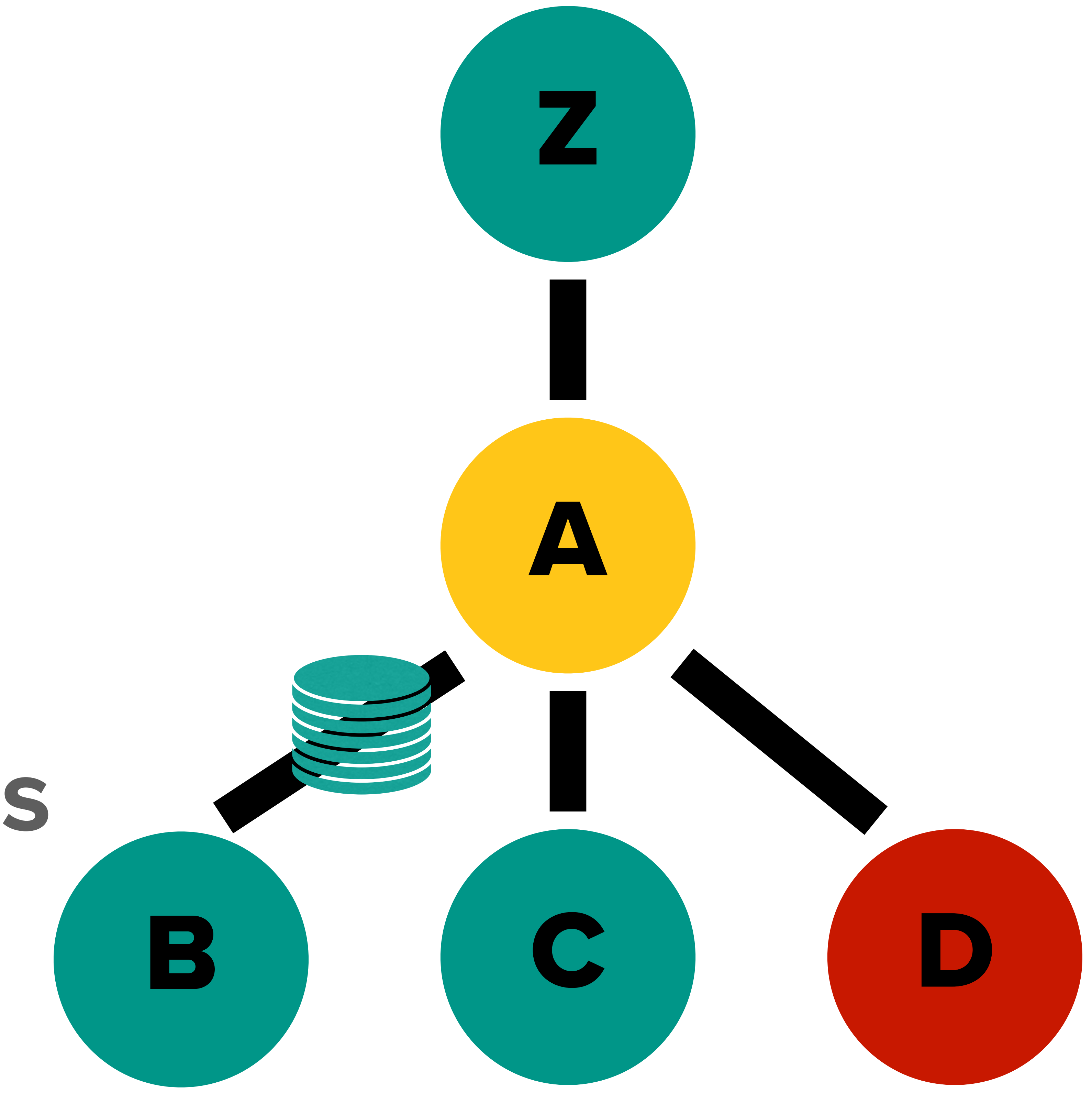
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



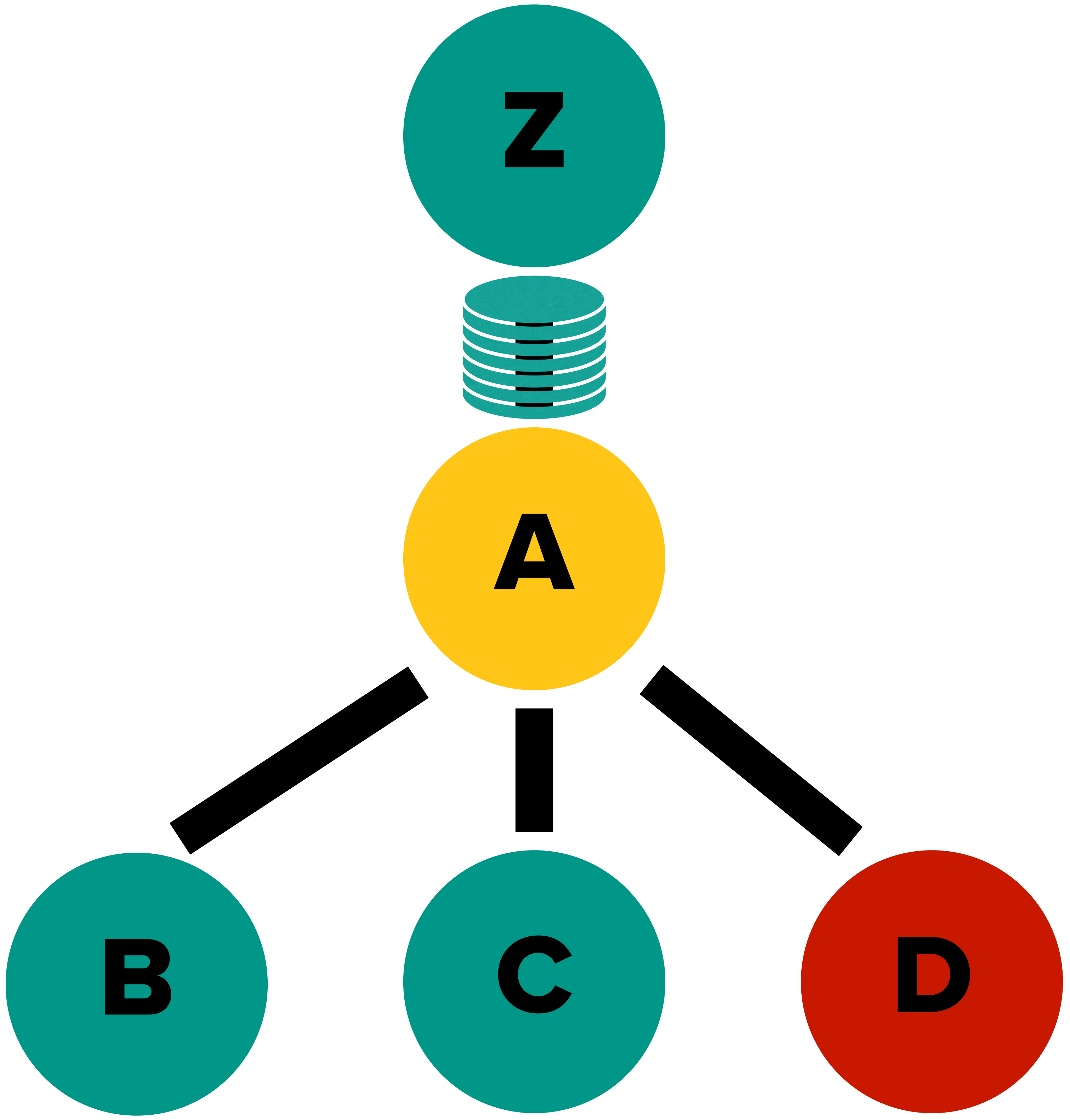
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



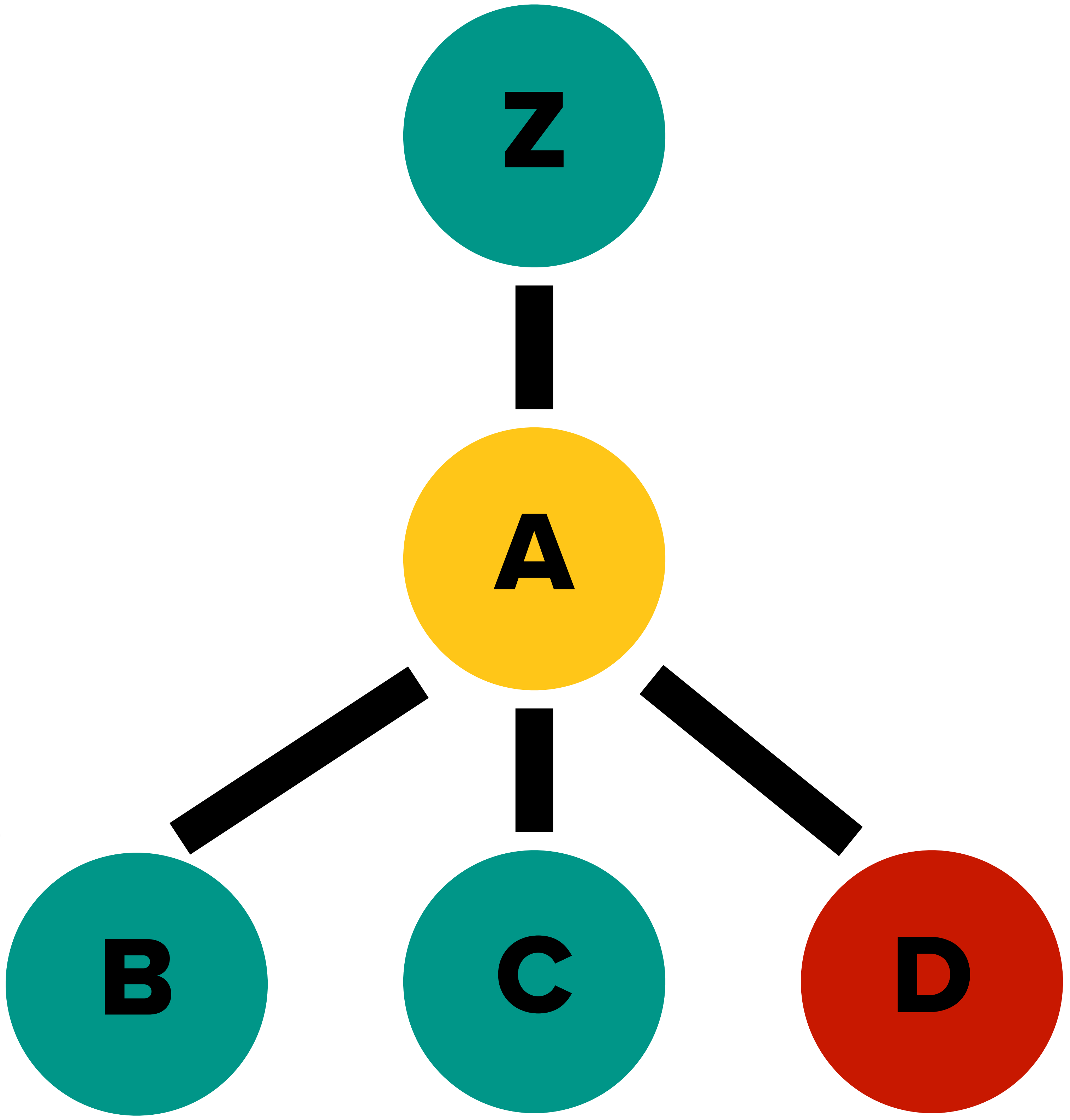
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



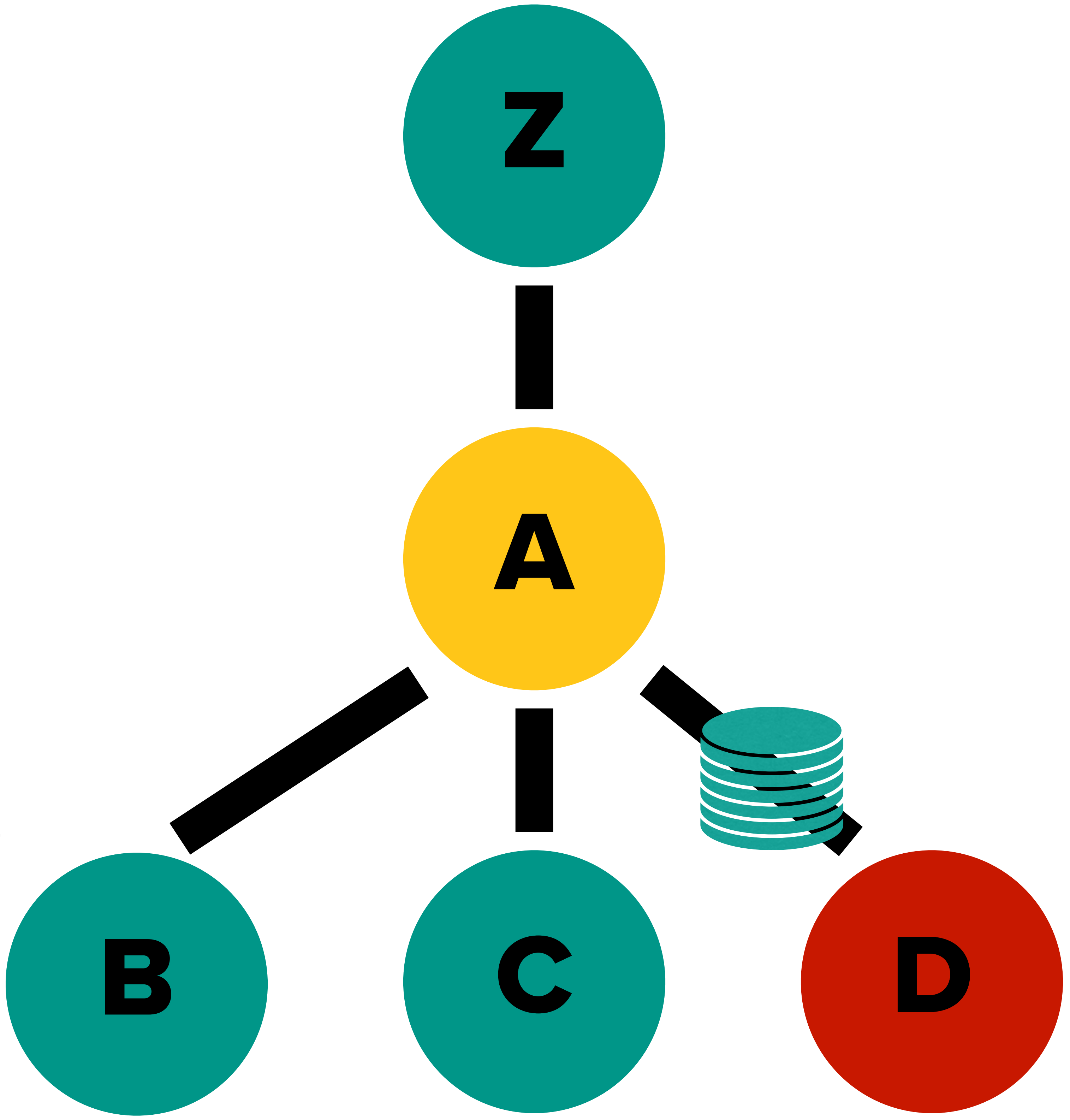
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



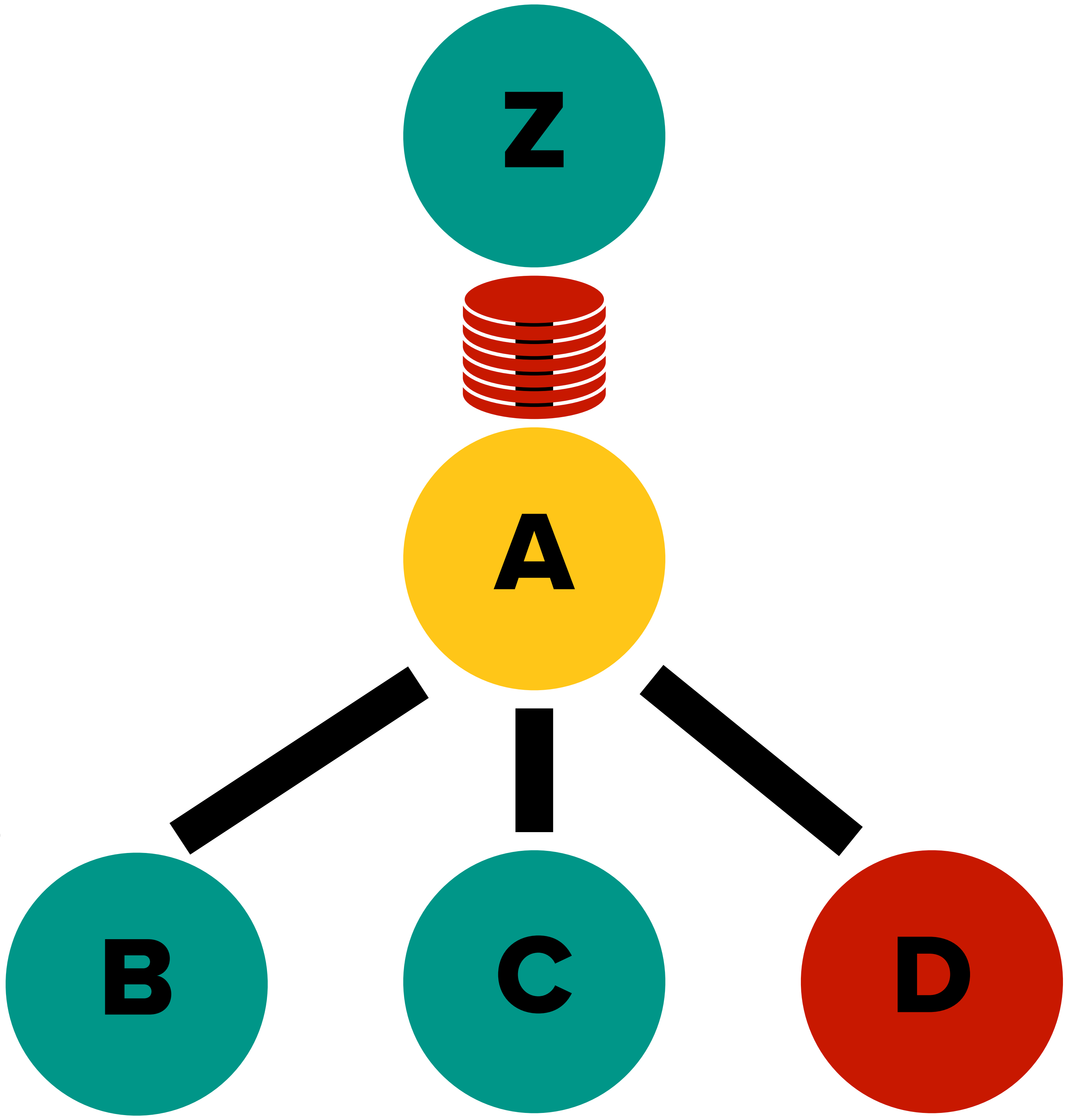
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



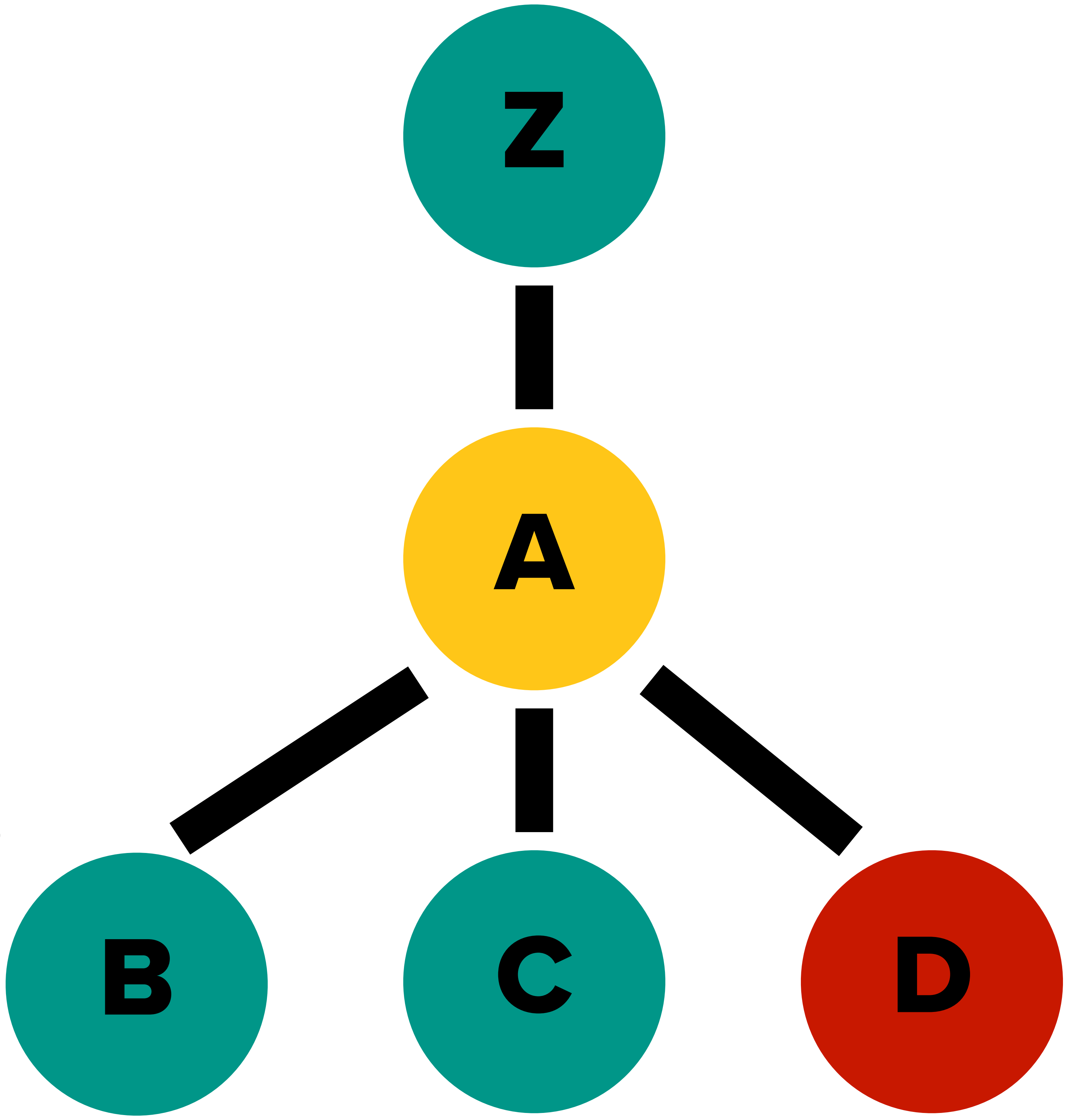
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



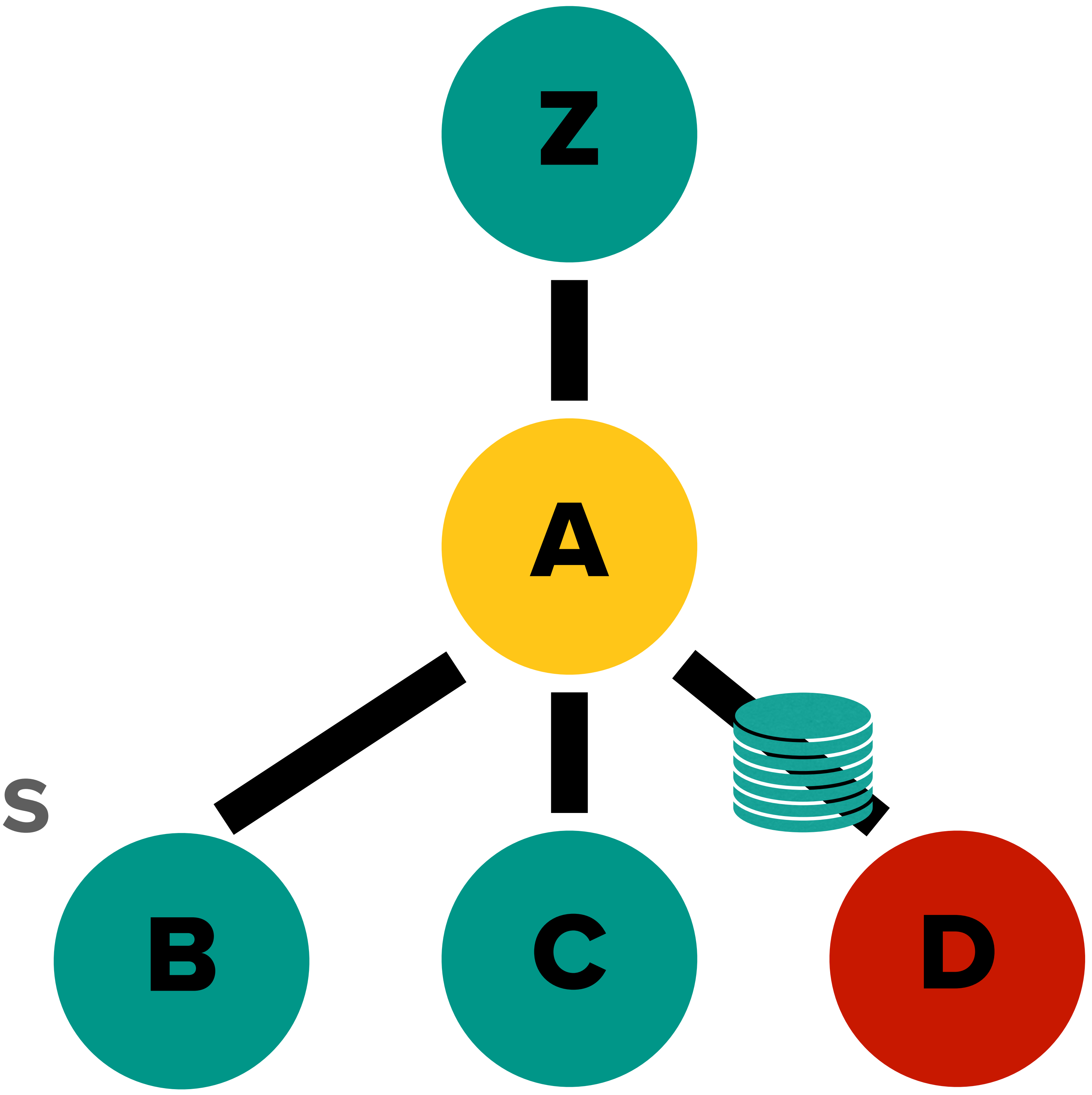
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



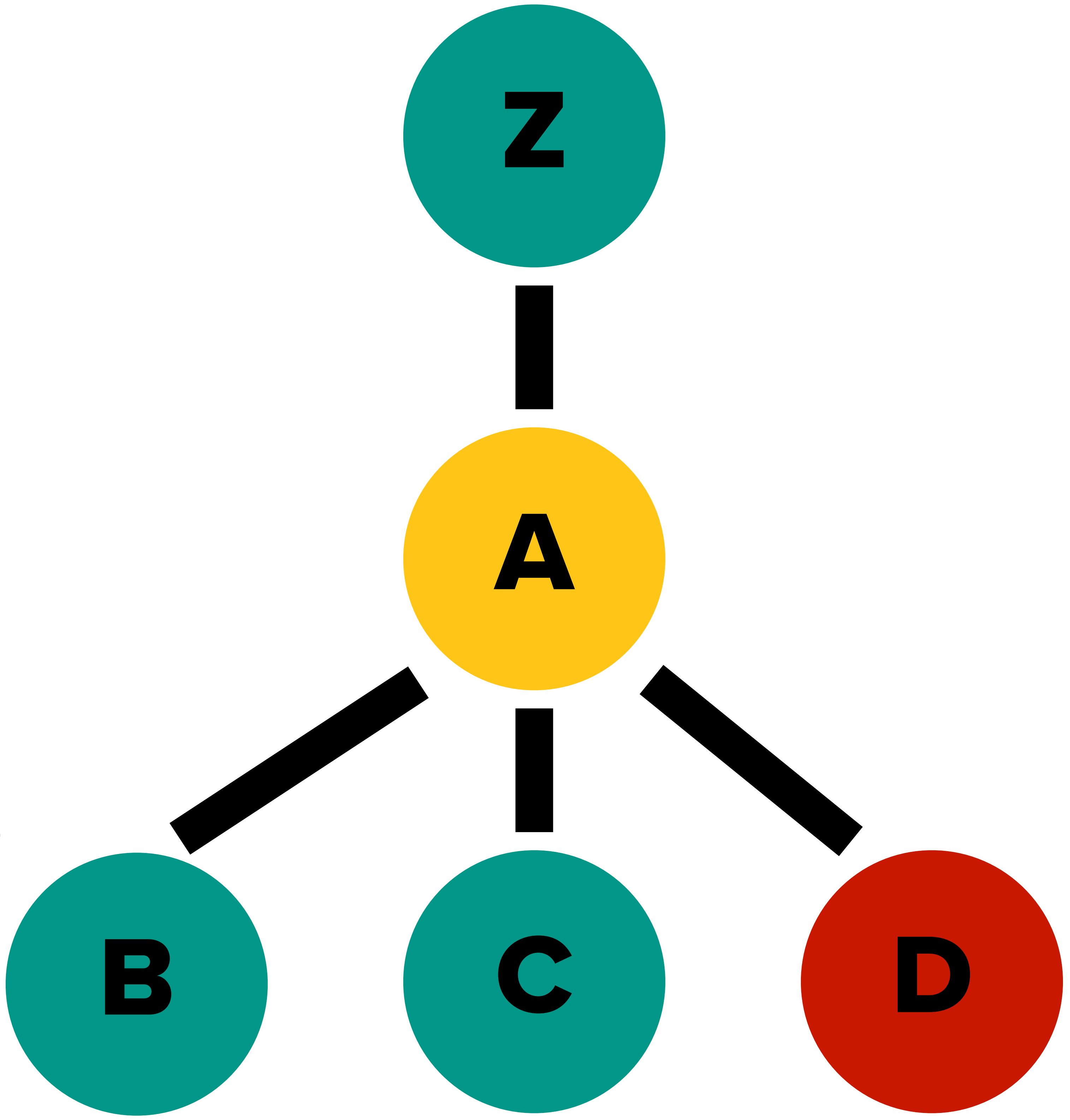
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



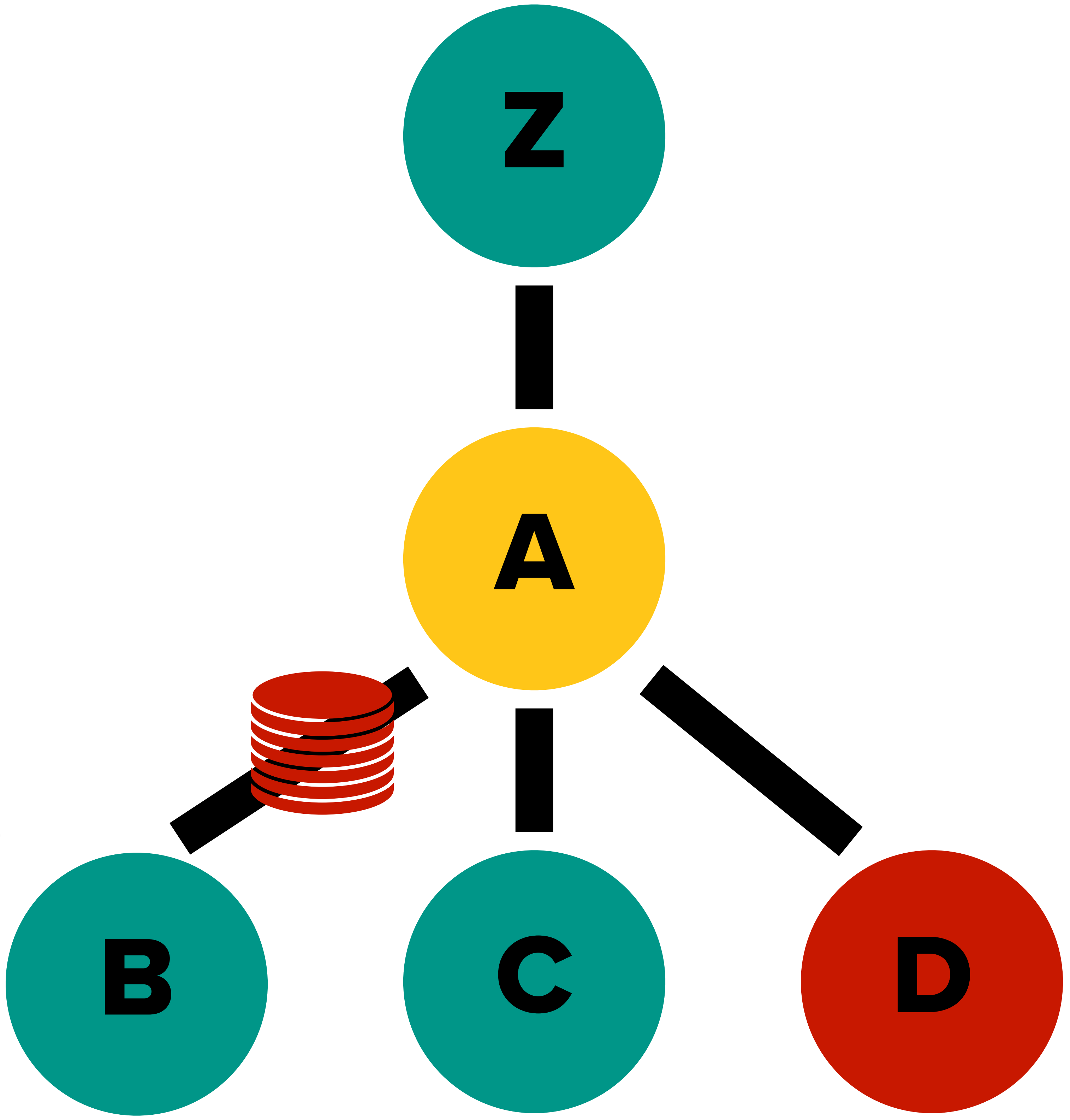
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



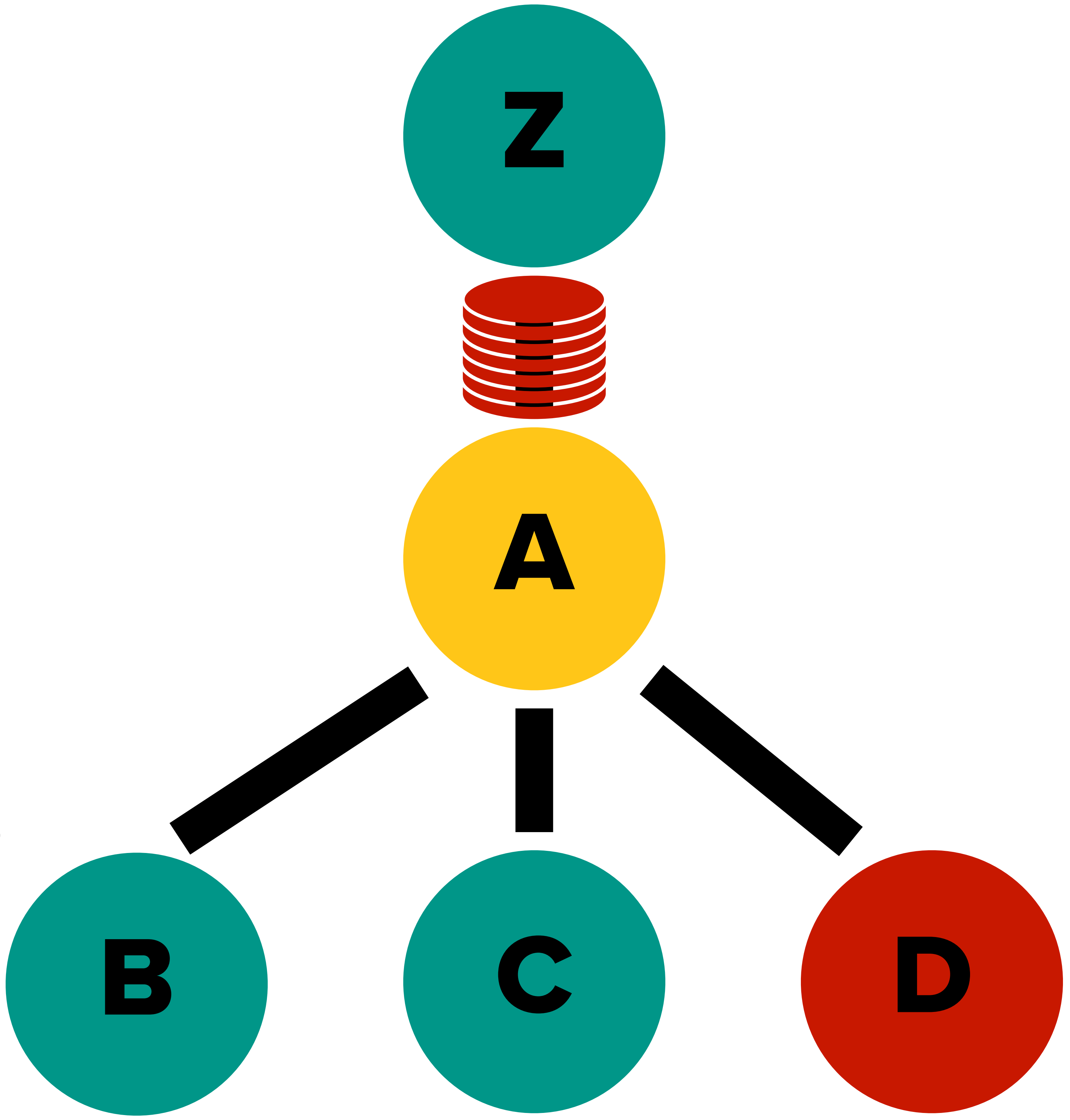
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



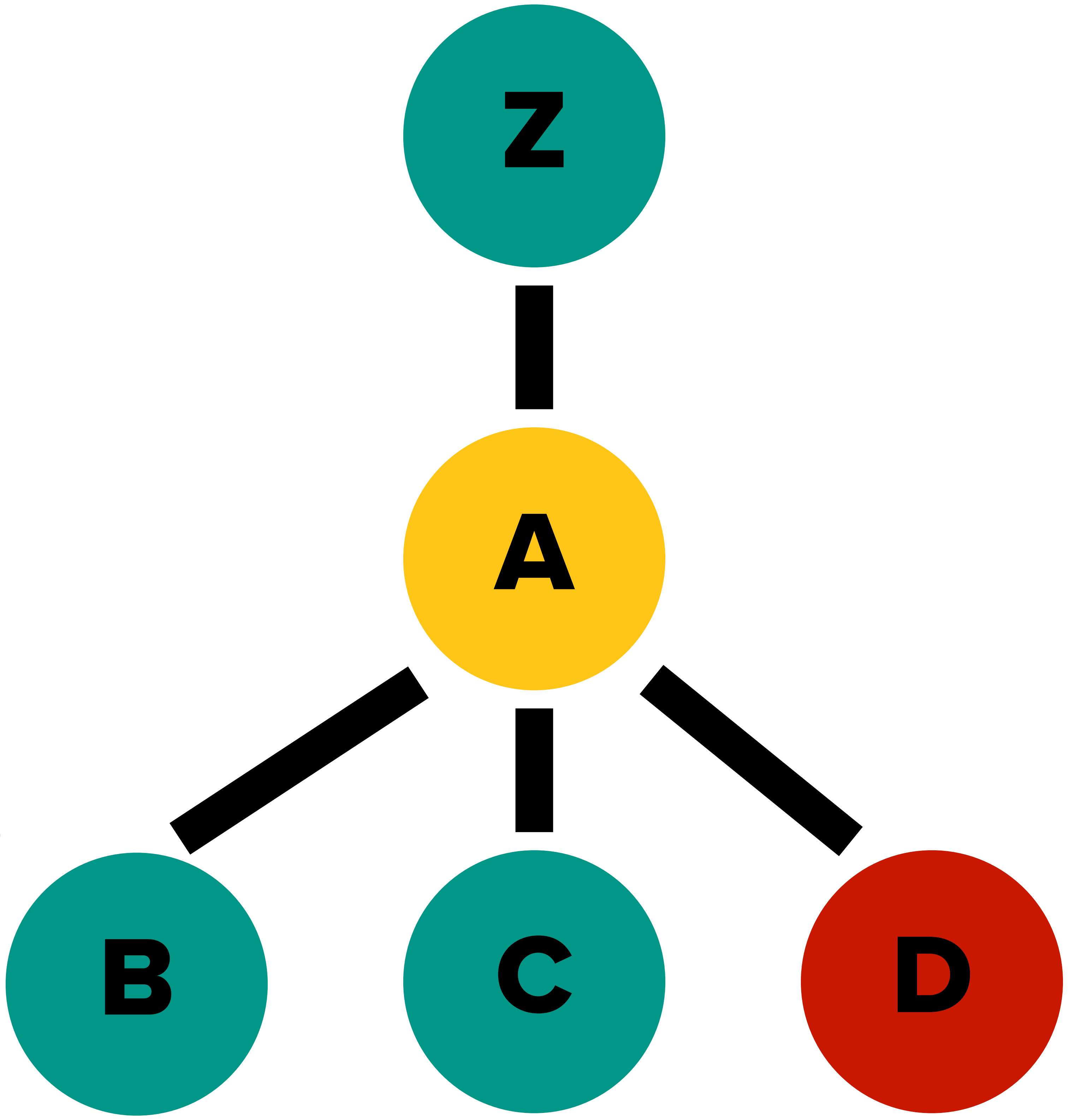
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



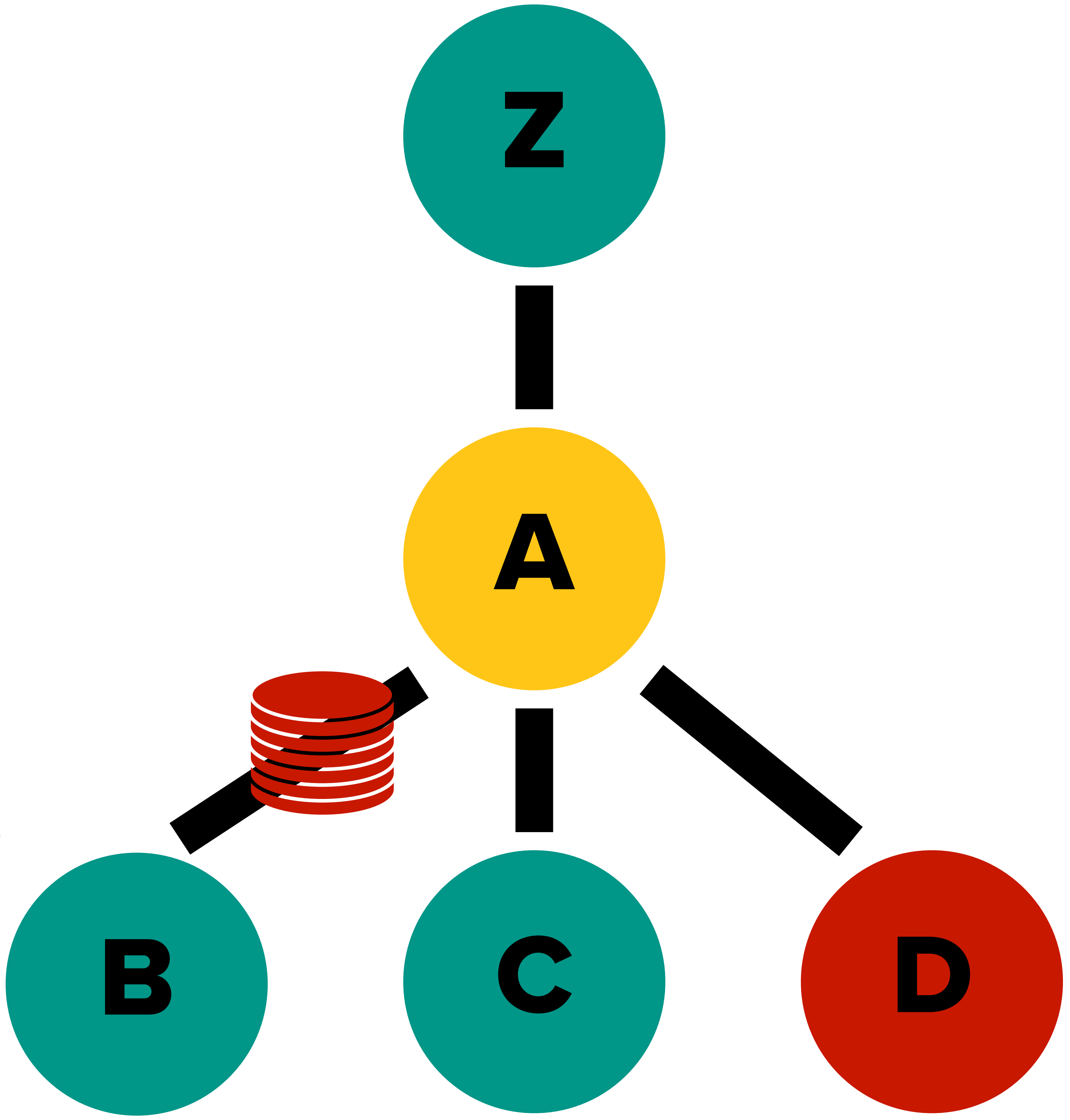
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



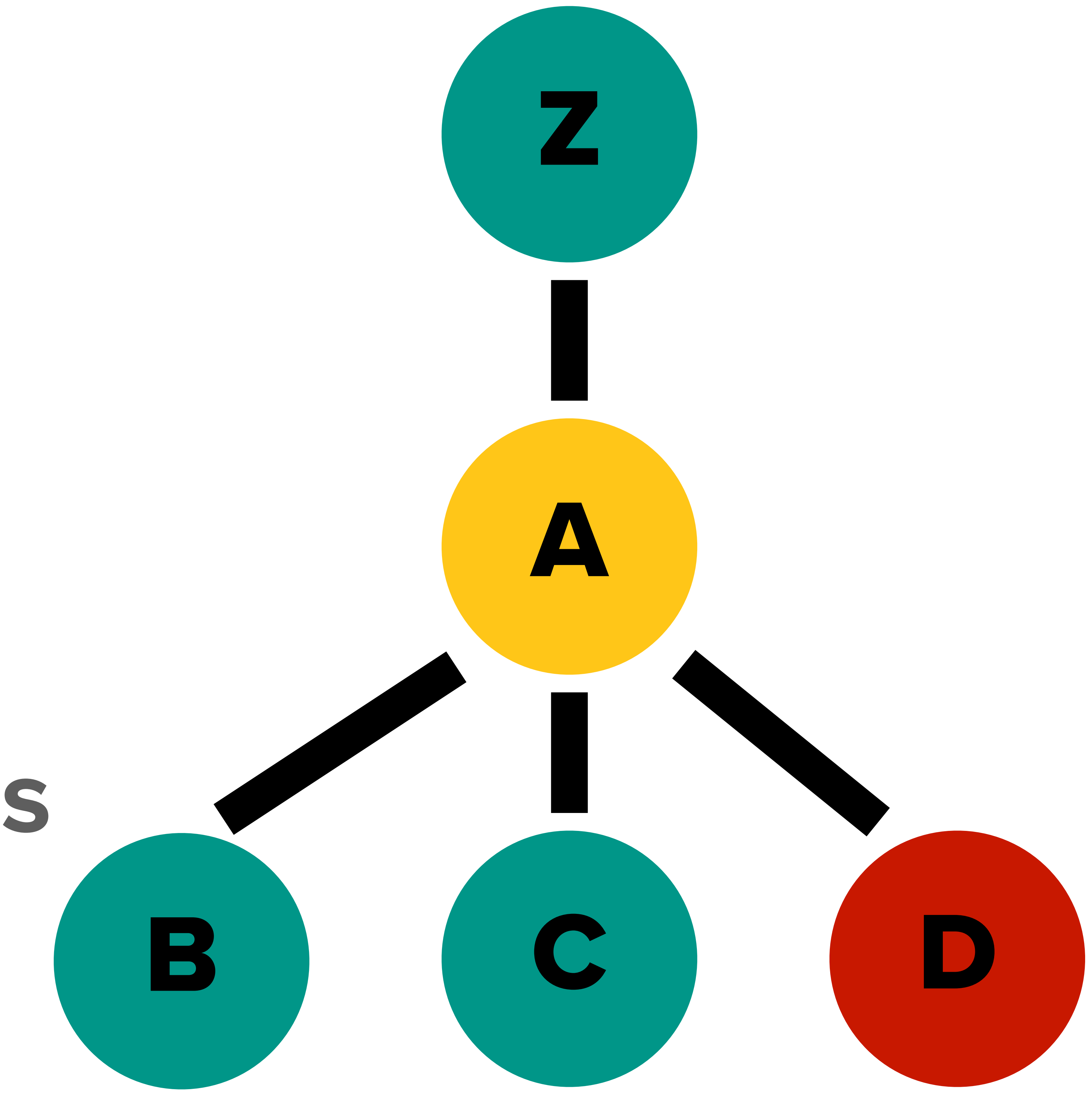
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts



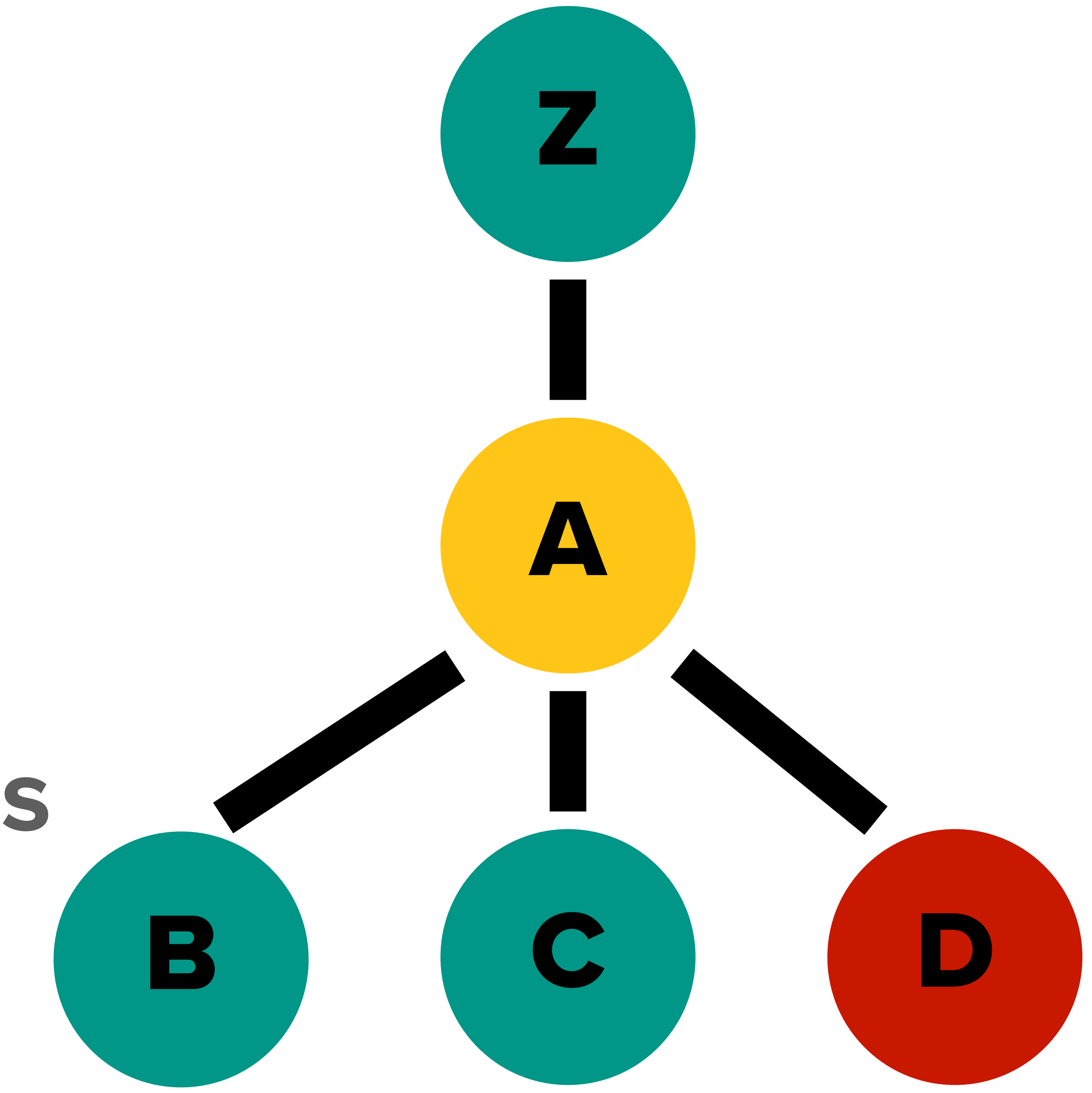
BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts

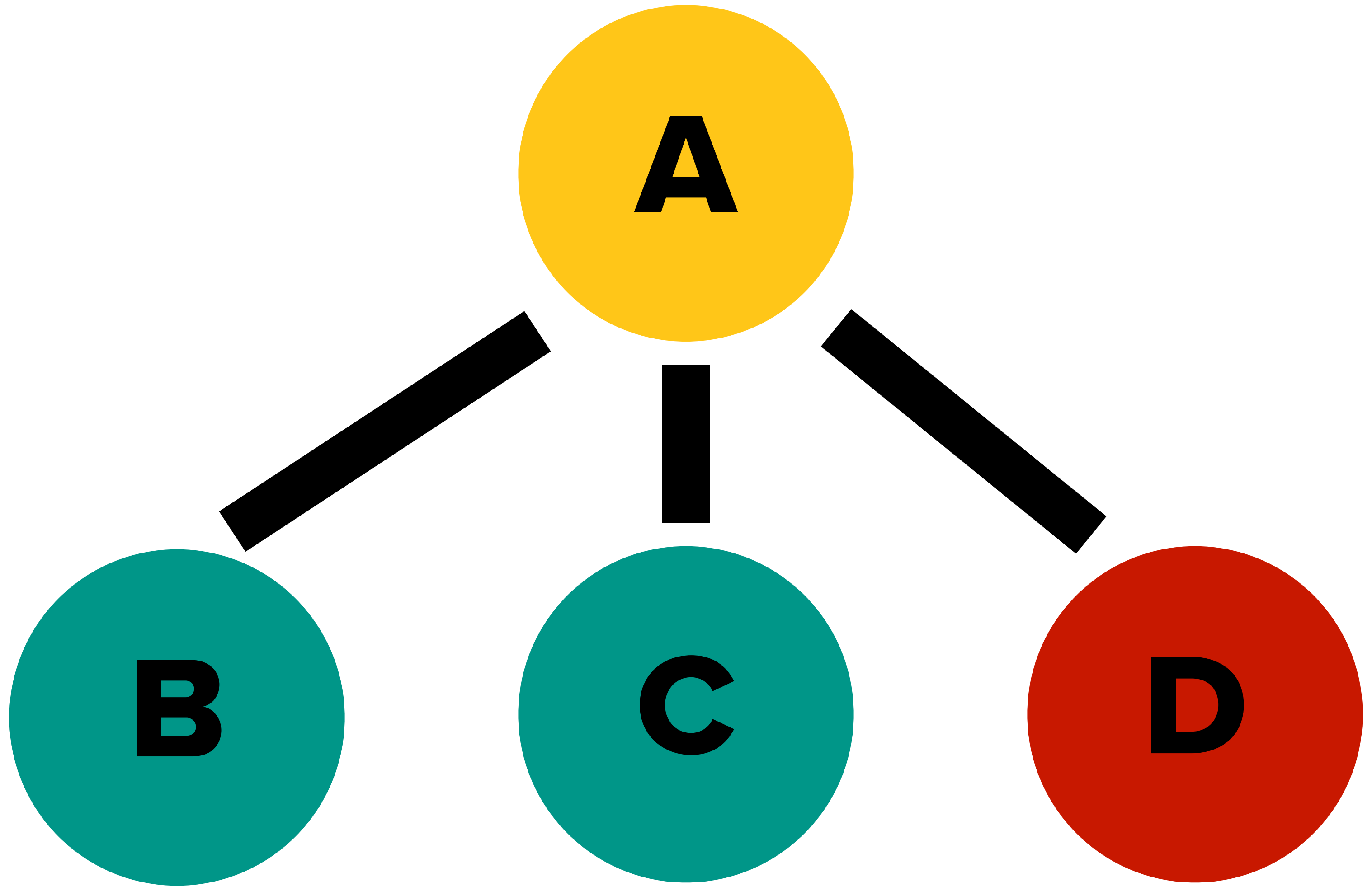


BINARY LOCAL PEER REPUTATION

- Each node assigns a reputation to its neighbors
- A neighbor can endorse a payment they forward
- Alice endorses a payment only if it comes from and endorsed by a neighbor she trusts
- In each channel, Alice allocates a limited liquidity and number of slots for un-endorsed

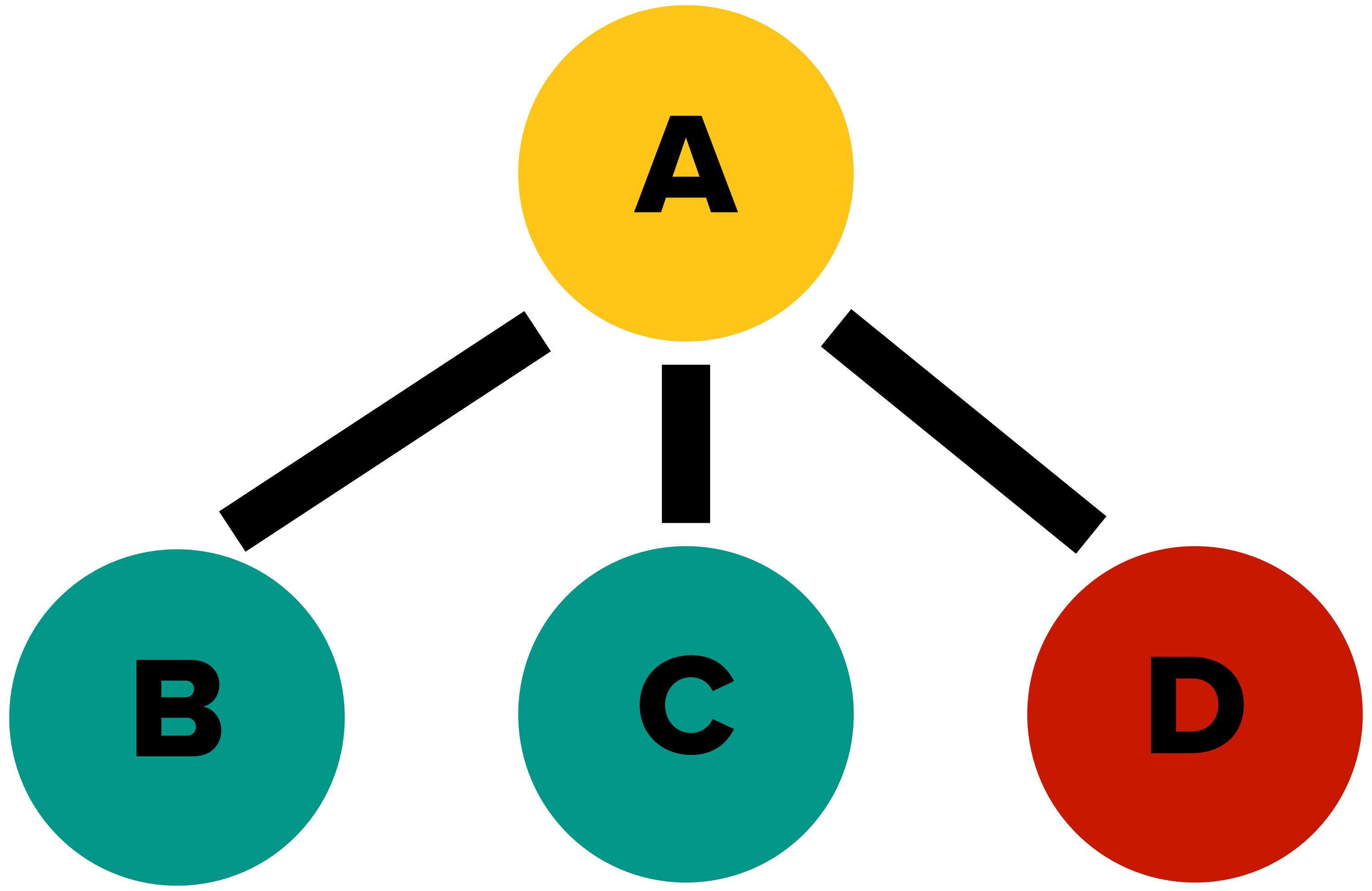


REPUTATION MANAGEMENT



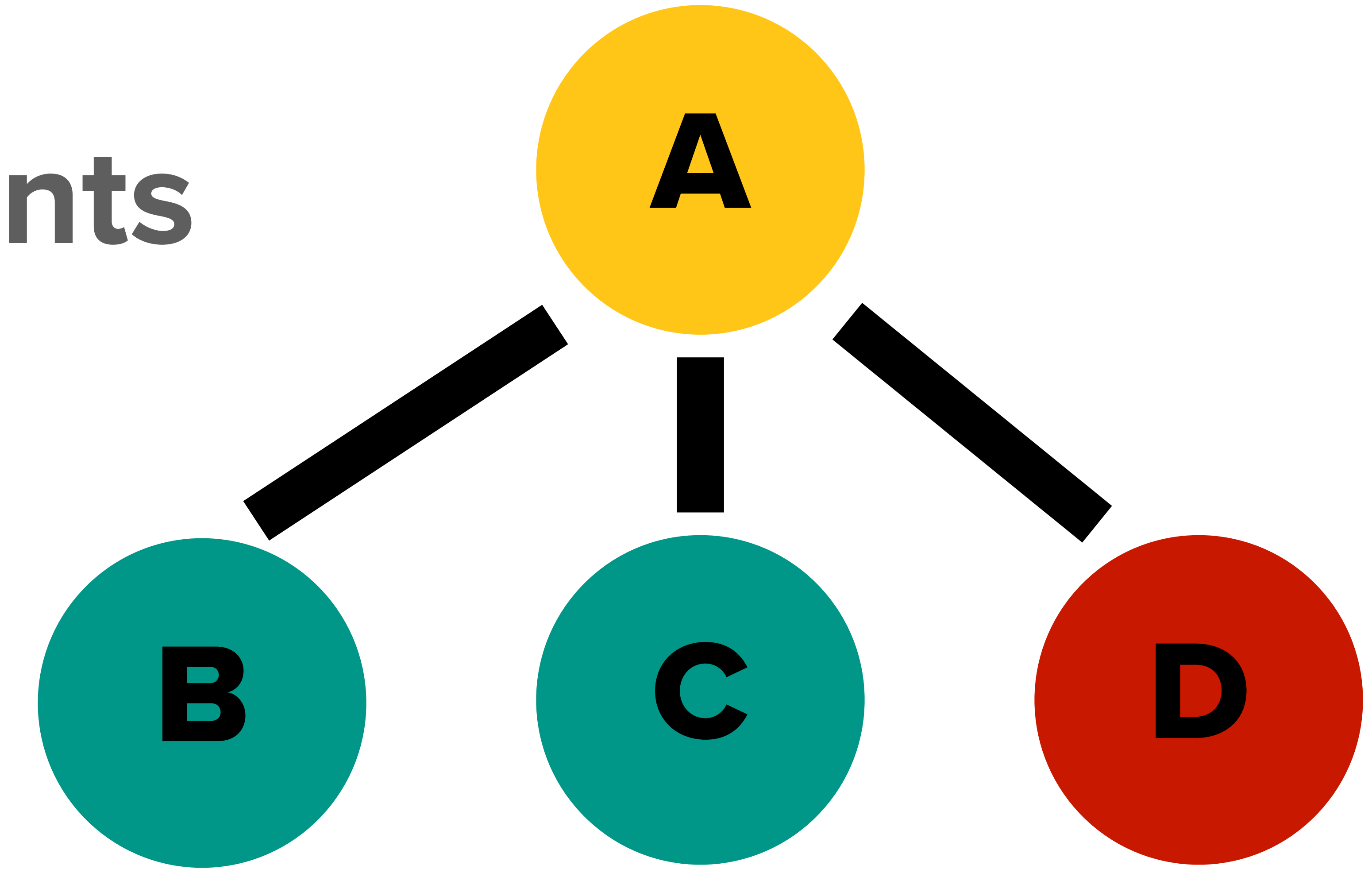
REPUTATION MANAGEMENT

- Reputation can be high or low



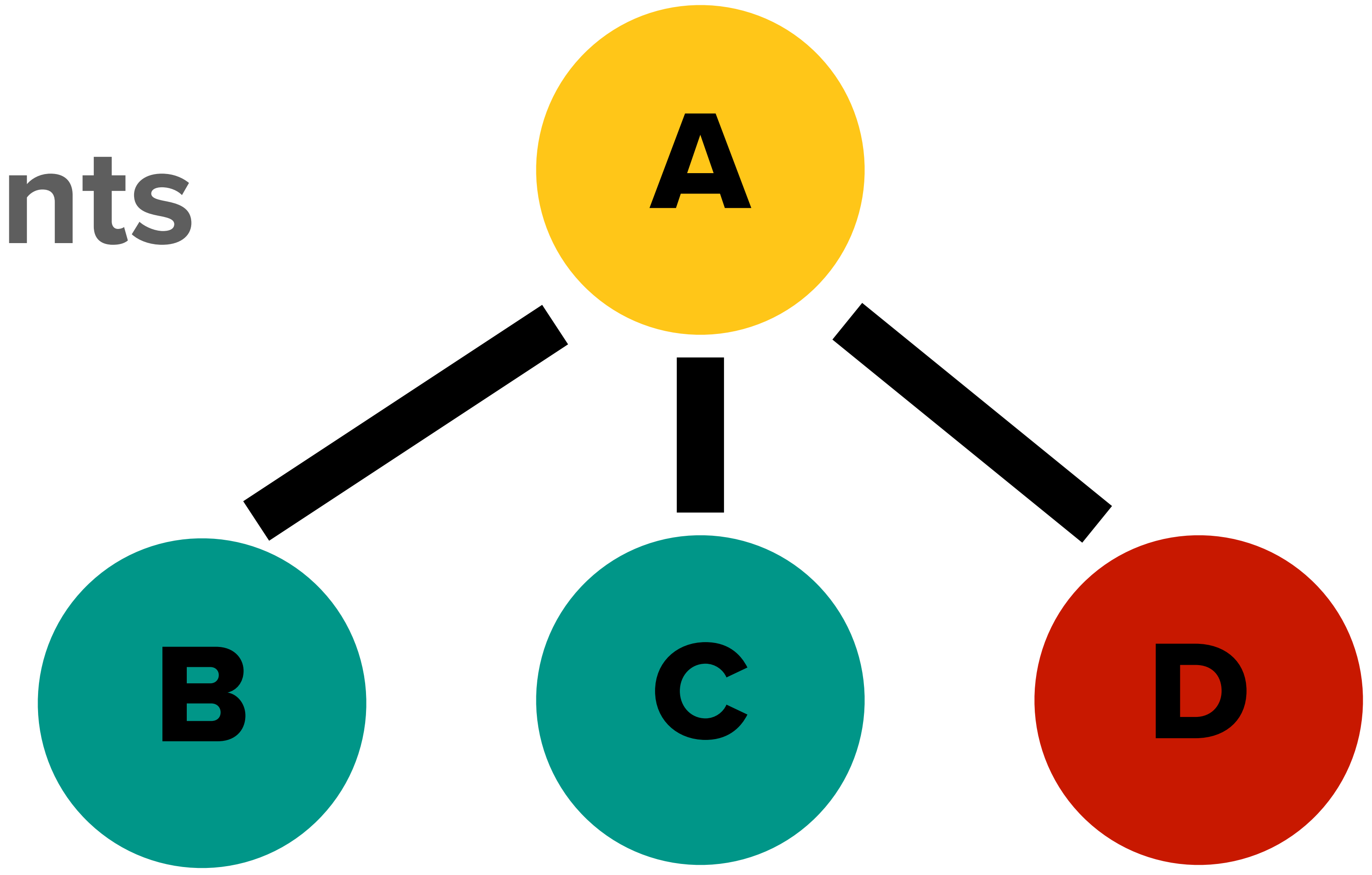
REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that



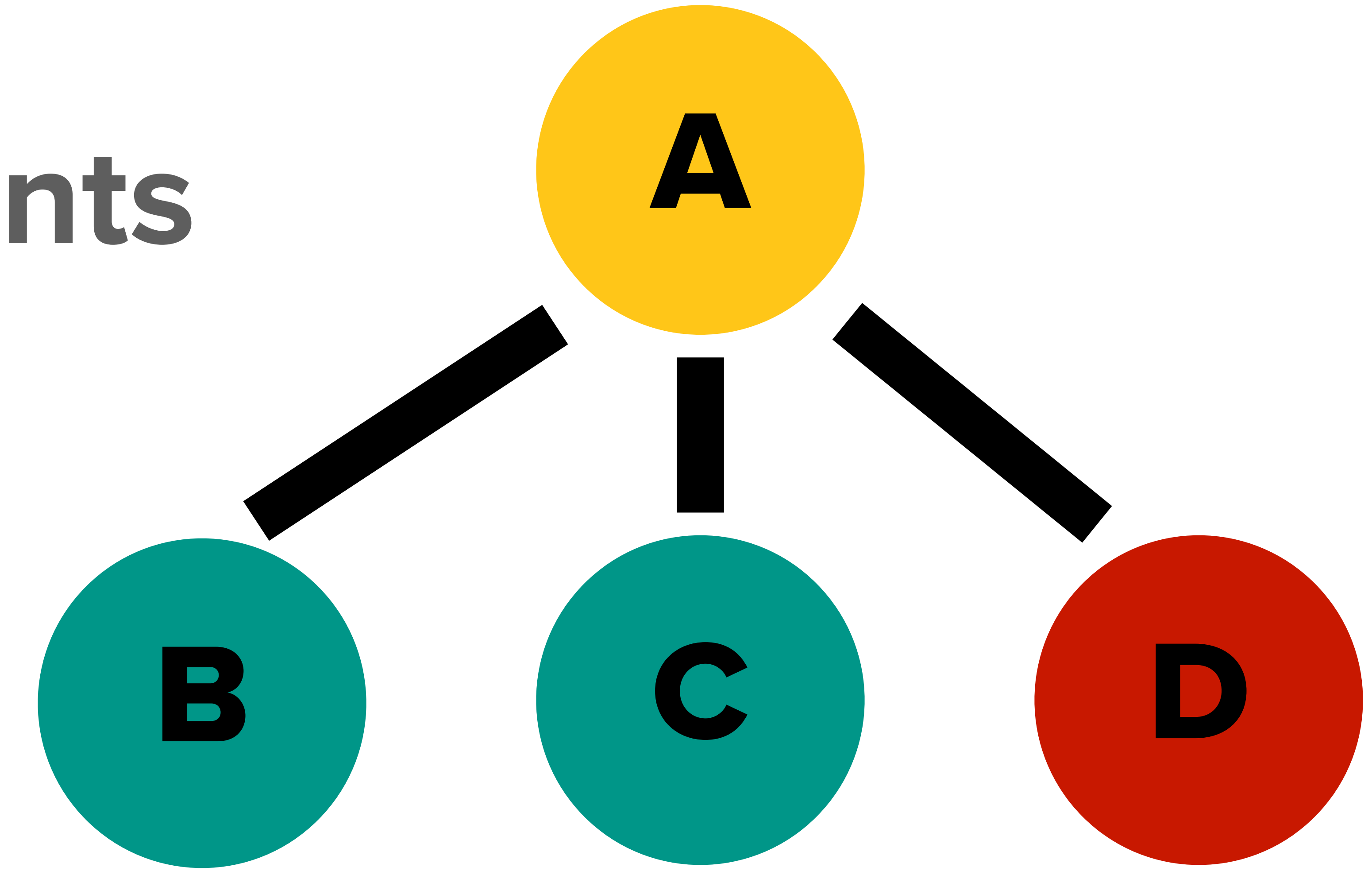
REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that
  - Succeed quickly



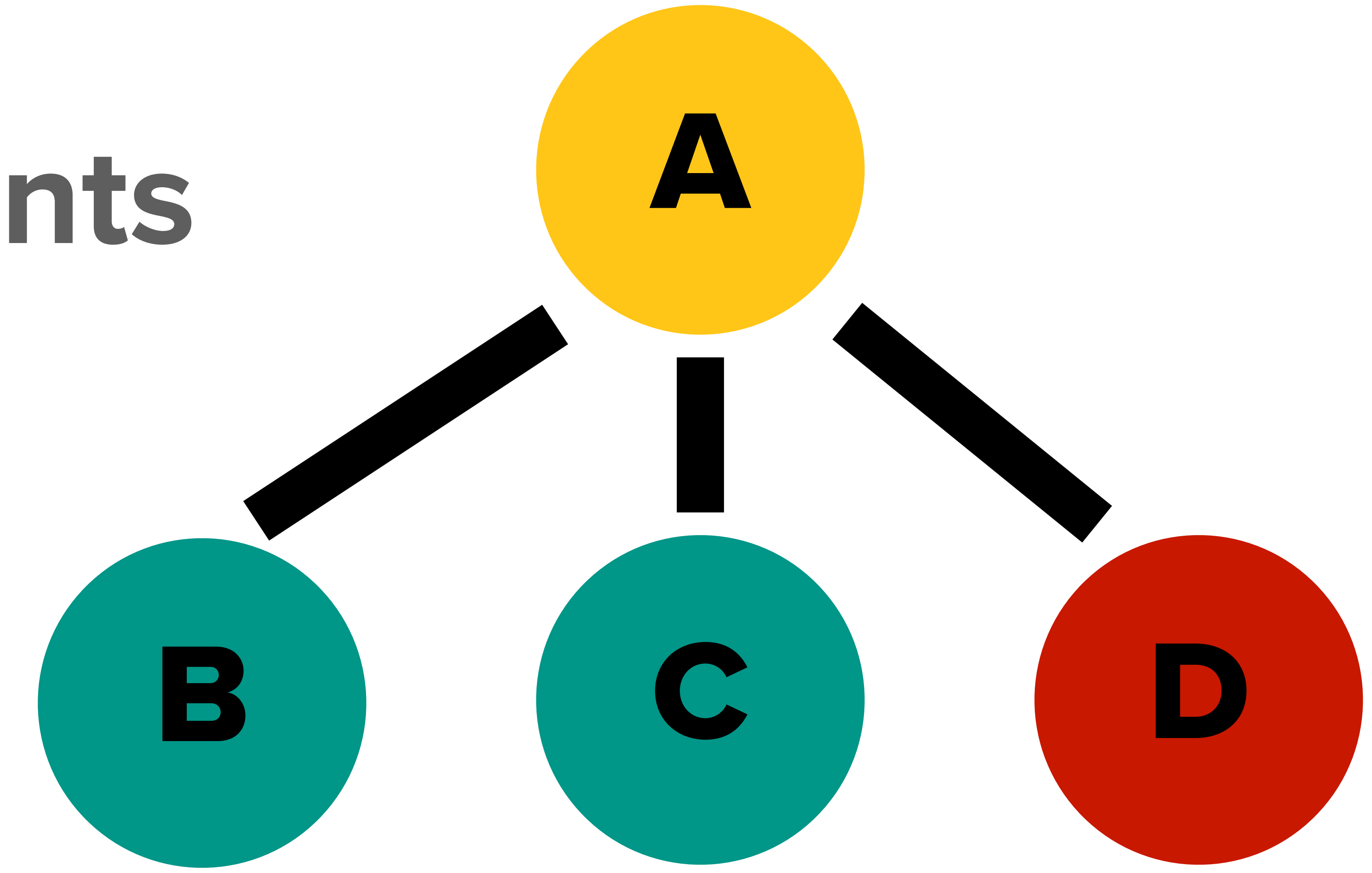
REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that
  - Succeed quickly
  - Pay enough fees



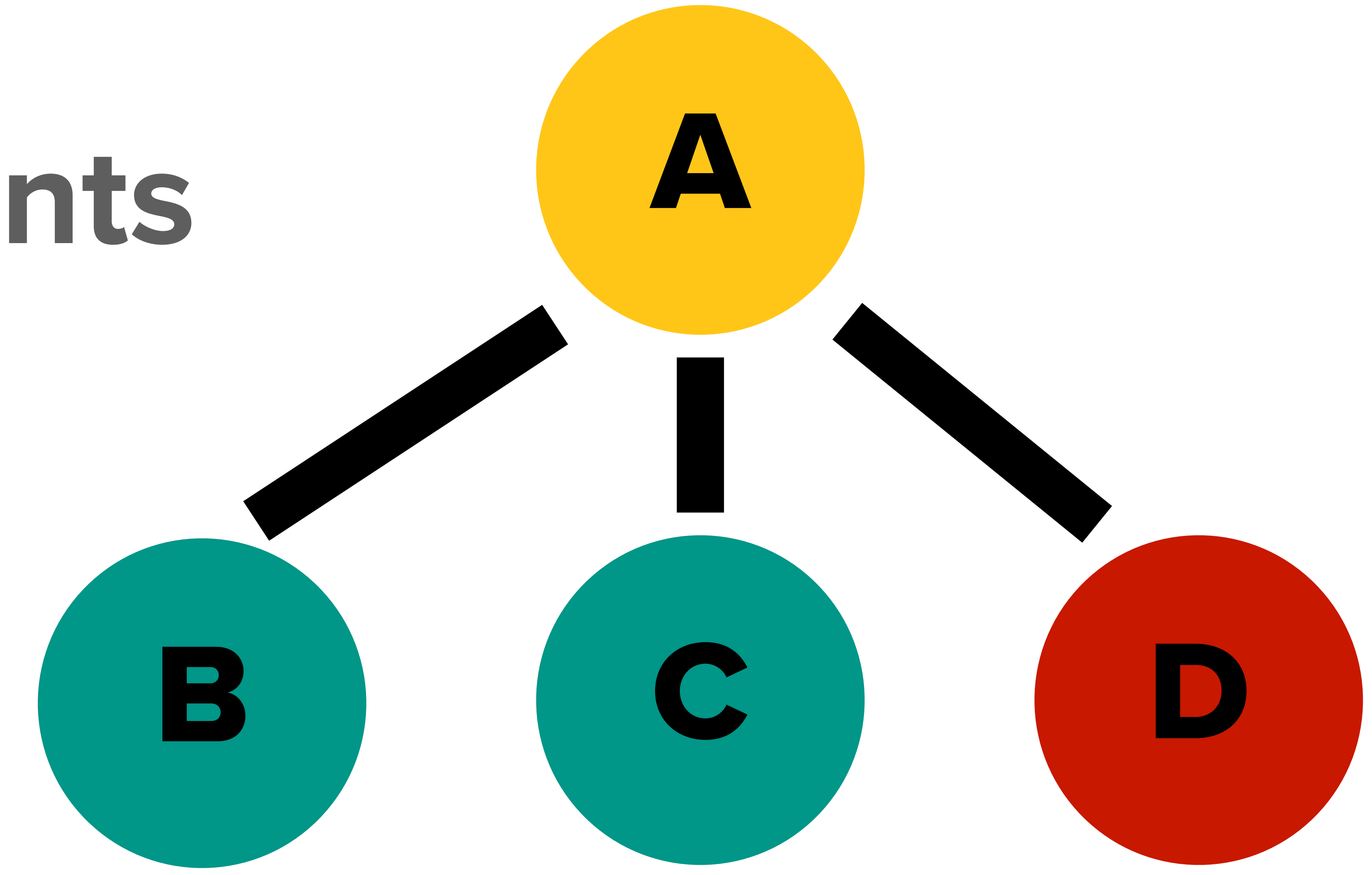
REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that
  - Succeed quickly
  - Pay enough fees
- Reputation is lost by forwarding payments that are



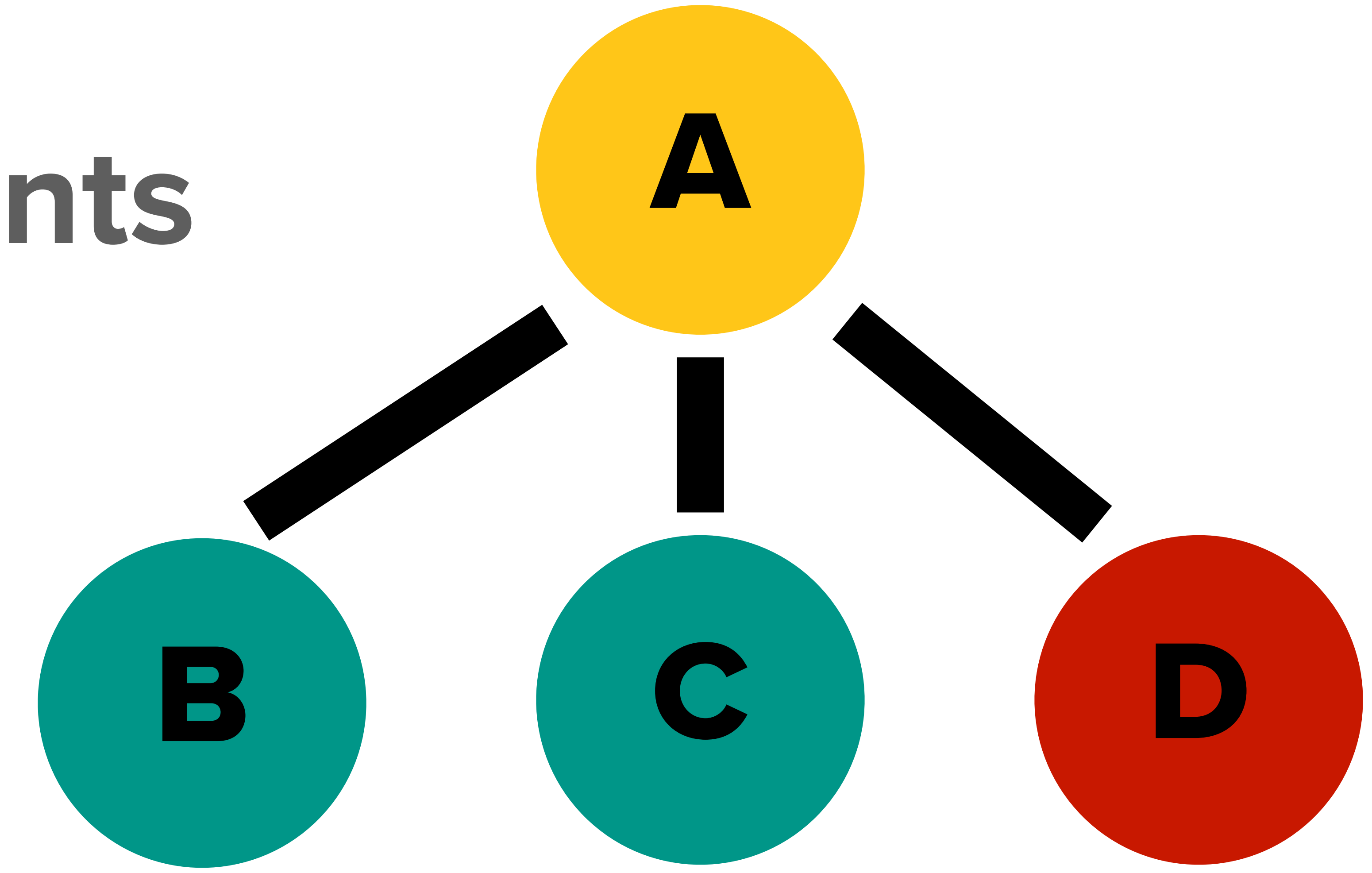
REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that
  - Succeed quickly
  - Pay enough fees
- Reputation is lost by forwarding payments that are
  - Clearly jams



REPUTATION MANAGEMENT

- Reputation can be high or low
- Reputation is gained by forwarding payments that
  - Succeed quickly
  - Pay enough fees
- Reputation is lost by forwarding payments that are
  - Clearly jams
  - Not paying enough fees



DOWN SIDE OF REPUTATION

DOWN SIDE OF REPUTATION

- Edge cases always exist

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

DOWN SIDE OF REPUTATION

- Edge cases always exist
    - Need to resolve in 10 seconds?

      Resolve in 9 seconds and resend

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

    Resolve in 9 seconds and resend
  - Need a 50% success history?

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

    Resolve in 9 seconds and resend
  - Need a 50% success history?

    Open several channels

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

    Resolve in 9 seconds and resend
  - Need a 50% success history?

    Open several channels
- These edge cases are what we call “quick jamming”

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

    Resolve in 9 seconds and resend
  - Need a 50% success history?

    Open several channels
- These edge cases are what we call “quick jamming”
  - Difficult to detect

DOWN SIDE OF REPUTATION

- Edge cases always exist
  - Need to resolve in 10 seconds?

    Resolve in 9 seconds and resend
  - Need a 50% success history?

    Open several channels
- These edge cases are what we call “quick jamming”
  - Difficult to detect
  - Can rarely happen to honest users

UNCONDITIONAL FEE

UNCONDITIONAL FEE

- Currently, fees are charged only for successful payments

UNCONDITIONAL FEE

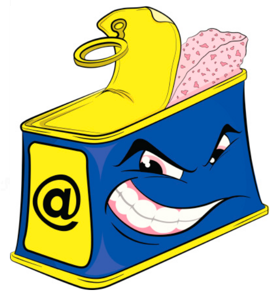
- Currently, fees are charged only for successful payments
- This allows:

UNCONDITIONAL FEE

- Currently, fees are charged only for successful payments
- This allows:
    - Jamming

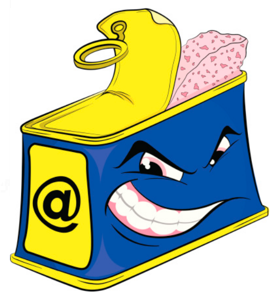
UNCONDITIONAL FEE

- Currently, fees are charged only for successful payments
- This allows:
  - Jamming
  - Spamming



UNCONDITIONAL FEE

- Currently, fees are charged only for **successful** payments
- This allows:
  - Jamming
  - Spamming
  - Probing



UNCONDITIONAL FEE

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

- Jamming can use one of the scarce resources

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

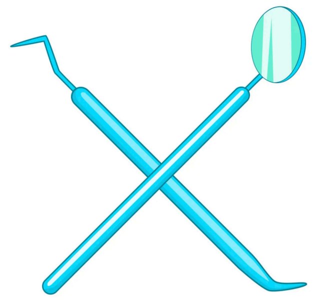
Base Fee + *Proportional Fee*

- Jamming can use one of the scarce resources
  - Liquidity

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

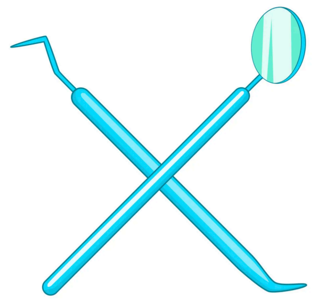
- Jamming can use one of the scarce resources
  - Liquidity 

UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

- Jamming can use one of the scarce resources
  - Liquidity
  - Slots

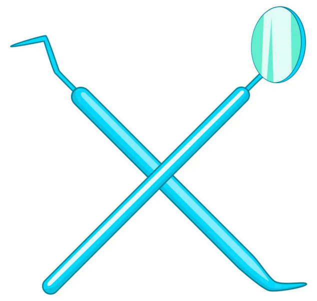


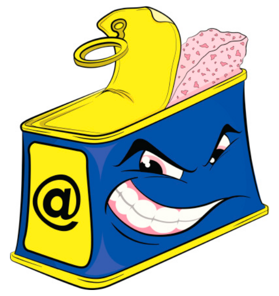
UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

  Base Fee + *Proportional Fee*

- Jamming can use one of the scarce resources
  - Liquidity
  - Slots



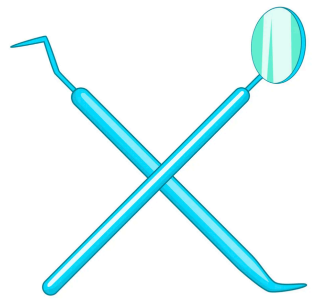


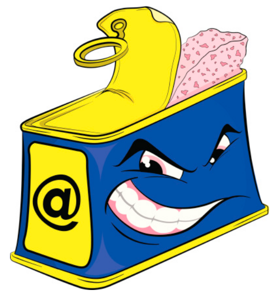
UNCONDITIONAL FEE

- Unconditional fee helps mitigate various attacks.
- The structure of the fee

Base Fee + *Proportional Fee*

- Jamming can use one of the scarce resources
  - Liquidity
  - Slots



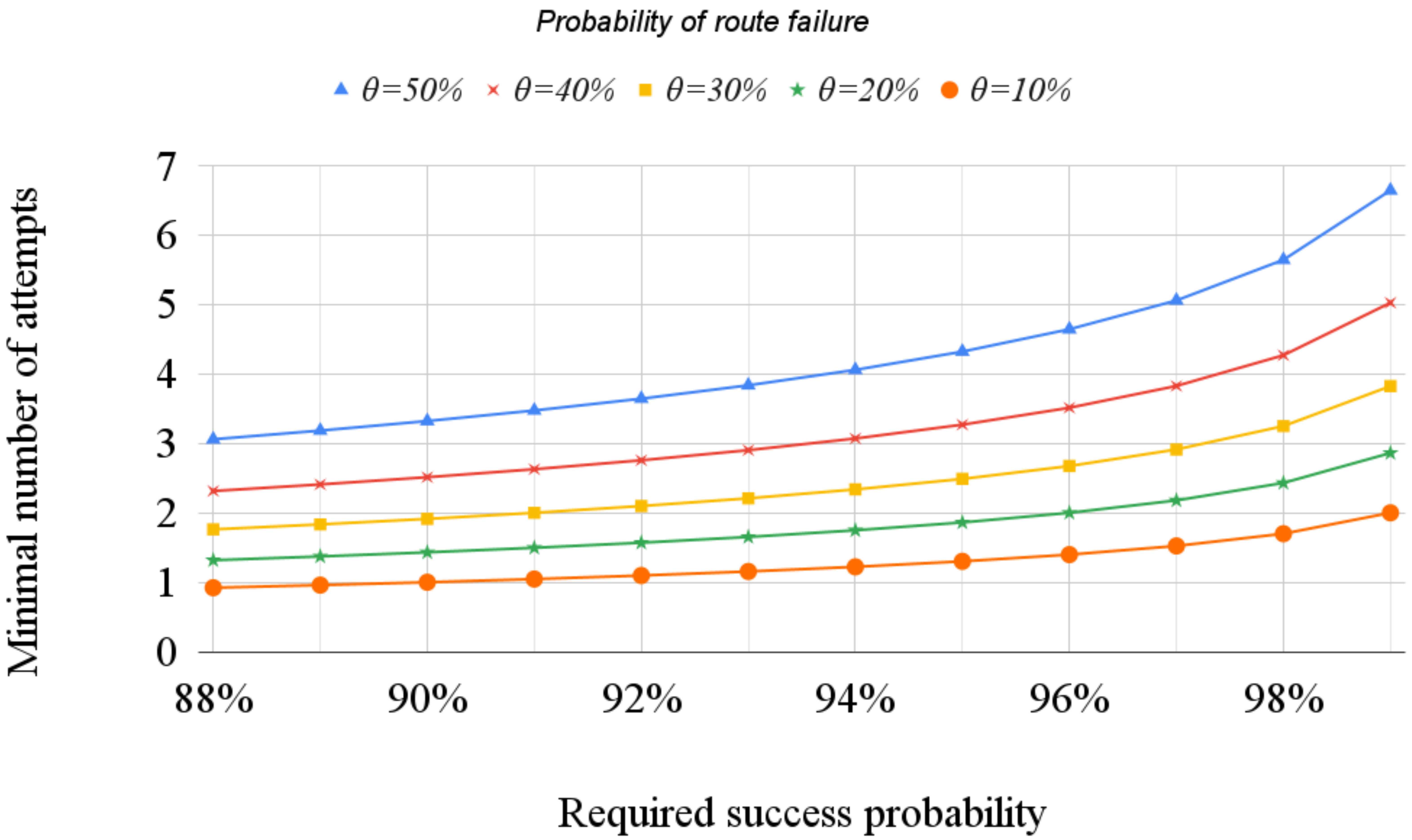


BUT THE UX!

BUT THE UX!

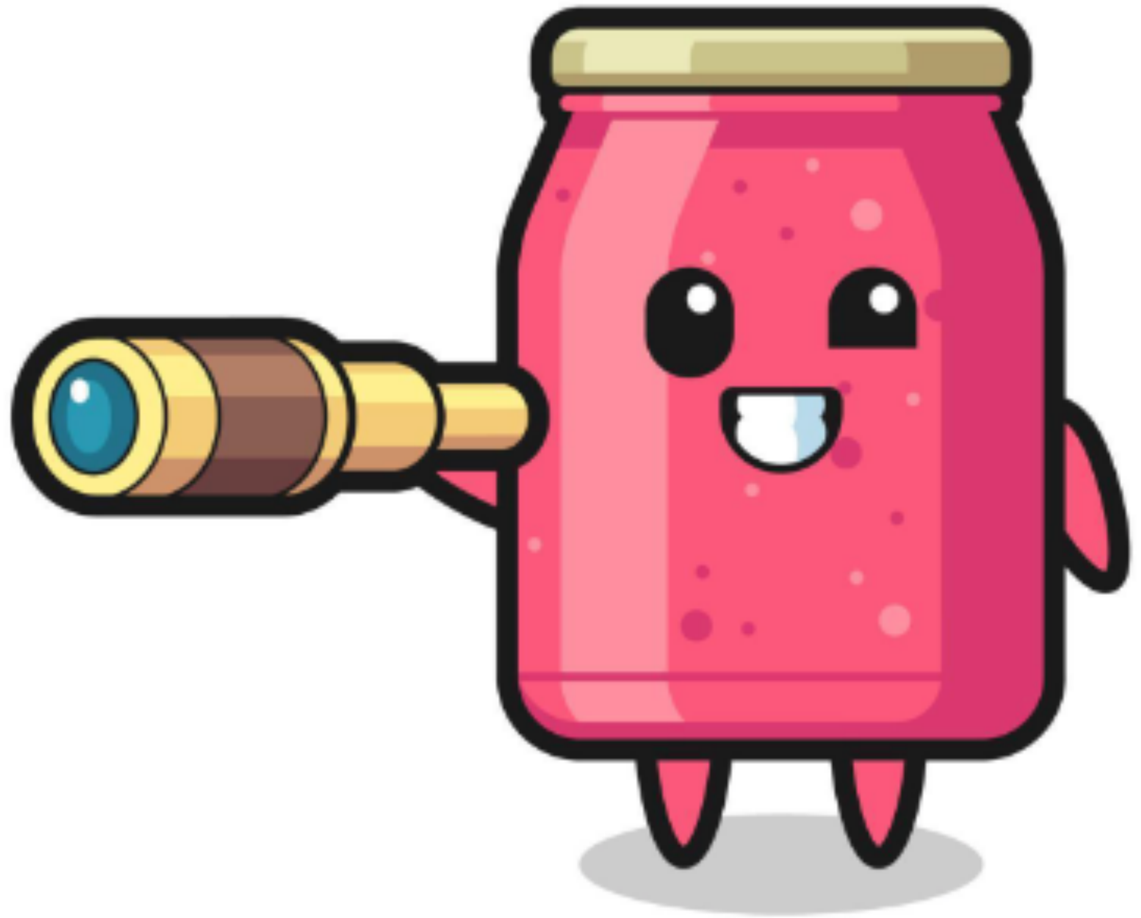
- The number of attempts to guarantee a high success rate

BUT THE UX!


Probability of route failure
▲ θ=50% × θ=40% ■ θ=30% ★ θ=20% ● θ=10%
Minimal number of attempts
0
1
2
3
4
5
6
7
88%
90%
92%
94%
96%
98%
Required success probability

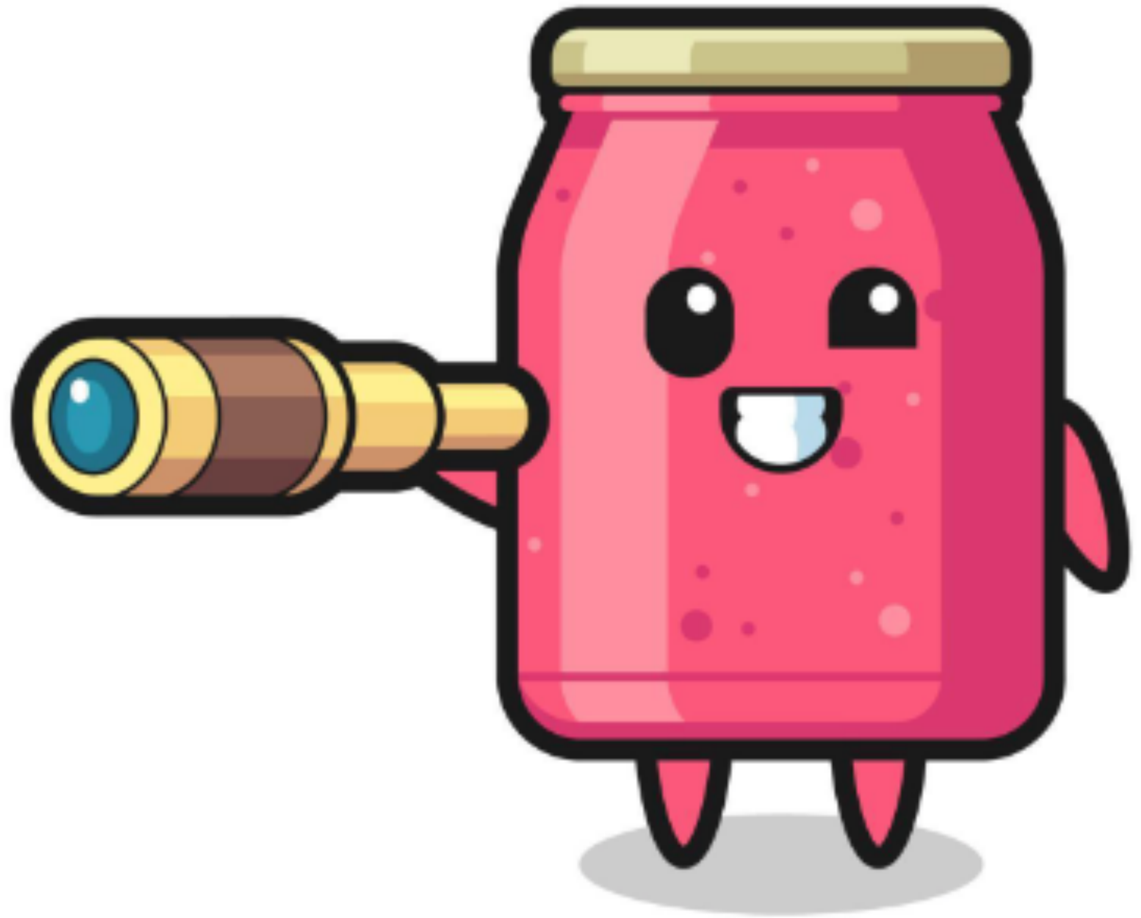
- The number of attempts to guarantee a high success rate

MAIN CHALLENGES



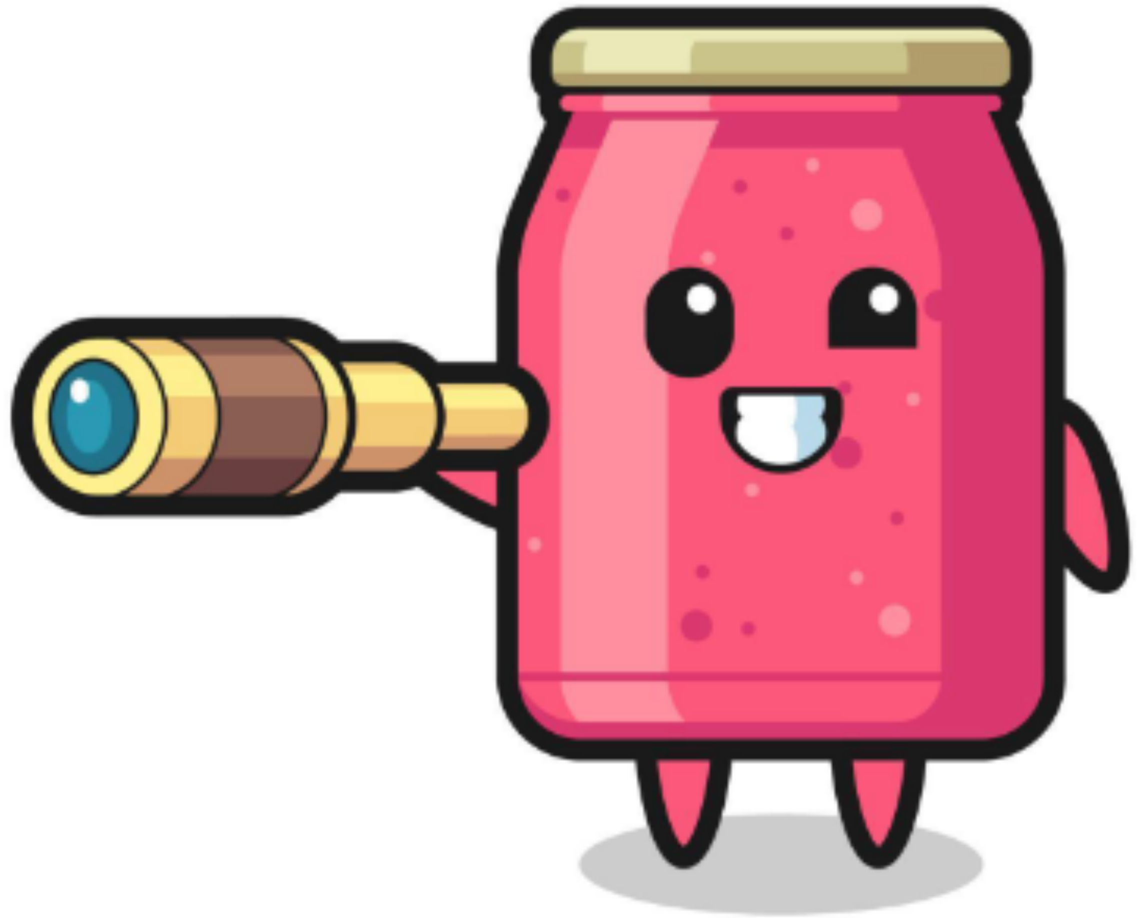
MAIN CHALLENGES

- Simulating attacks in times of peace



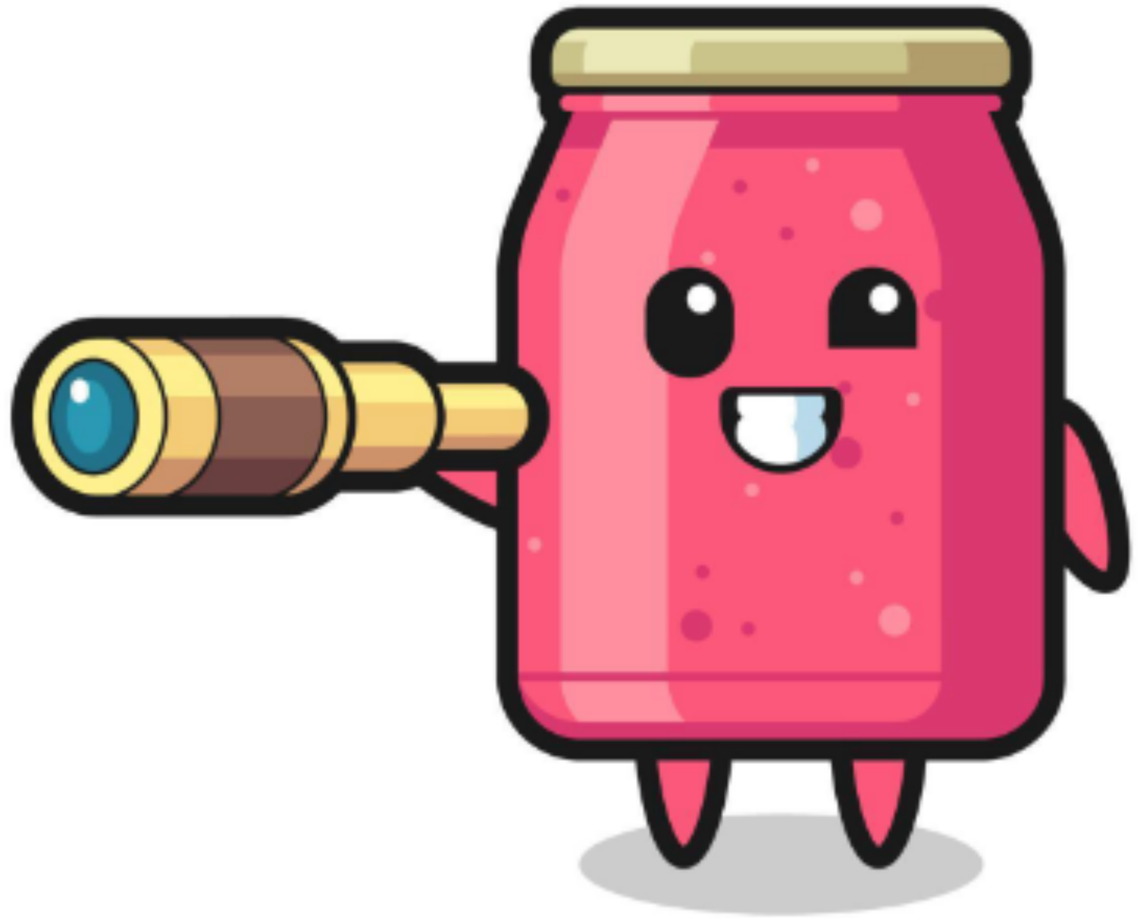
MAIN CHALLENGES

- Simulating attacks in times of peace
- Mitigation strategies creating new attack vectors



MAIN CHALLENGES

- Simulating attacks in times of peace
- Mitigation strategies creating new attack vectors
- Influence on honest users



FEES+REPUTATION

FEES+REPUTATION

- Blog post (+links to paper and PoC):

  https://research.chaincode.com/2022/11/15/unjamming-lightning/

FEES+REPUTATION

- Blog post (+links to paper and PoC):

  https://research.chaincode.com/2022/11/15/unjamming-lightning/

- Spec

  https://github.com/lightning/bolts/pull/1052

FEES+REPUTATION

- Blog post (+links to paper and PoC):

  https://research.chaincode.com/2022/11/15/unjamming-lightning/

- Spec

  https://github.com/lightning/bolts/pull/1052

- Feedback or questions?

  @ClaraShik, clara@chaincode.com

FEES+REPUTATION

- Blog post (+links to paper and PoC):

  https://research.chaincode.com/2022/11/15/unjamming-lightning/

- Spec

  https://github.com/lightning/bolts/pull/1052

- Feedback or questions?

  @ClaraShik, clara@chaincode.com

- Join our calls, every two weeks!