Lightning Network:

Cryptography to the Rescue of Cryptography?

Marco **Benedetti**, Roberto **Favaroni**,
Giuseppe **Galano,** Andrea **Gentili**,
Davide **Magnanimi**, Michela **Santangelo***

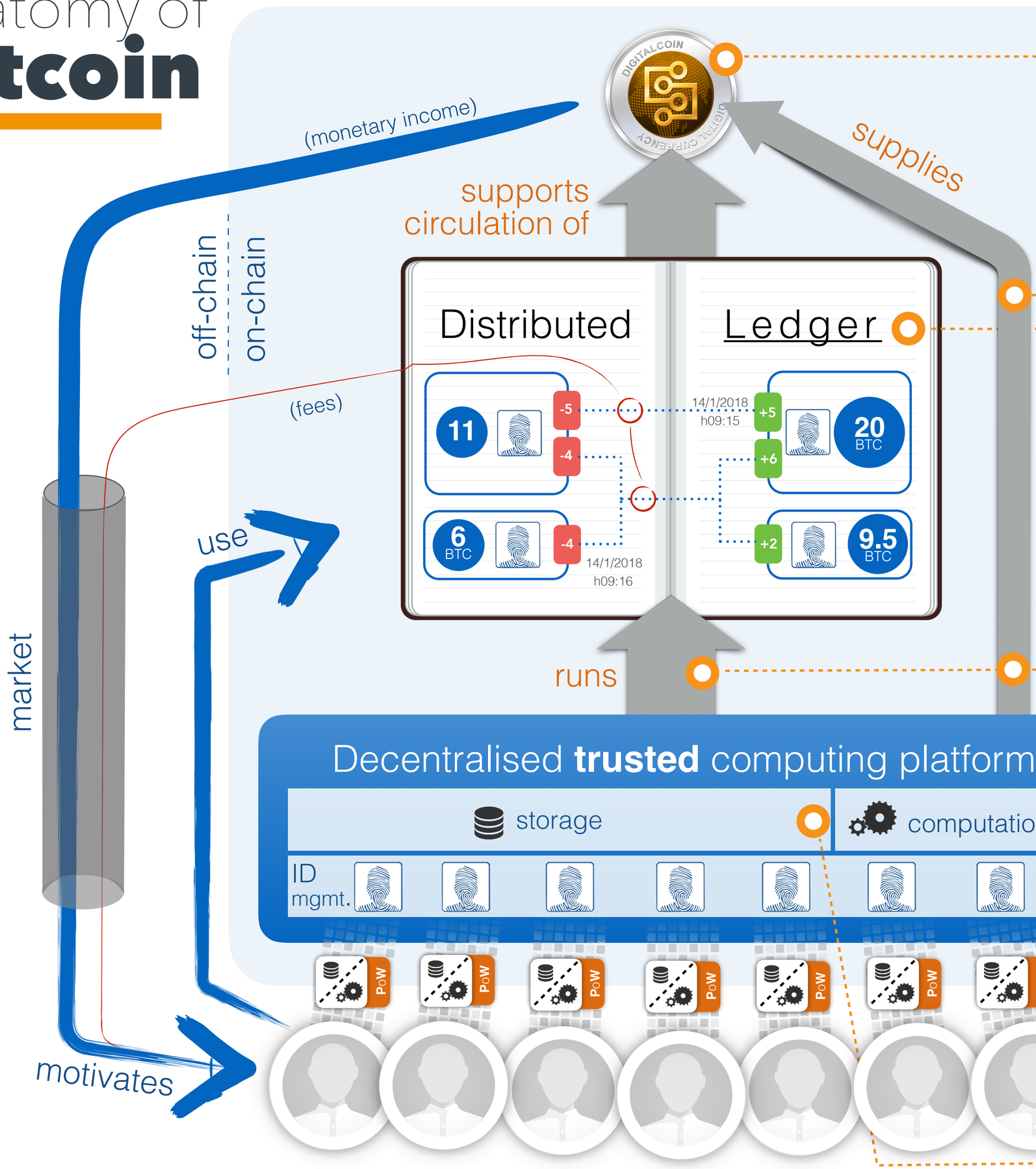
[NAME].[SURNAME]@bancaditalia.it

ART
www.bankit.art

*Intern at ART

Our “reference” cryptoasset: **Bitcoin**

Anatomy of **Bitcoin**



Virtual currency
- aka, cryptoasset
- aka, mathematical currency
- internal unit of account

Mining
- Predefined supply curve
- Decreasing rate of growth
- "Seigniorage" to participants

No double-spending
- Prevent the same token from being used twice or more

Distributed application
- e.g., Bitcoin
- several other (monetary & non-monetary) proposals

Consensus protocol
"*In case of multiple/inconsistent blockchains, every node must prefer the one backed by the most work*"

User addresses/identities
- self-generated
- no central registry; no clearance

Based on:
➥Public-Key Cryptography

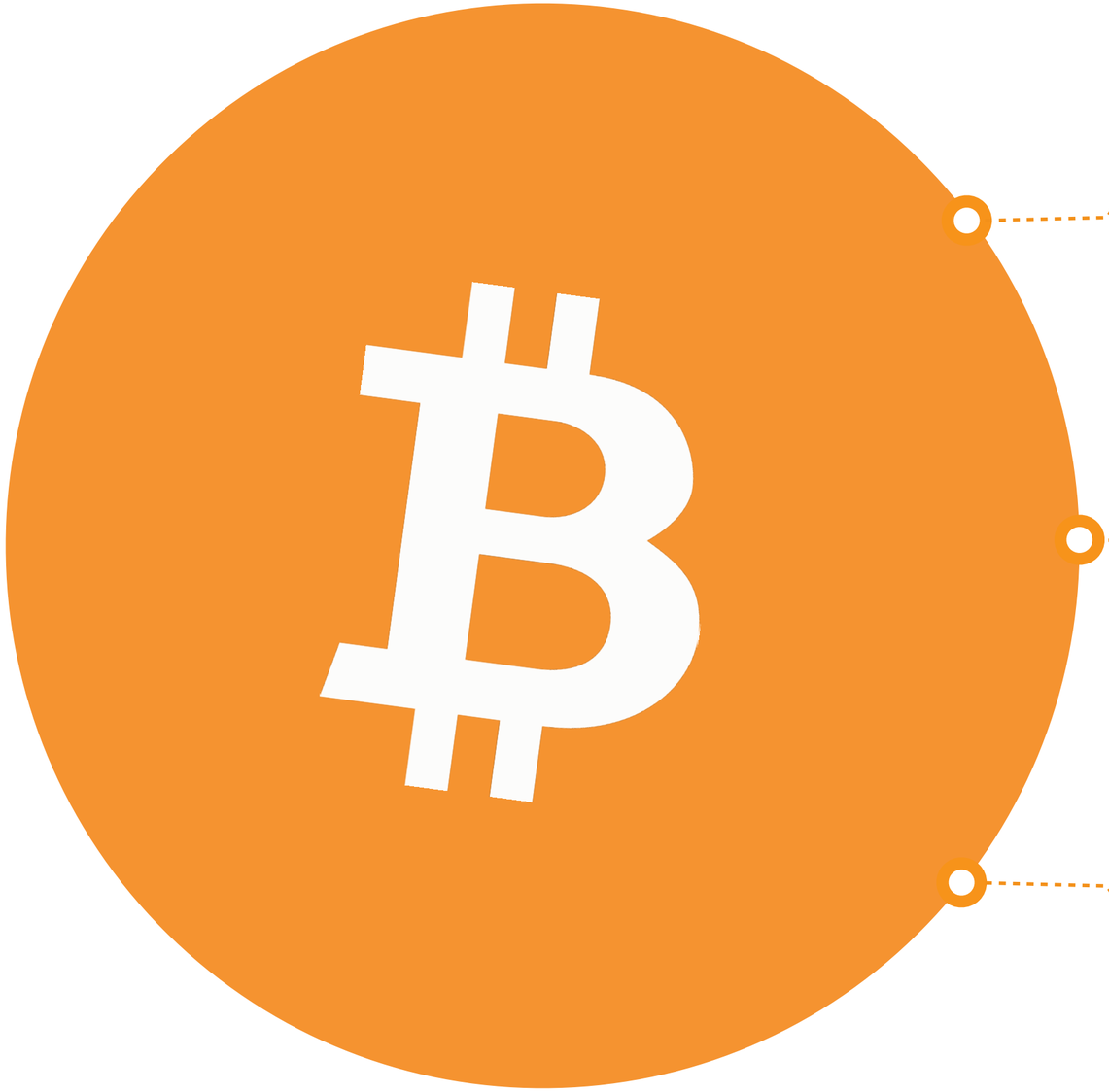
Proof of work
"*Contributions accepted only if corroborated by evidence of hard, express, dedicated work (fee)*"

Based on:
➥Cryptographic Puzzle

Blockchain

➥Integrity/structure via hash func.

Our “reference” cryptoasset: **Bitcoin**

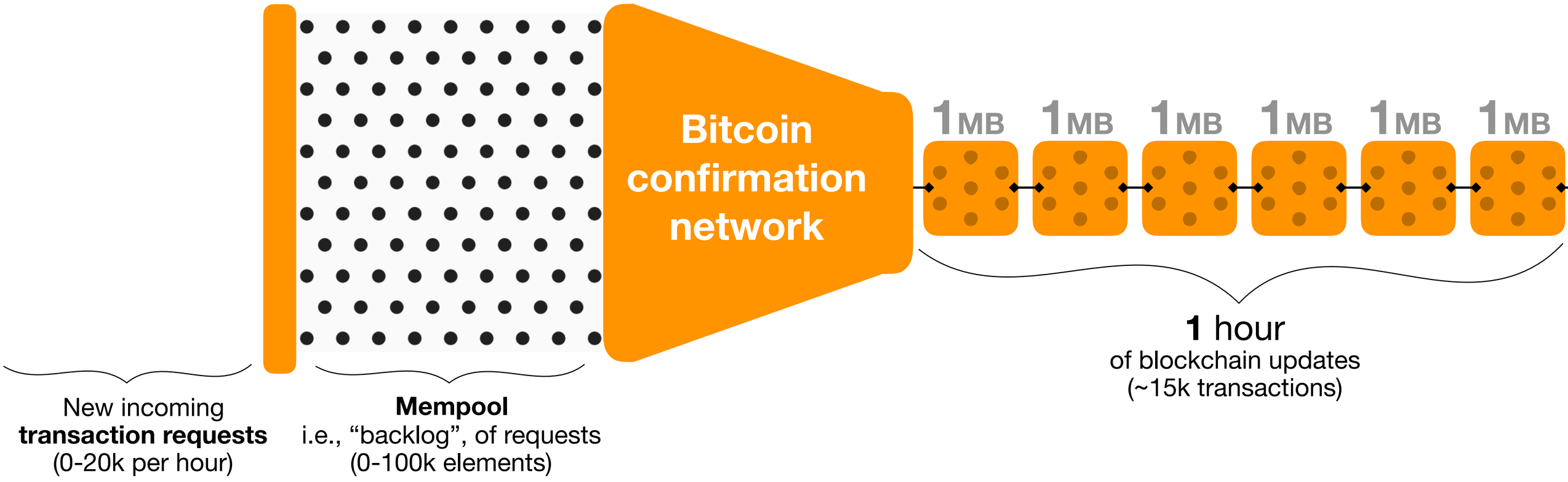


Scalability issues

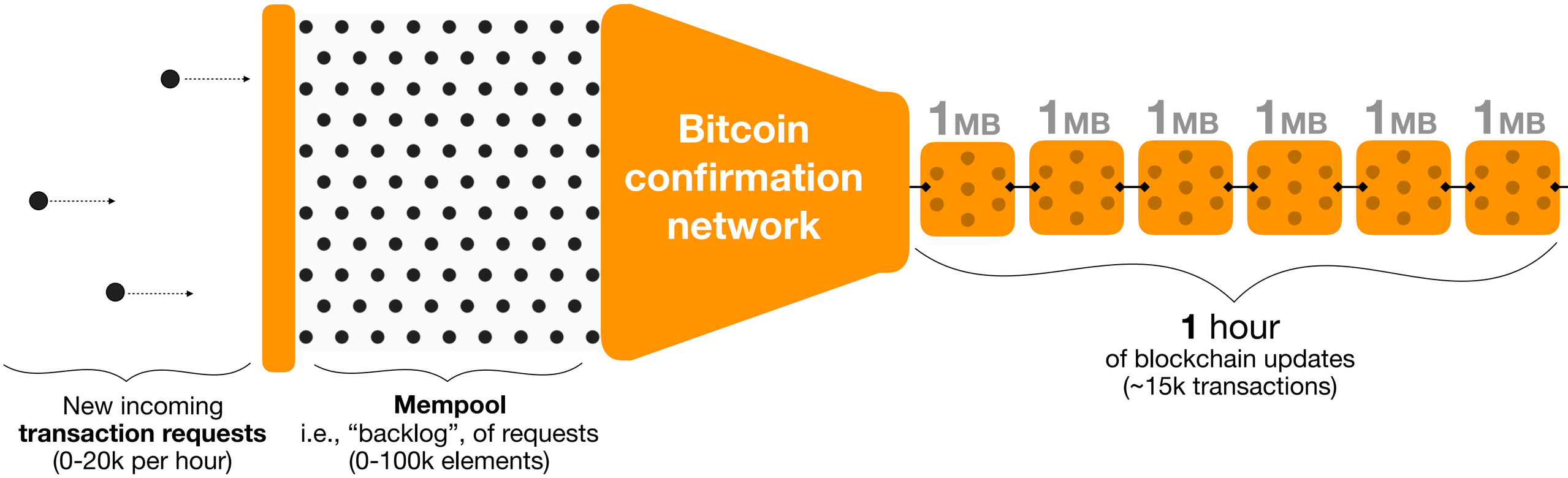
Confidentiality issues

Volatility issues (market value & fees)

Mechanics of
BTC confirmation



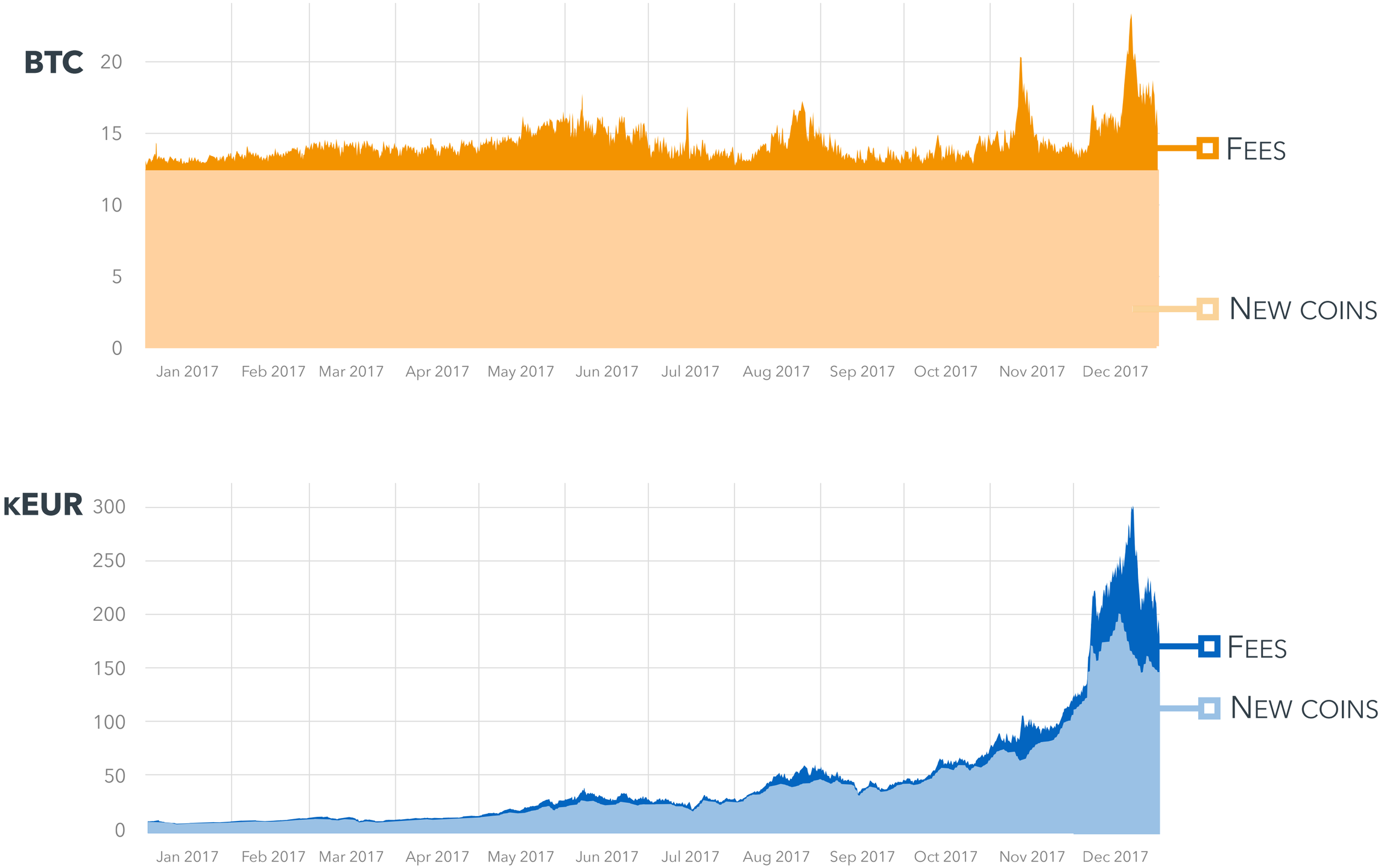
Mechanics of **BTC confirmation**



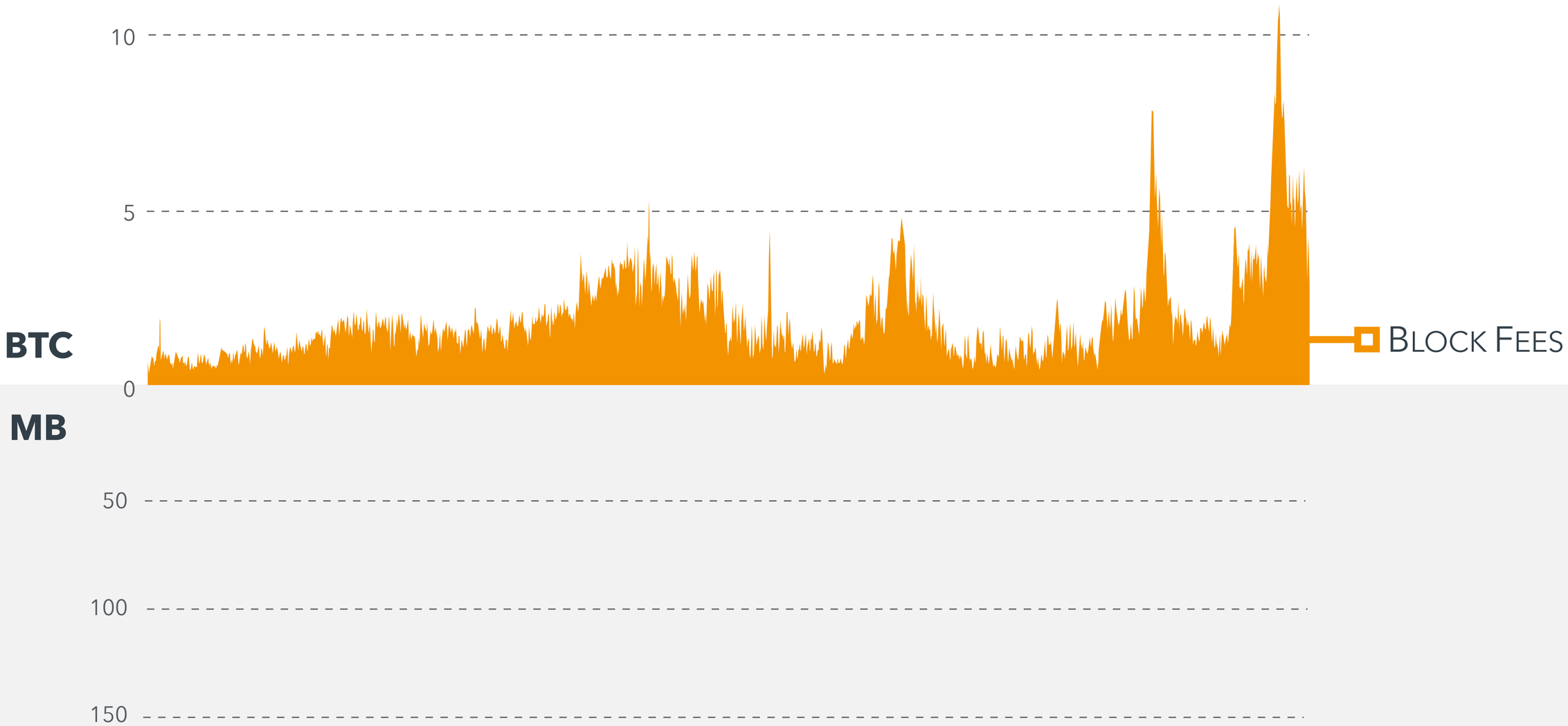
What can we do?…

- **Enlarge** the block?
  - No longer a P2P system: Too many resources required
- **Speed-up** confirmation time?
  - Less security, more instability, more double spendings to resolve
- Furthermore:
  - All these params are **fixed** at protocol level
  - Large **consensus** required to change them: unlikely

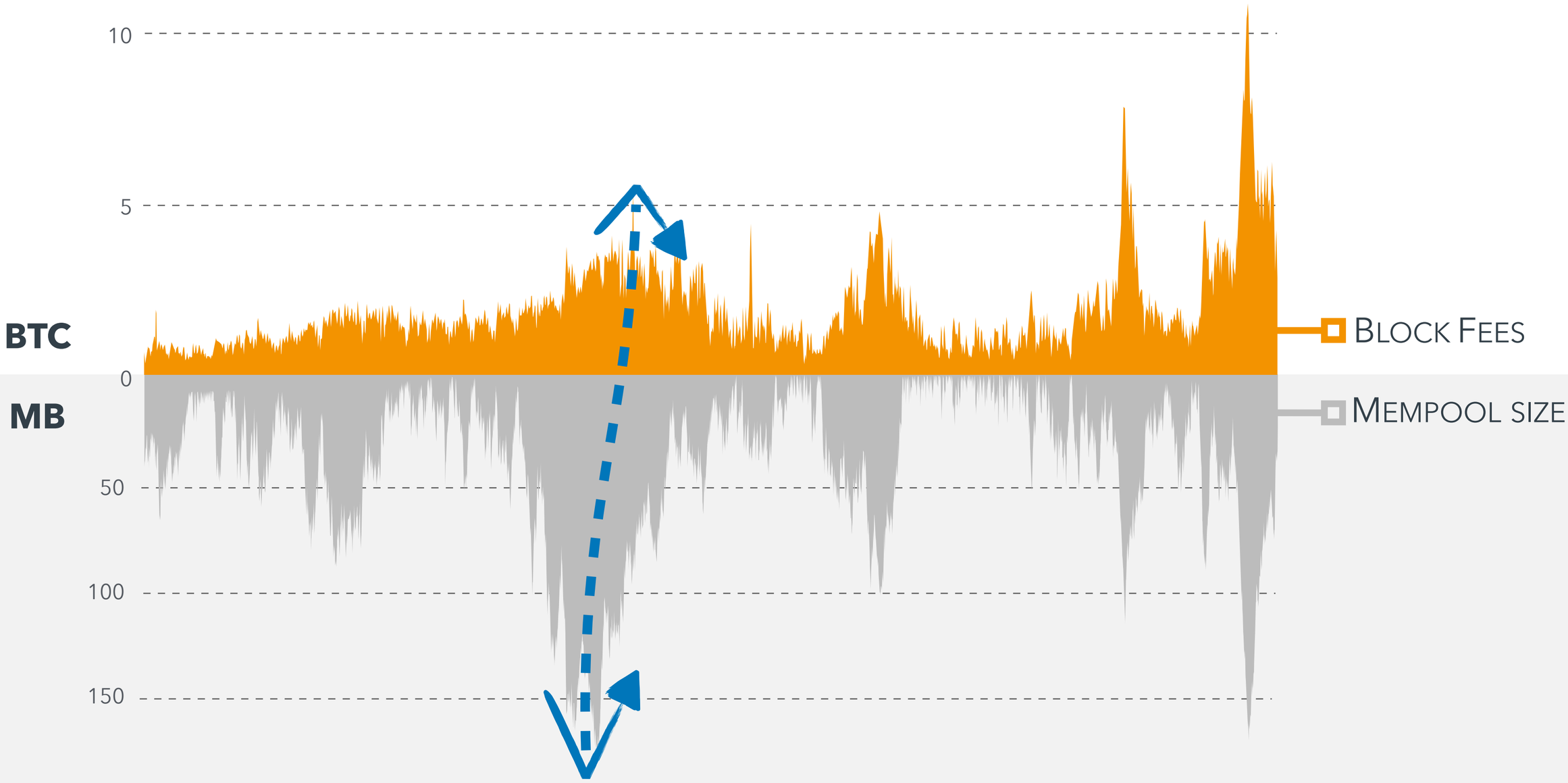
Mining **revenues** (2017) (b)



Mining **revenues** (2017) (b)

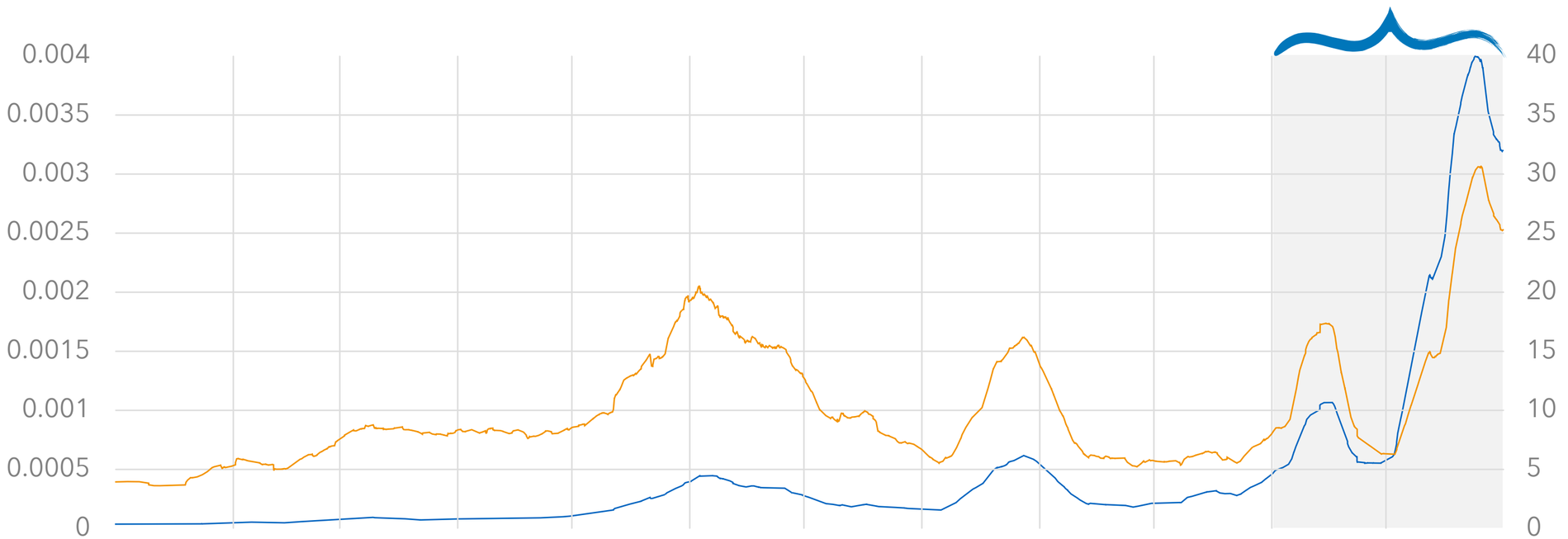


Mempool **size** VS block **fees** (b)

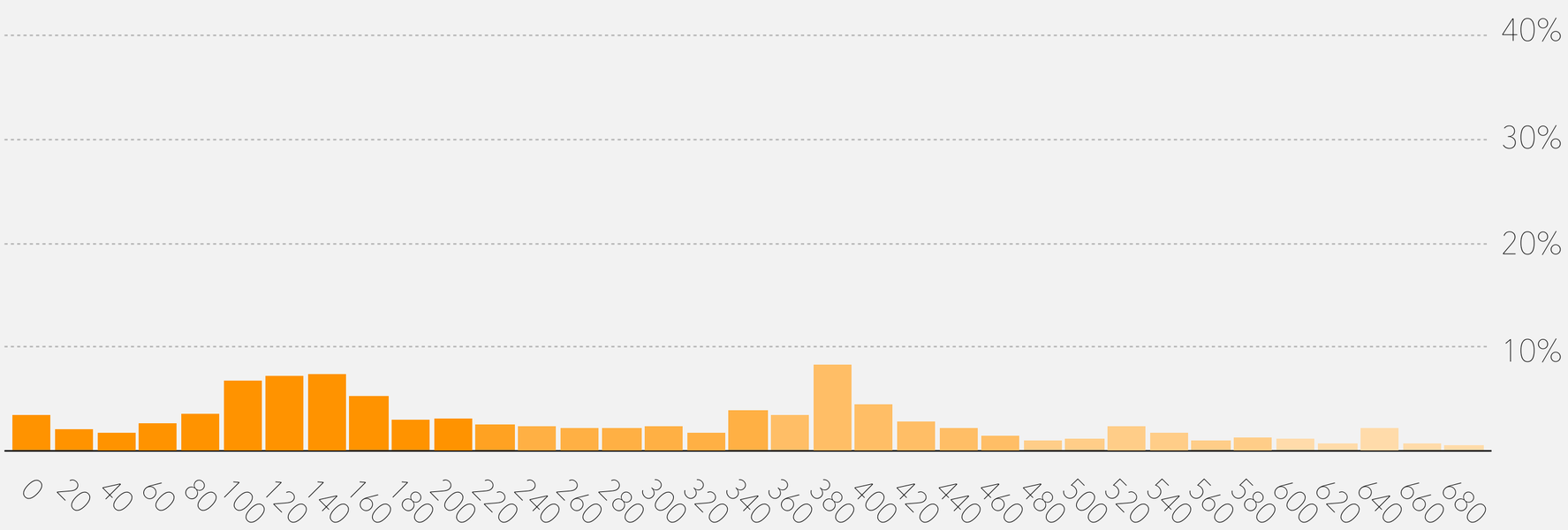


Fee **amount** and **distribution** [**2017**] (d)

Fees may suddenly become too high (>5eu/tr) for many common use cases (e.g., retail payments)



AVERAGE FEE
PER TRANSACTION
IN BTC AND EUR

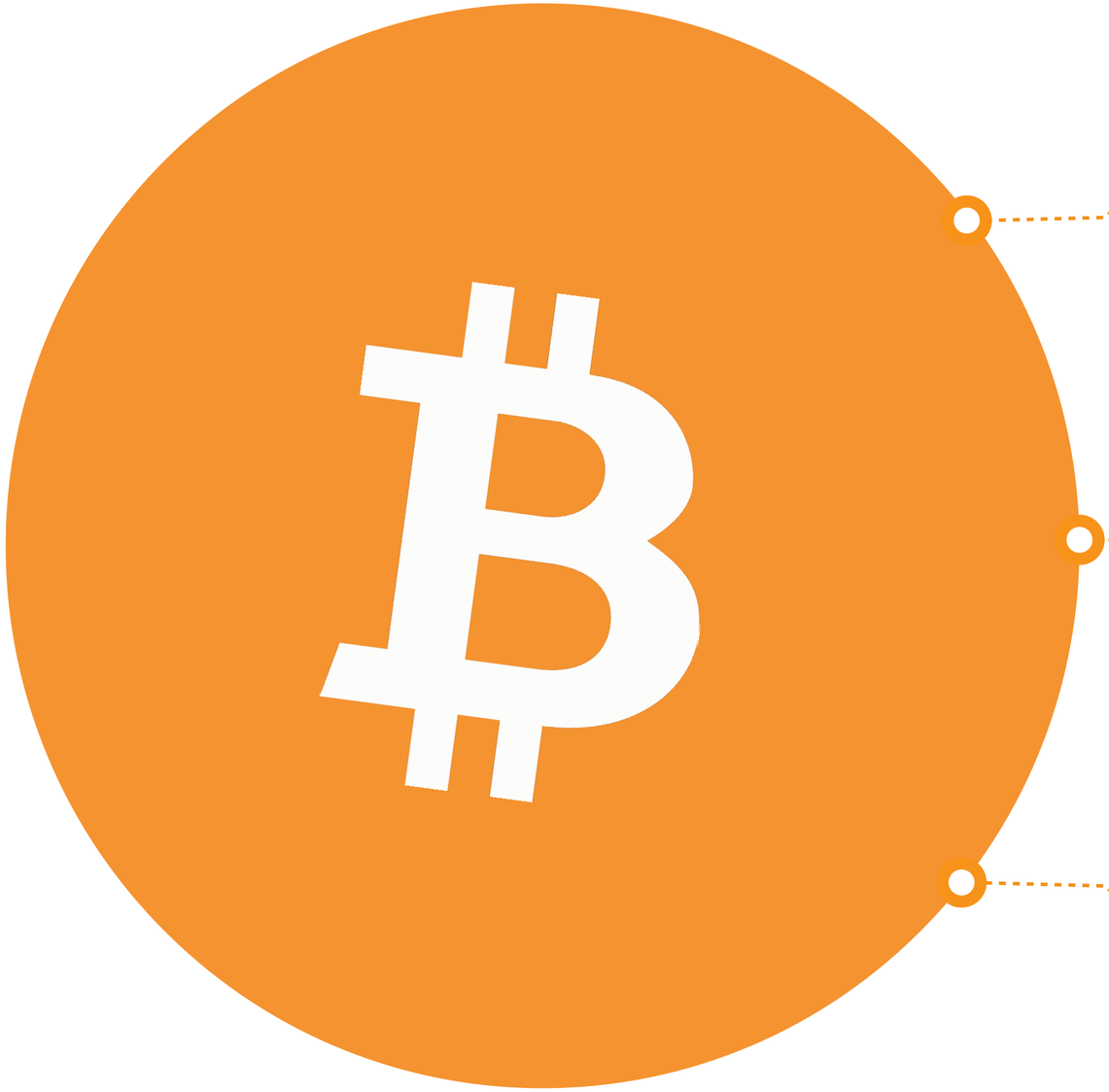


PERCENTAGE
OF TRANSACTIONS
WITH A GIVEN FEE

SATOSHI/BYTE

(WEIGHTED BY SIZE)

Our “reference” cryptoasset: **Bitcoin**

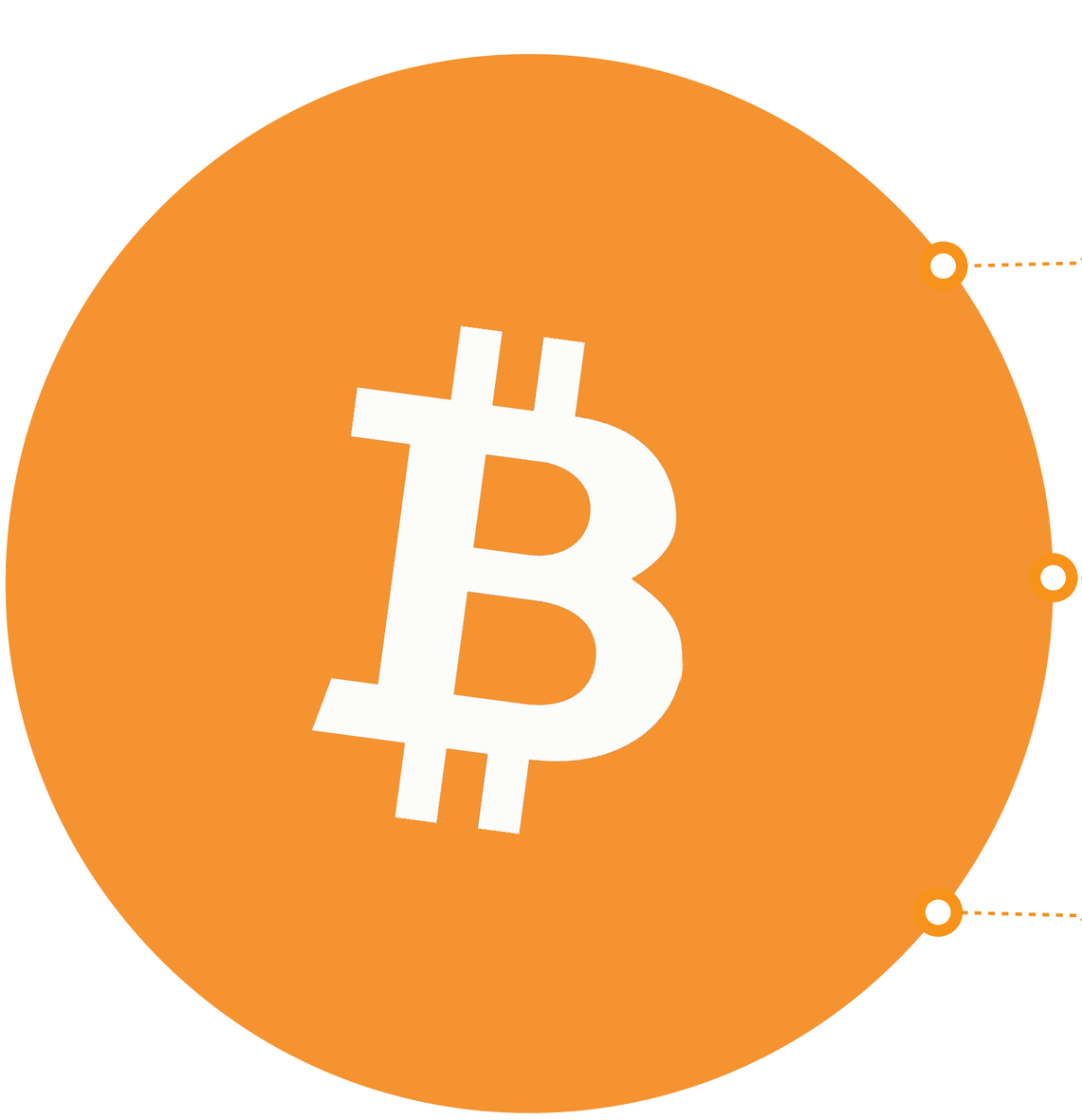


Scalability issues

Confidentiality issues

Volatility issues (market value & fees)

Our “reference” cryptoasset: **Bitcoin**

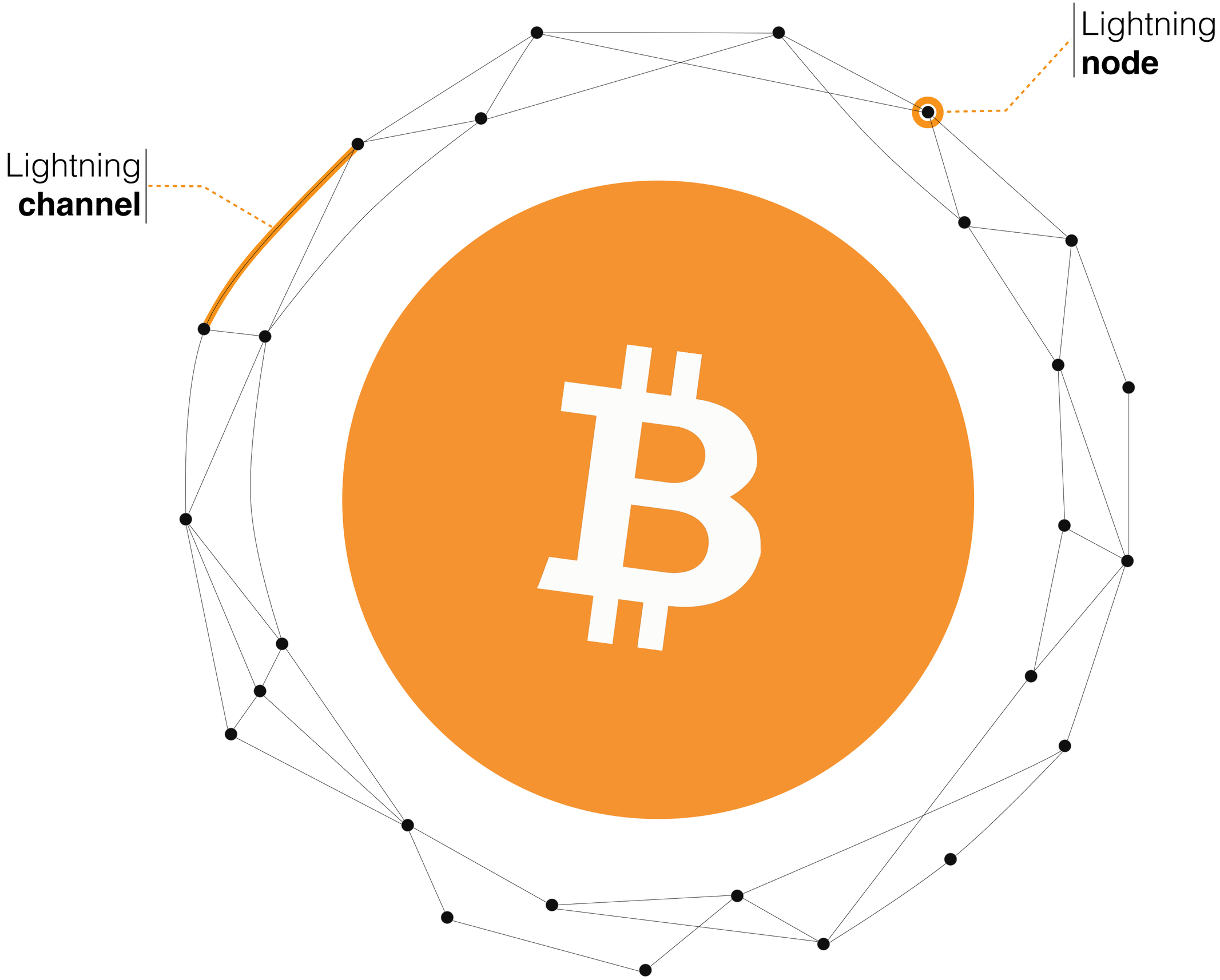


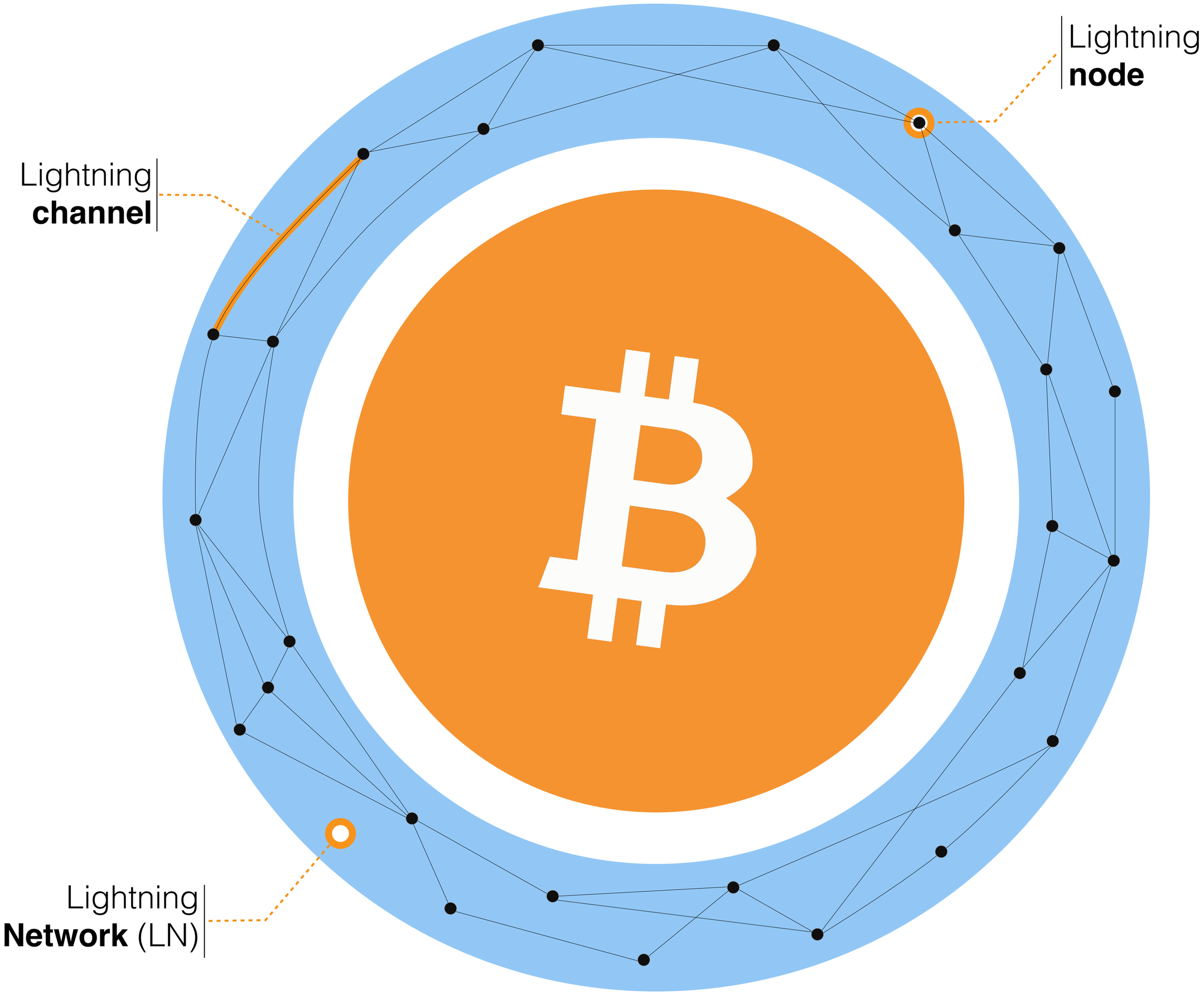
Scalability issues

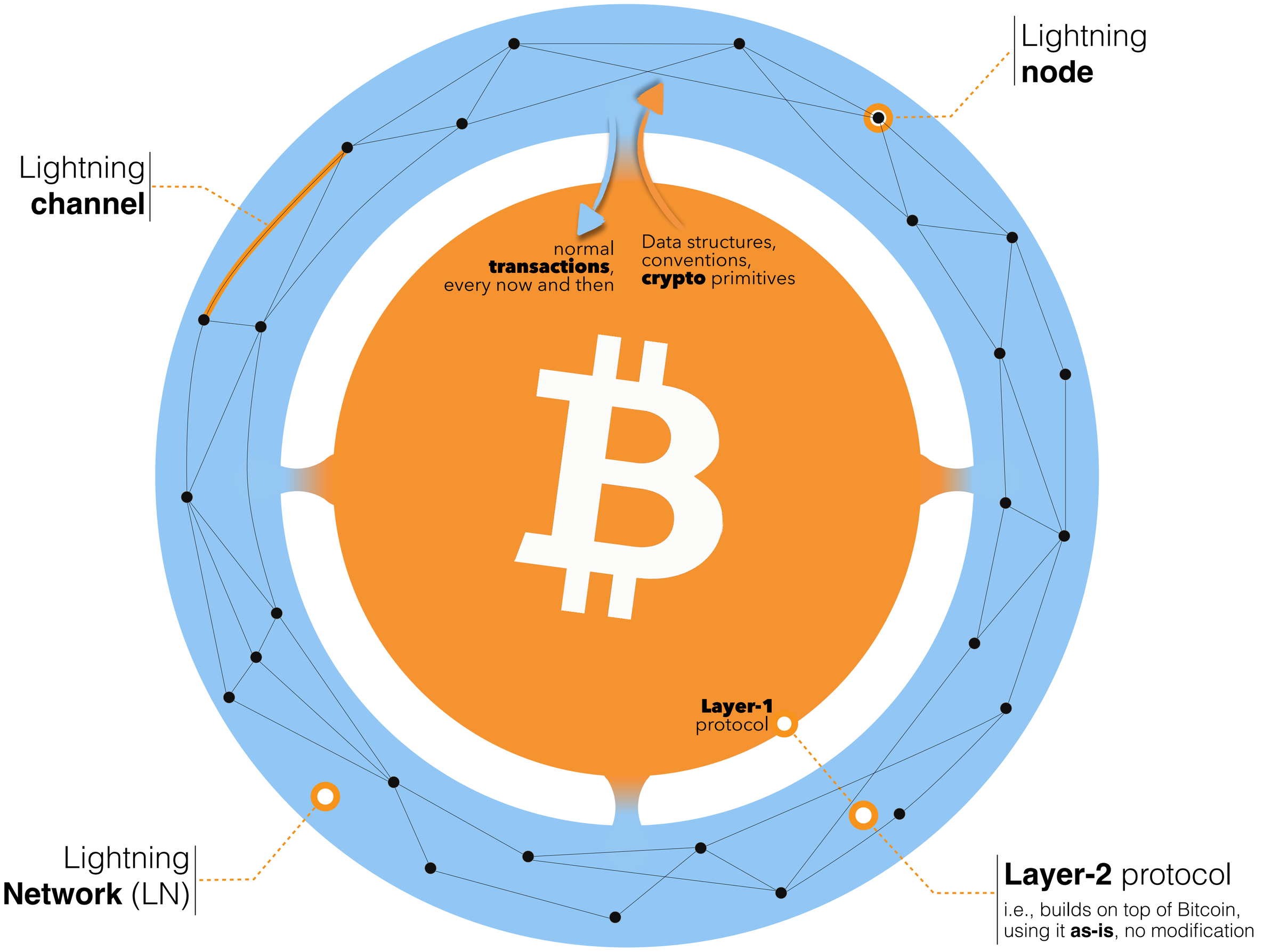
Confidentiality issues

Volatility issues (market value & fees)

Is “**crypto**” the **root** of all these issues?
Or, conversely, we can **fix** them with even more “**crypto**”?


Lightning
node
Lightning
channel


Lightning
node
Lightning
channel
Lightning
Network (LN)


Lightning
node
Lightning
channel
normal
transactions,
every now and then
Data structures,
conventions,
crypto primitives
Layer-1
protocol
Lightning
Network (LN)
Layer-2 protocol
i.e., builds on top of Bitcoin,
using it **as-is**, no modification

Lightning
channel

Lightning
node

“Real” **P2P**

Each node only maintains **its own state**, and talks with its direct **neighbours** only (computationally light)

normal
transactions,
every now and then

Data structures,
conventions,
crypto primitives

P2P…at the network level; but: **gossiping** used, meant to realise a decentralised ledger of which every node needs a copy (computationally heavy)

even **more**
trust-less
network

Layer-1
protocol

Trust-less network

nodes don’t trust each other nor anything else (but Bitcoin)

Lightning
Network (LN)

Layer-2 protocol

i.e., builds on top of Bitcoin, using it **as-is**, no modification

Fast transactions ➥ **instant**-payments
Low fees ➥ **micro**-payments

Lightning
channel

Lightning
node

“Real” **P2P**

Each node only maintains **its own state**, and talks with its direct **neighbours** only (computationally light)

normal **transactions**, every now and then

Data structures, conventions, **crypto** primitives

Slow and **costly** transactional engine

P2P…at the network level; but: **gossiping** used, meant to realise a decentralised ledger of which every node needs a copy (computationally heavy)

even **more** **trust-less** network

Layer-1 protocol

Trust-less network

nodes don’t trust each other nor anything else (but Bitcoin)

Lightning
Network (LN)

Layer-2 protocol

i.e., builds on top of Bitcoin, using it **as-is**, no modification

Fast transactions ➥ **instant**-payments
Low fees ➥ **micro**-payments

Lightning
channel

Lightning
node

"Real" **P2P**

Each node only maintains **its own state**, and talks with its direct **neighbours** only (computationally light)

normal **transactions**, every now and then

Data structures, conventions, **crypto** primitives

Slow and **costly** transactional engine

P2P…at the network level; but: **gossiping** used, meant to realise a decentralised ledger of which every node needs a copy (computationally heavy)

even **more trust-less** network

Layer-1 protocol

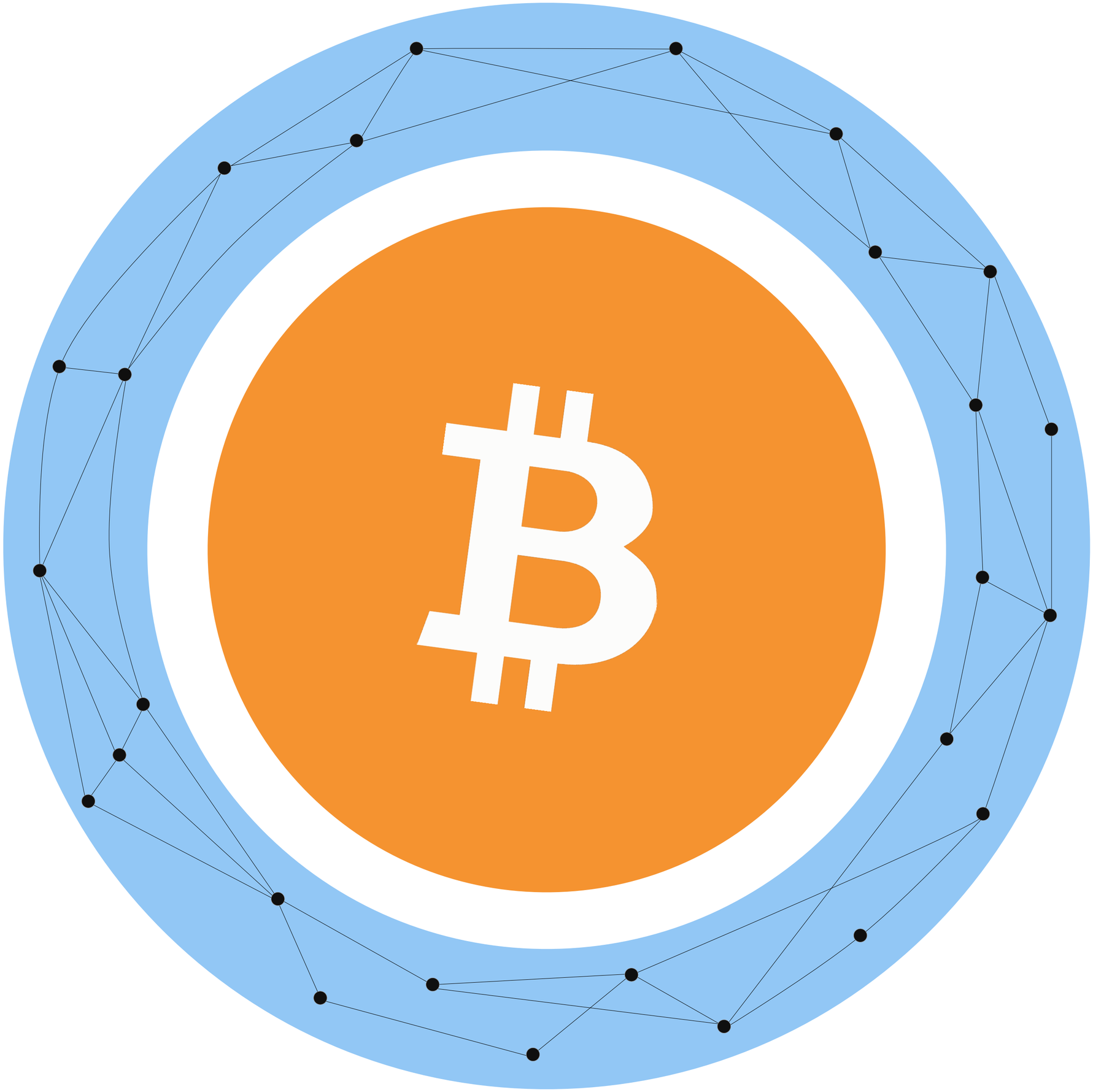
Trust-less network

nodes don't trust each other nor anything else (but Bitcoin)

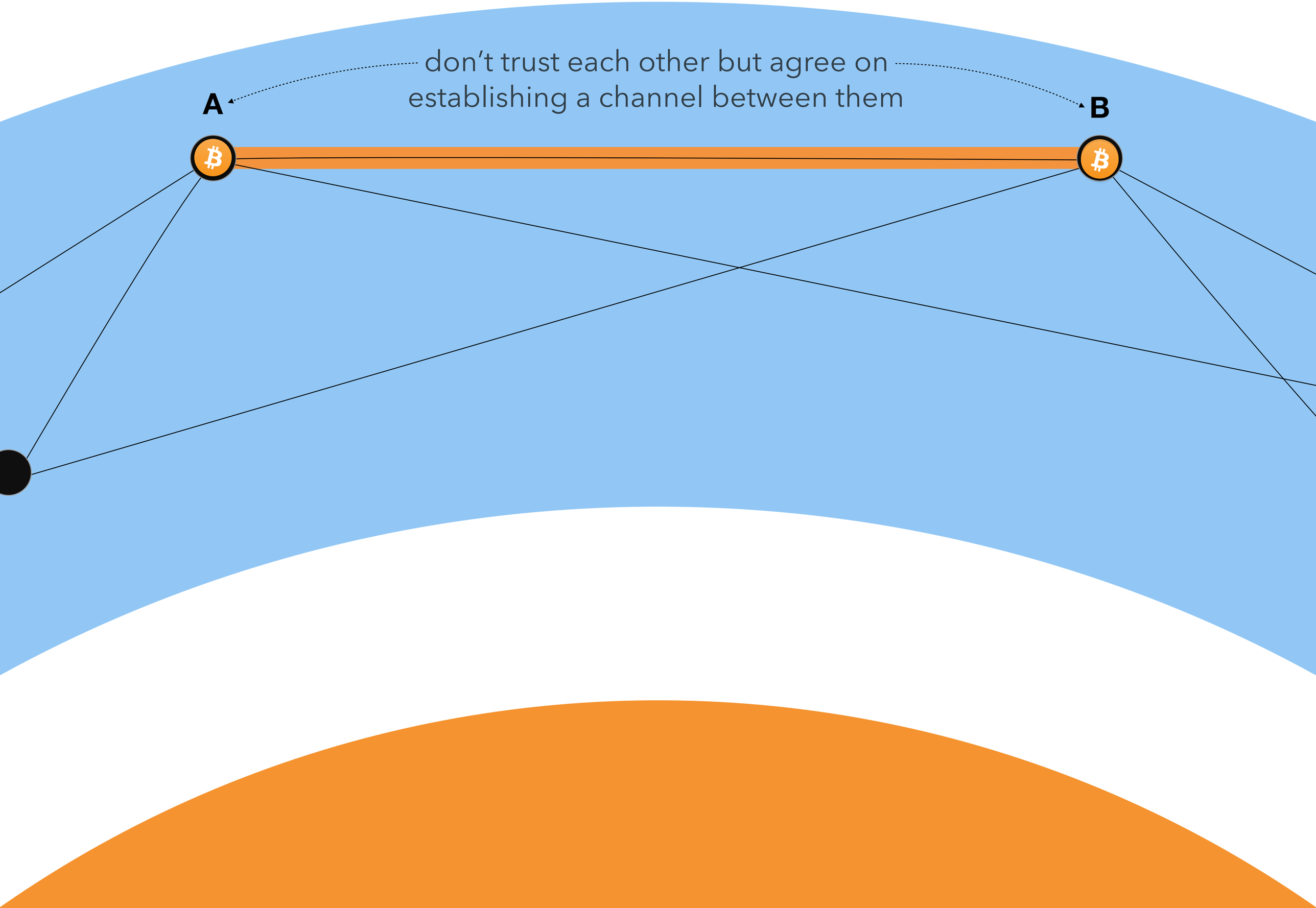
Lightning
Network (LN)

Layer-2 protocol

i.e., builds on top of Bitcoin, using it **as-is**, no modification



1st ingredient: Trust-less Bi-directional Payment Channel



1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on establishing a channel between them

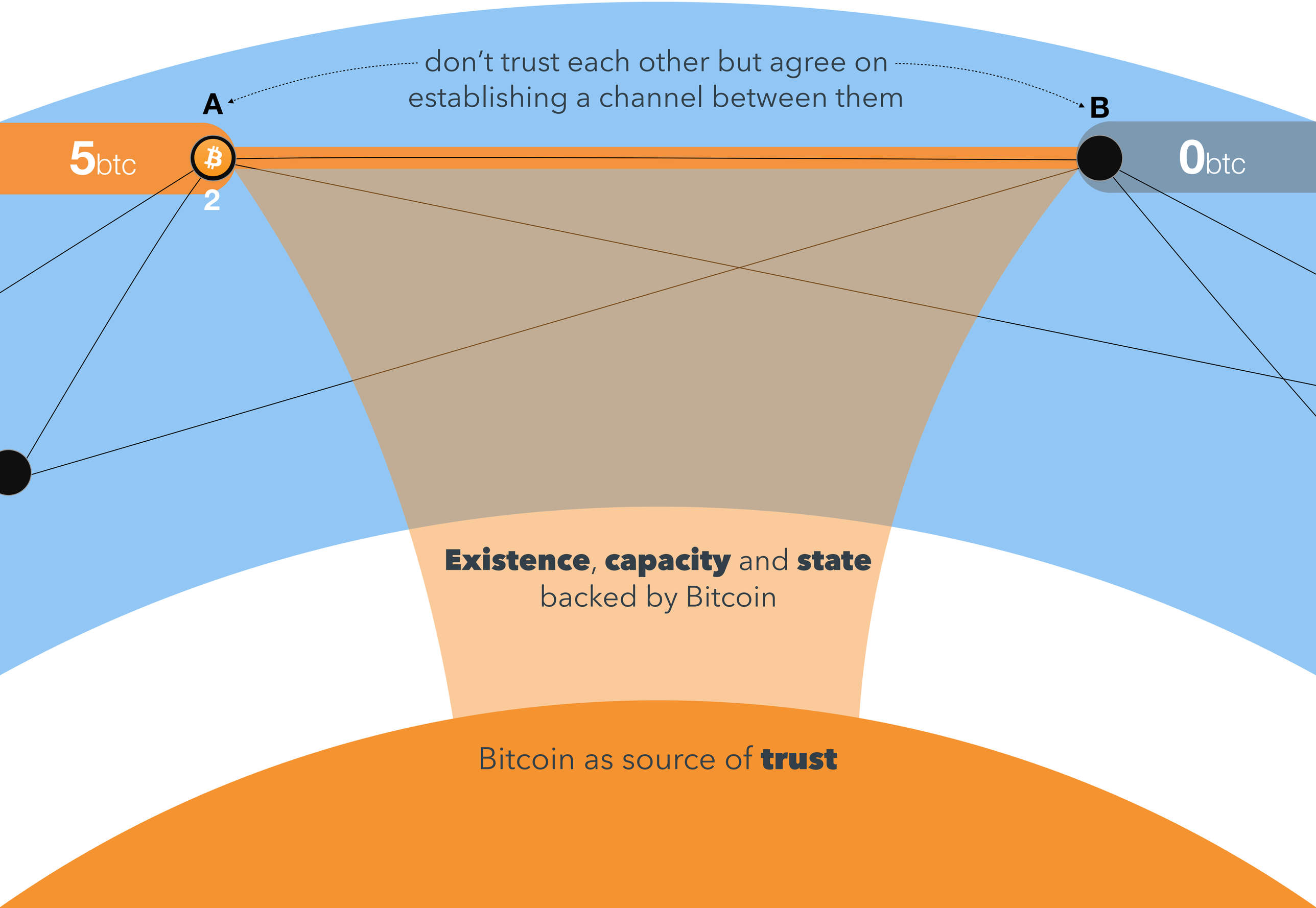
A

B

Existence, **capacity** and **state** backed by Bitcoin

Bitcoin as source of **trust**

1st ingredient: Trust-less Bi-directional Payment Channel



1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on
establishing a channel between them

A

B

3btc

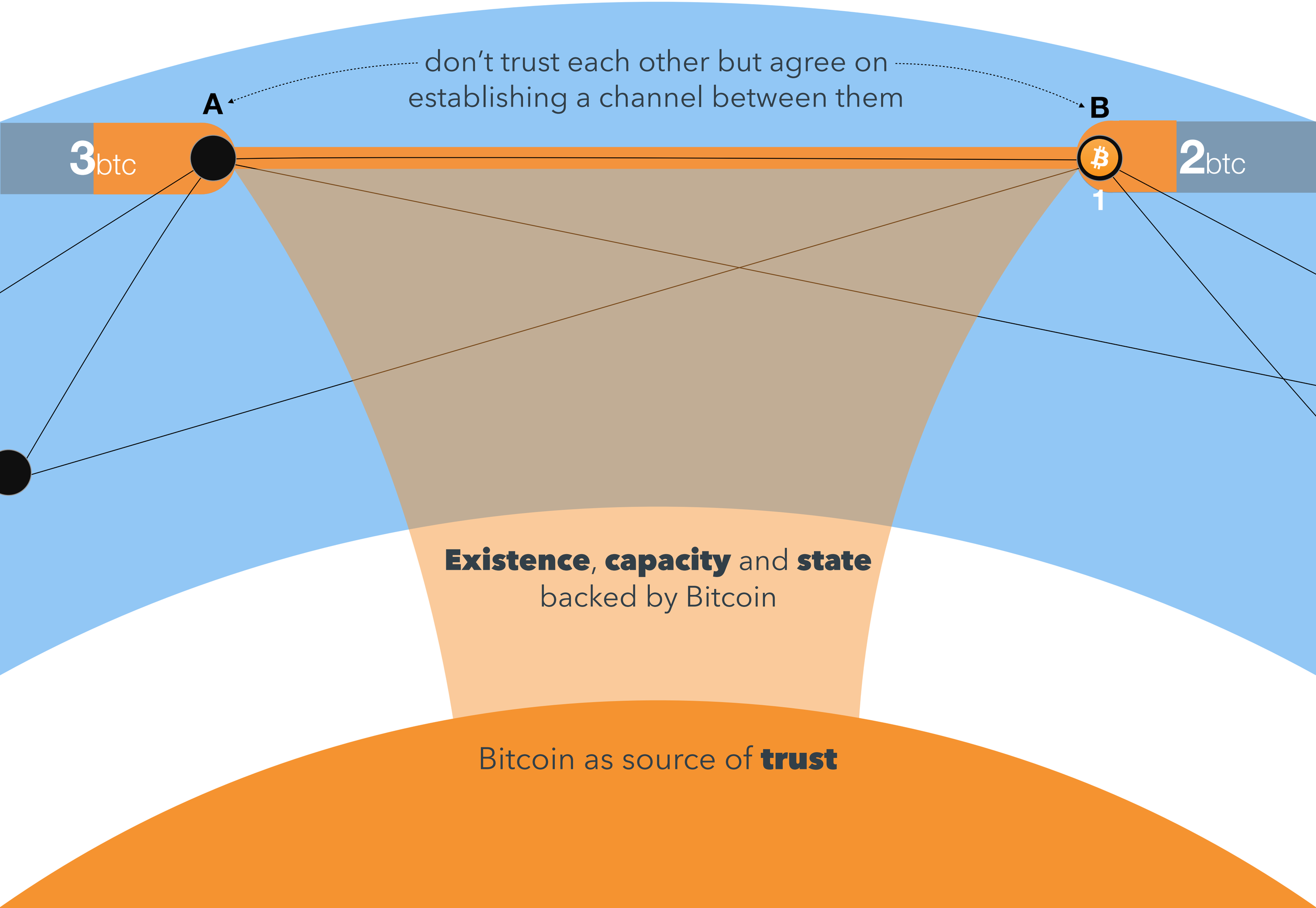
2btc

2

Existence, **capacity** and **state**
backed by Bitcoin

Bitcoin as source of **trust**

1st ingredient: Trust-less Bi-directional Payment Channel



1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on
establishing a channel between them

A

B

4btc

1

1btc

Existence, **capacity** and **state**
backed by Bitcoin

Bitcoin as source of **trust**

1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on establishing a channel between them

A

B

4btc

1btc

a **Existence, capacity**

b **state updates**

c **Netting & closure**

backed by Bitcoin

Bitcoin as source of **trust**

1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on establishing a channel between them

A

B

4btc

1btc

a

Existence, capacity

1 transaction
(always, slow)

b

state updates

never on-chain,
if cooperative
(lightning fast)

c

Netting & closure

1-2 transactions
(always, slow or very slow)

Bitcoin as source of **trust**

1st ingredient: Trust-less Bi-directional Payment Channel

don't trust each other but agree on establishing a channel between them

A

B

4btc

1btc

off-chain

a

Existence, capacity

1 transaction (always, slow)

b

state updates

never on-chain, if cooperative (lightning fast)

1-2 transactions if things go astray

c

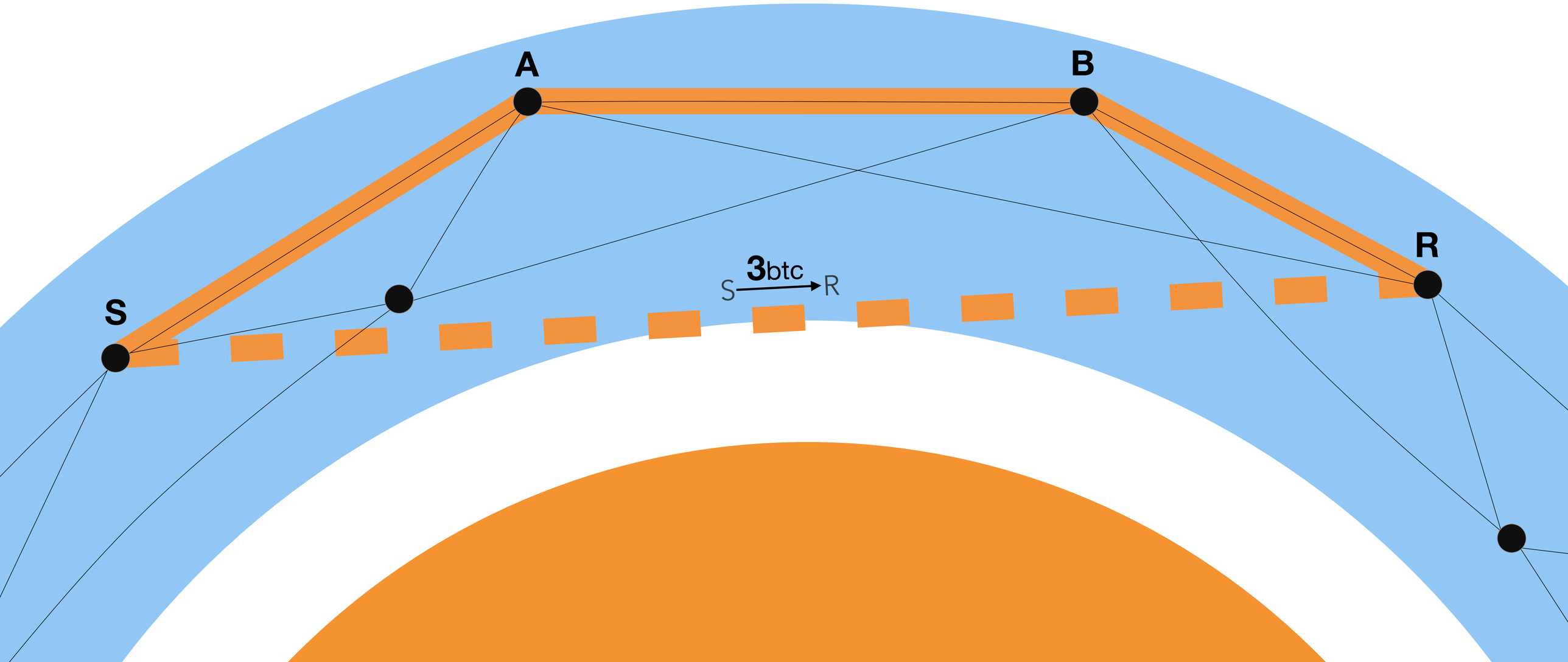
Netting & closure

1-2 transactions (always, slow or very slow)

on-chain

Bitcoin as source of **trust**

2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism



2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism

Dear A, please:
take these 3BTC and **send them to B**; also, tell B to **forward the money to R** from me.

A

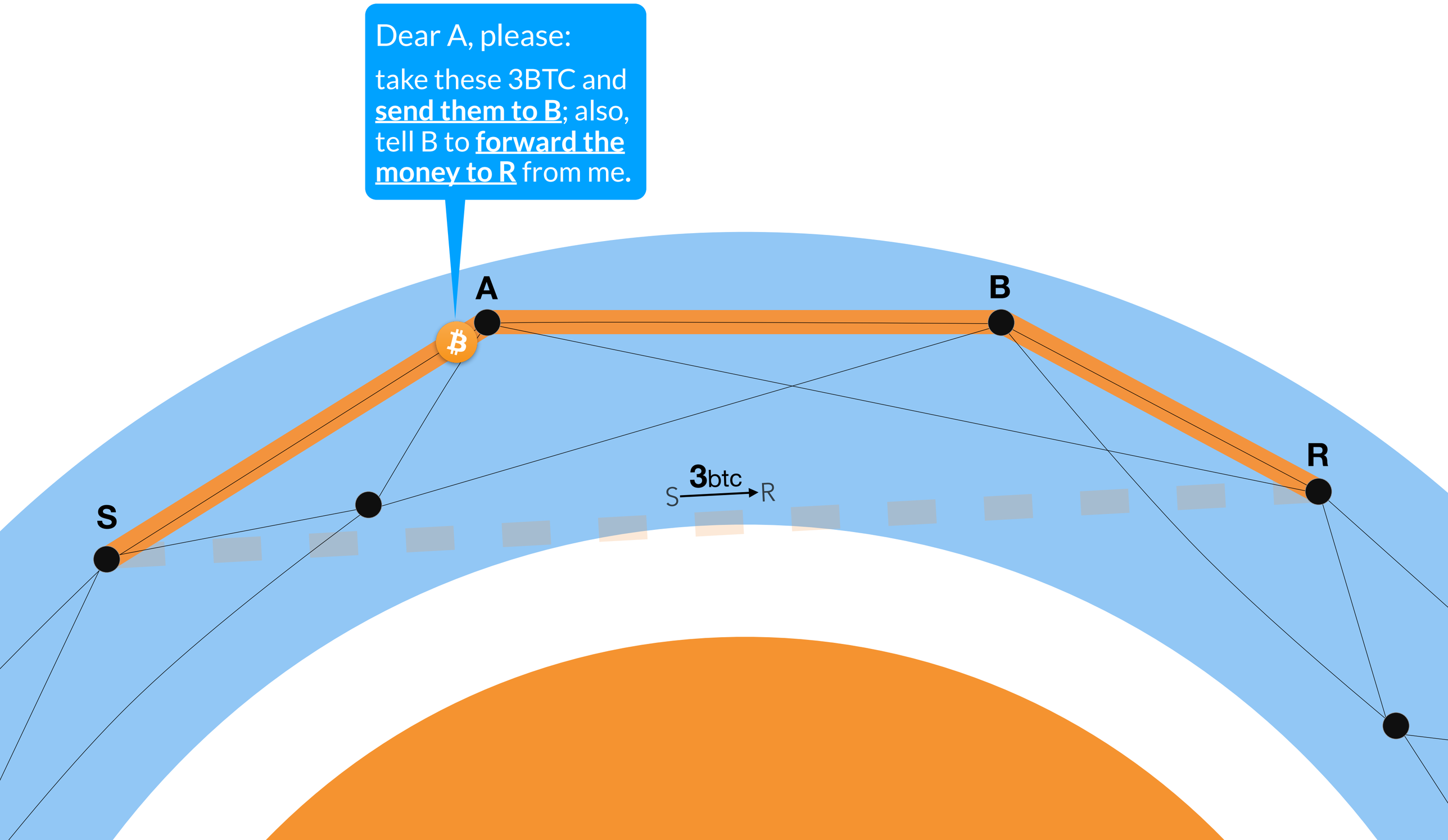
B

S

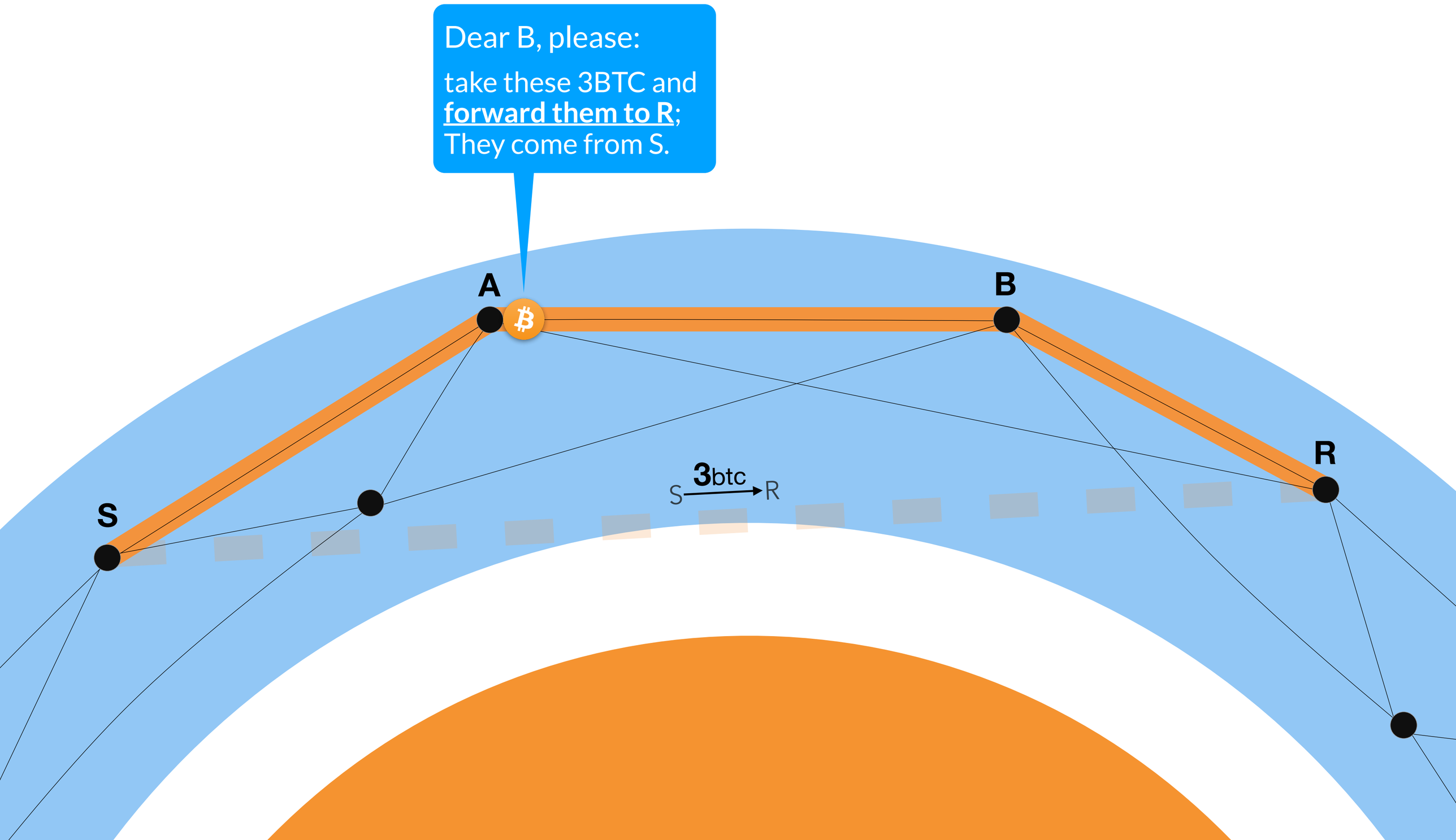
R

3btc
S → R

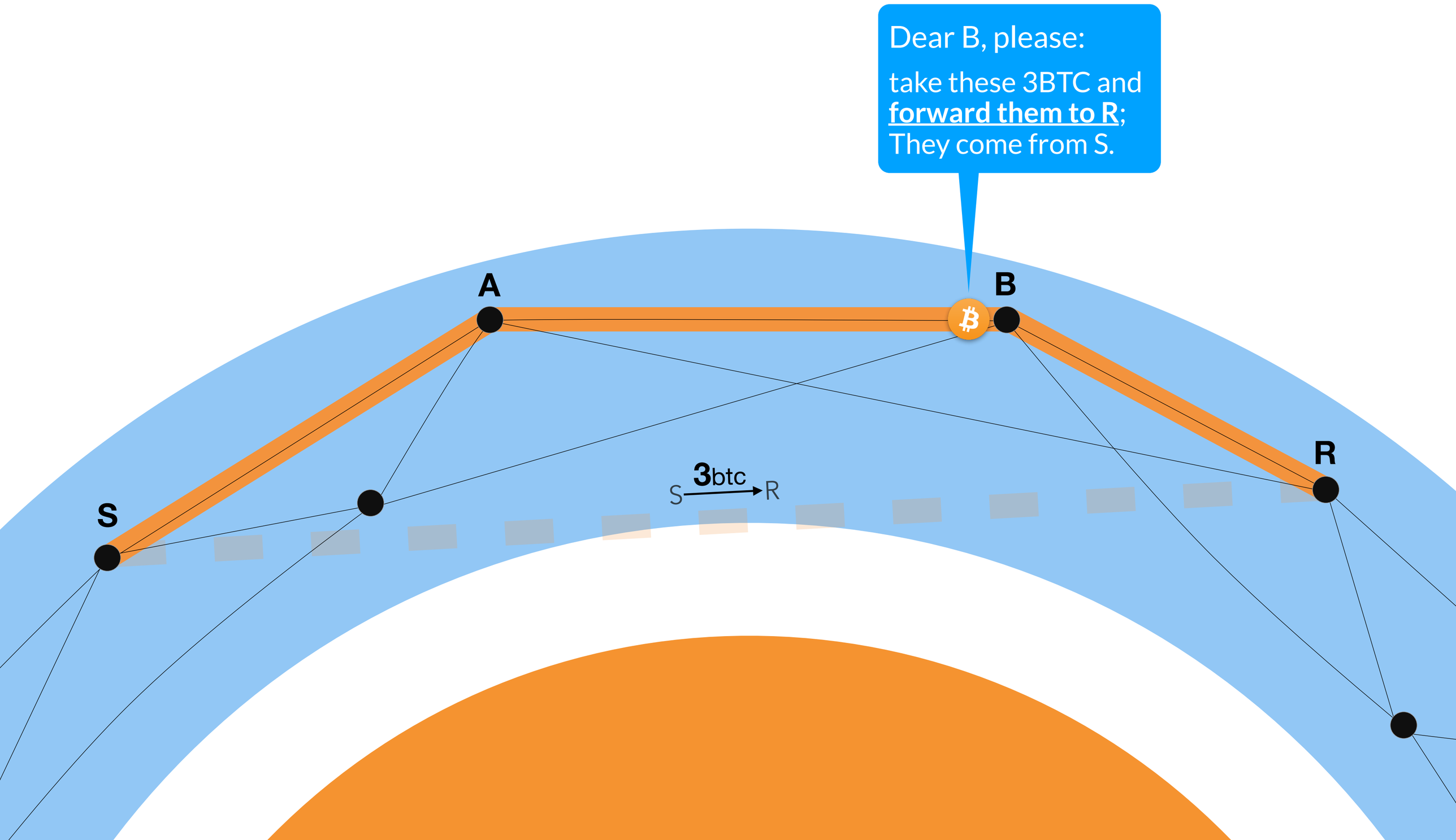
2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism



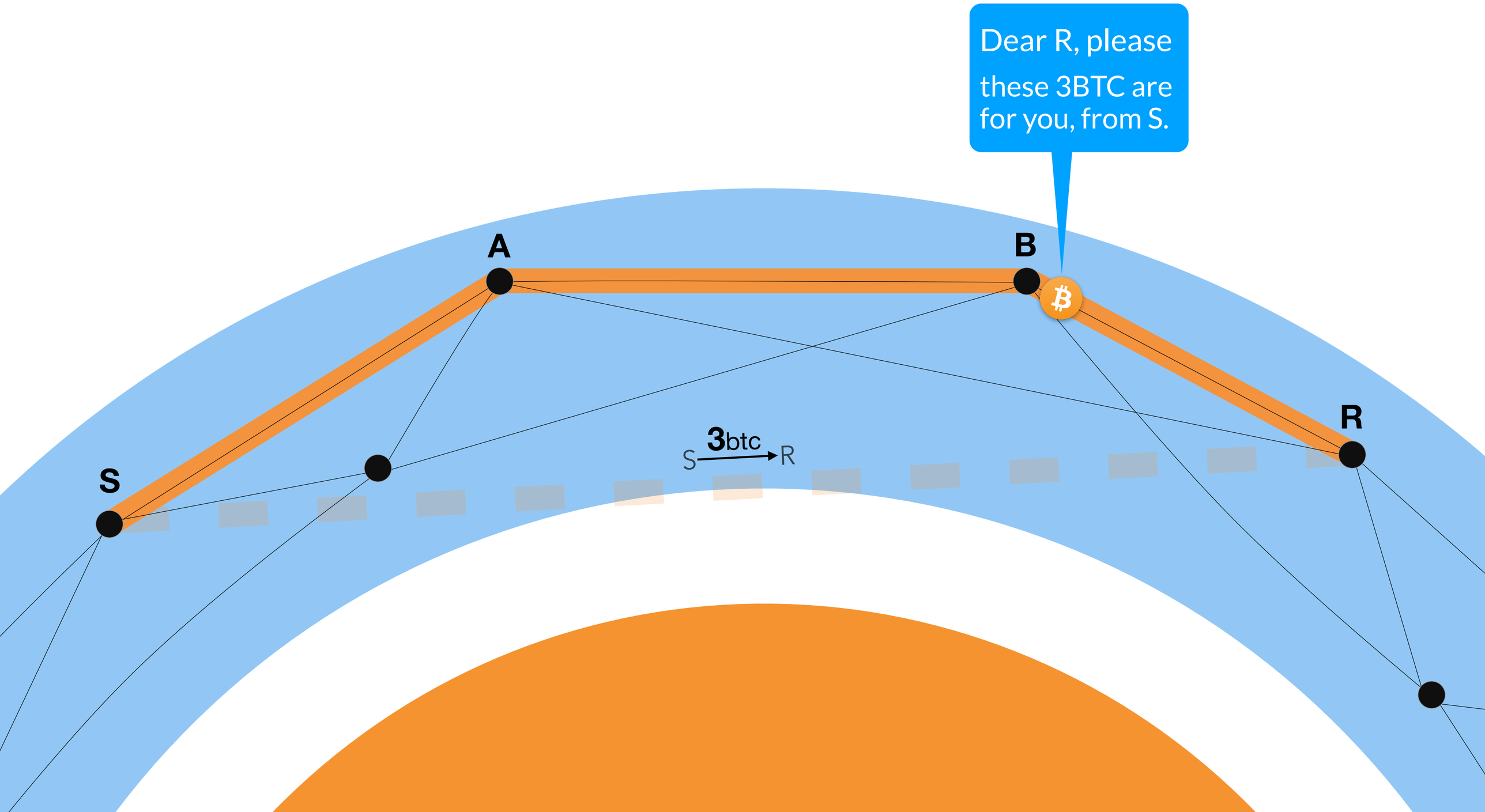
2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism



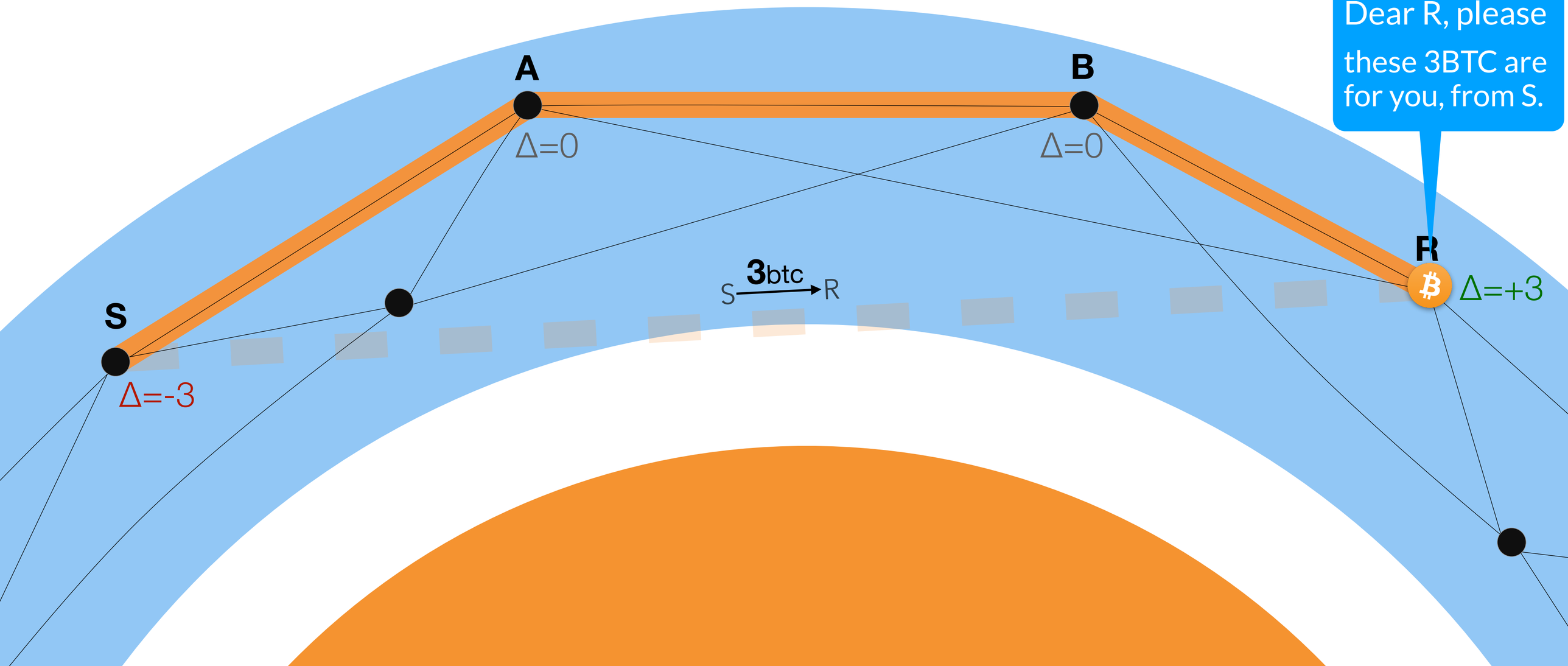
2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism



2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism

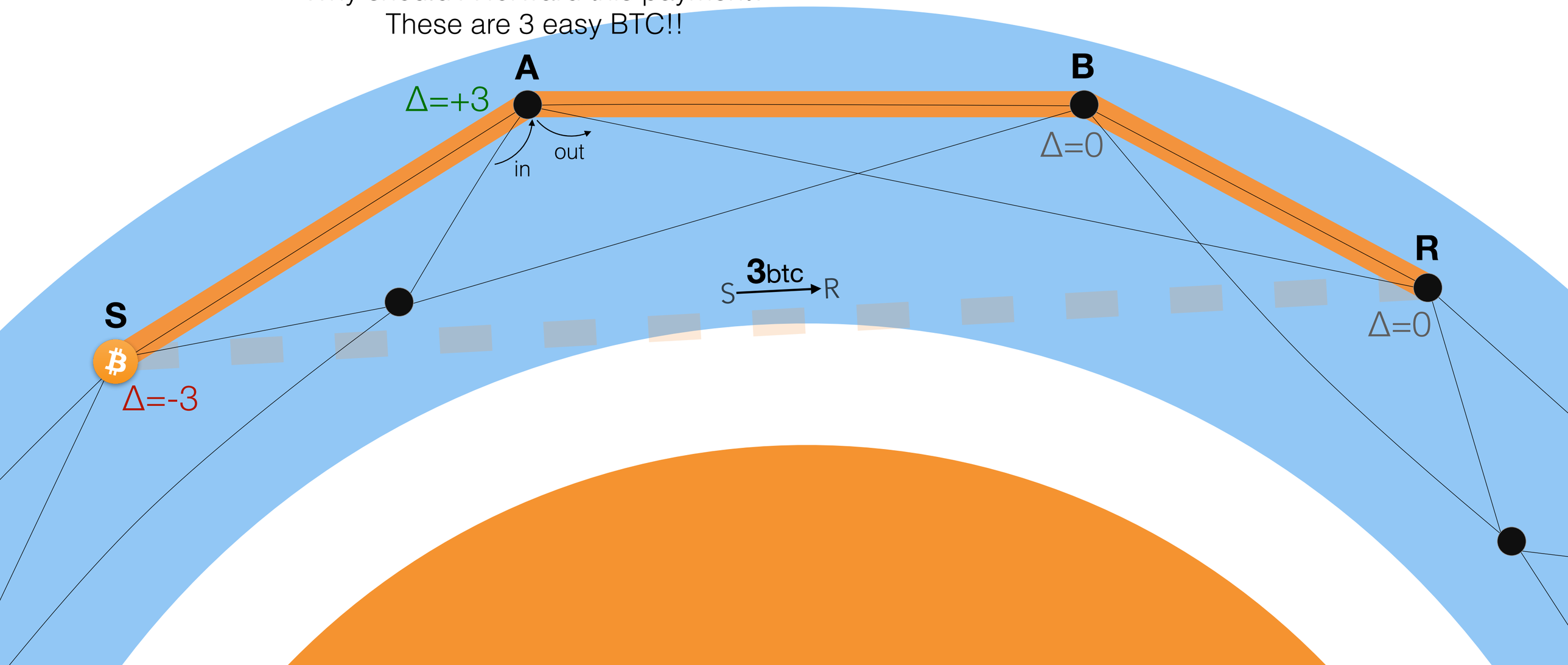


2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism



2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism

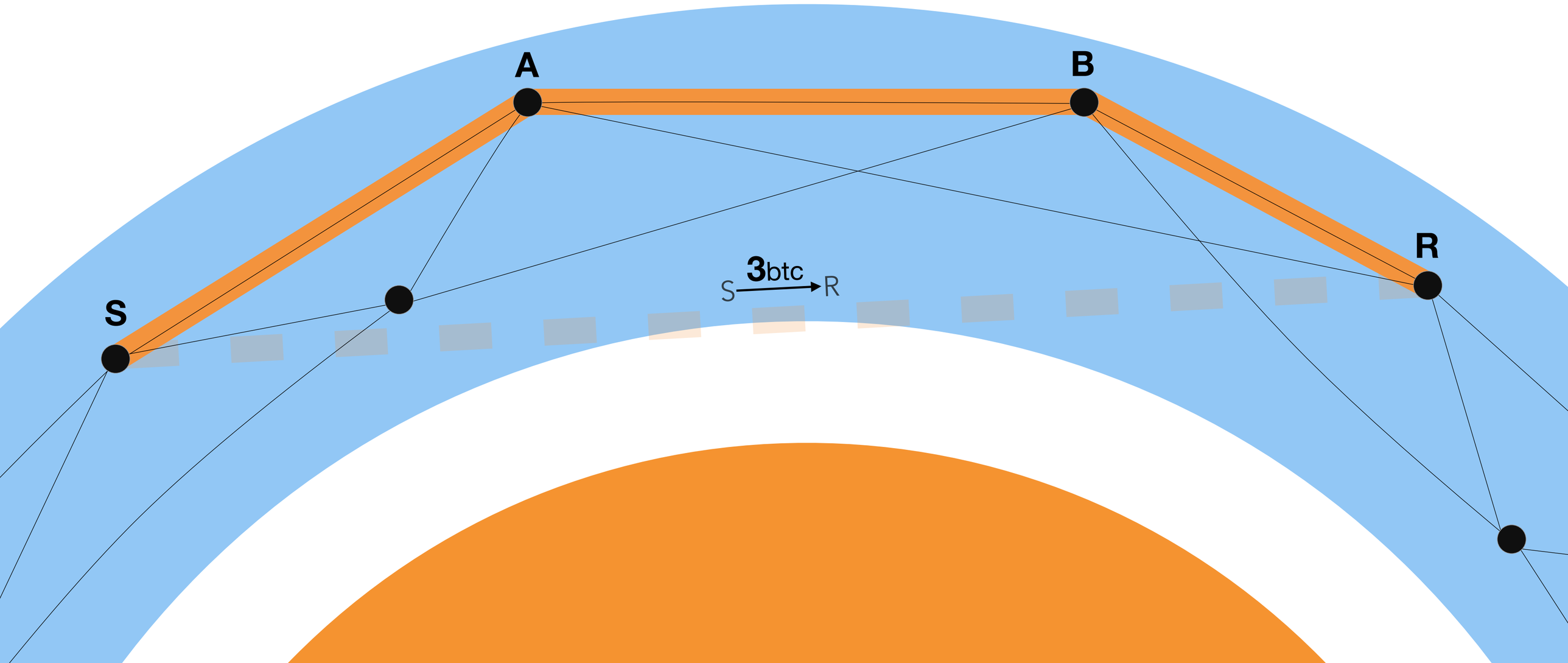
Why should A forward this payment?
These are 3 easy BTC!!



2nd ingredient: Trust-less Multi-Hop Forwarding Mechanism

A way to rewrite those (smart) contracts in such a way that **intermediate non-trusted nodes** have a **vested interest** in correctly **forwarding** the payments:

Hashed Time-Locked Contract (HTLC)



2nd ingredient: Hashed Time-Locked Contract (HTLC)

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

HTLC smart contract

+3
+0.2

-3
-0.2

A

B

R

S

3 h

s 1

h=f(s) 2

2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

(source-based) routing information

+3 +0.2 4

A

B

R

S

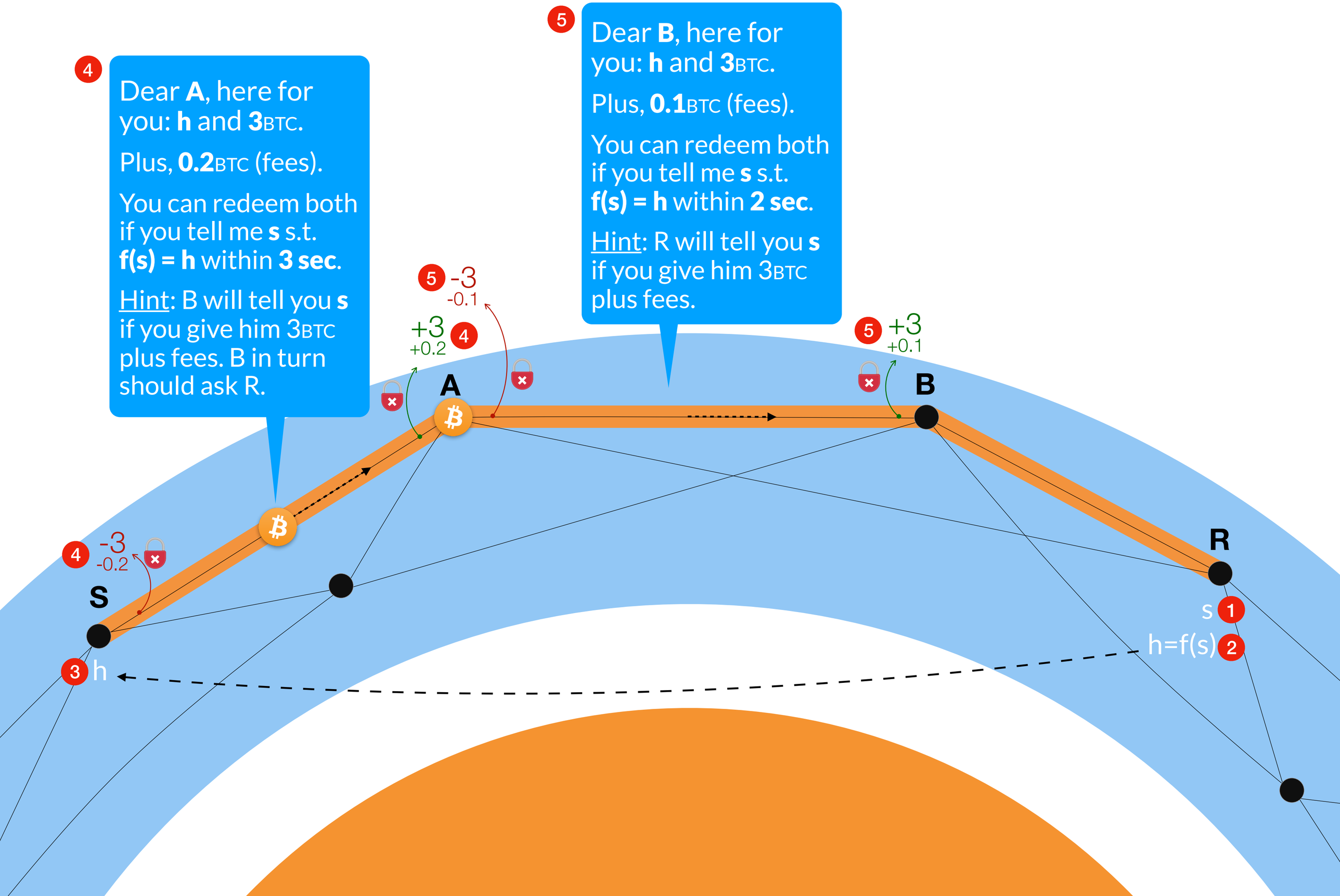
4 -3 -0.2

s 1

h=f(s) 2

3 h

2nd ingredient: Hashed Time-Locked Contract (HTLC)



2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

5 Dear **B**, here for you: **h** and **3**BTC.

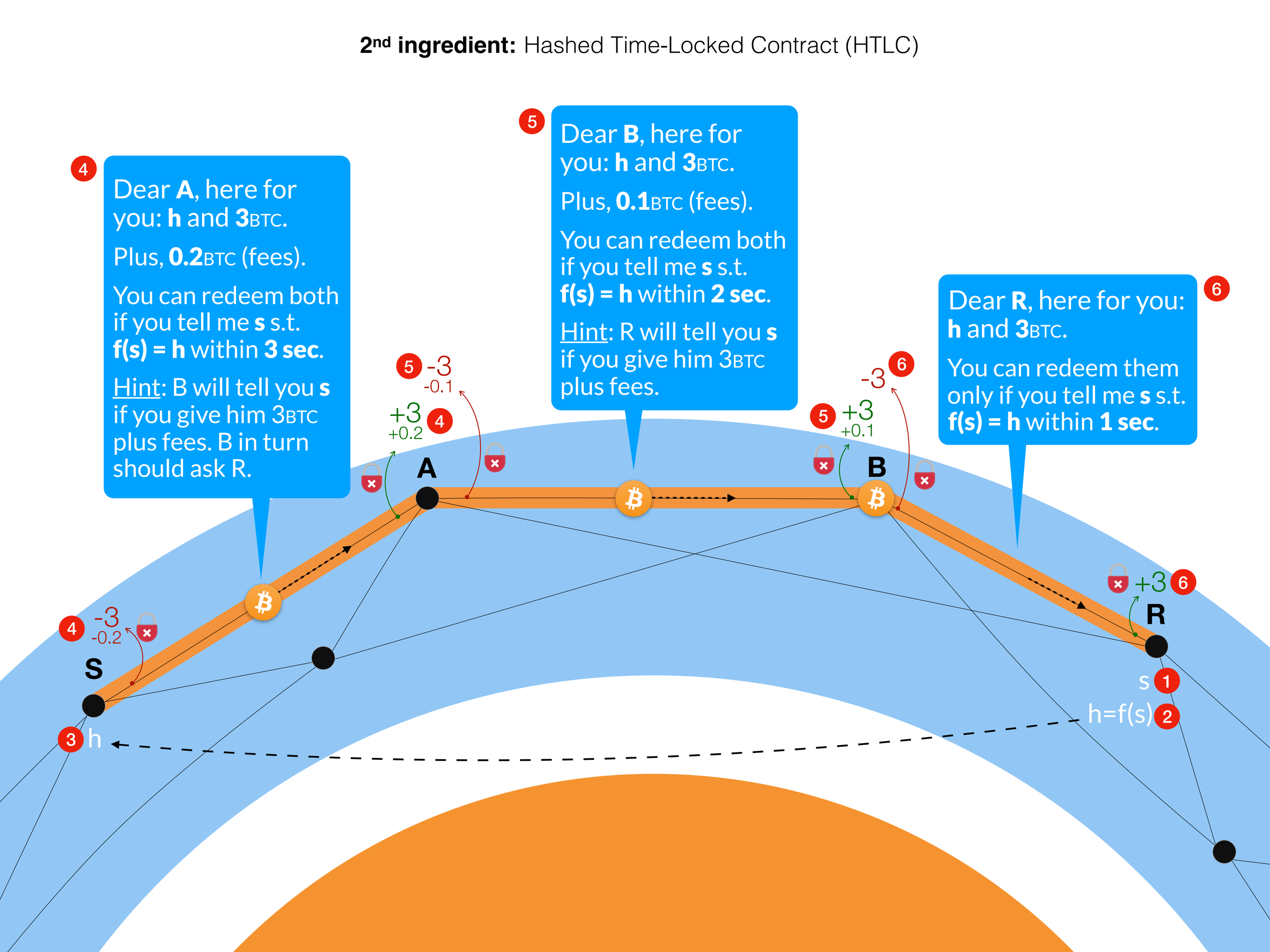
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

6 Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

5 Dear **B**, here for you: **h** and **3**BTC.

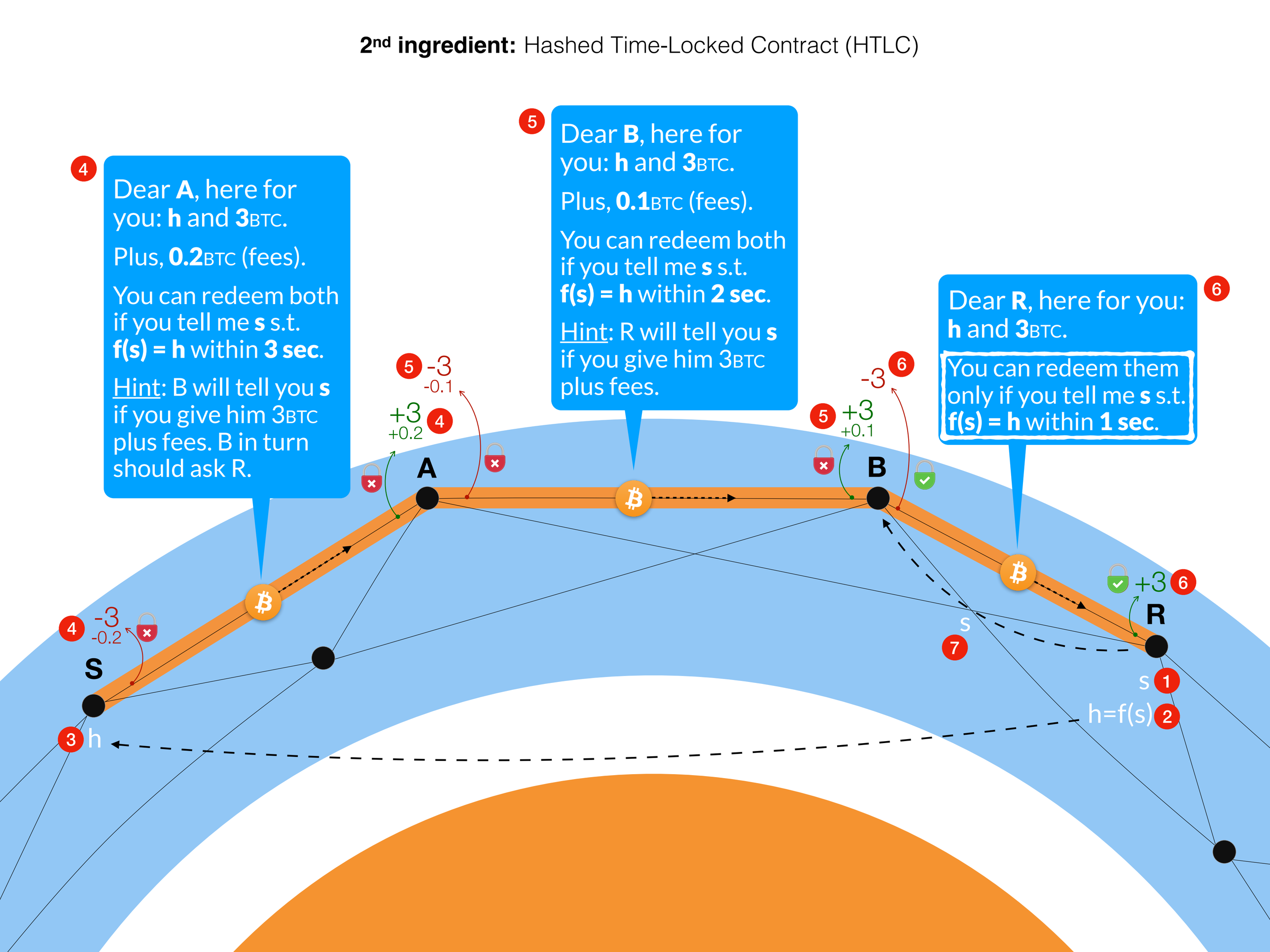
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

6 Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

5 Dear **B**, here for you: **h** and **3**BTC.

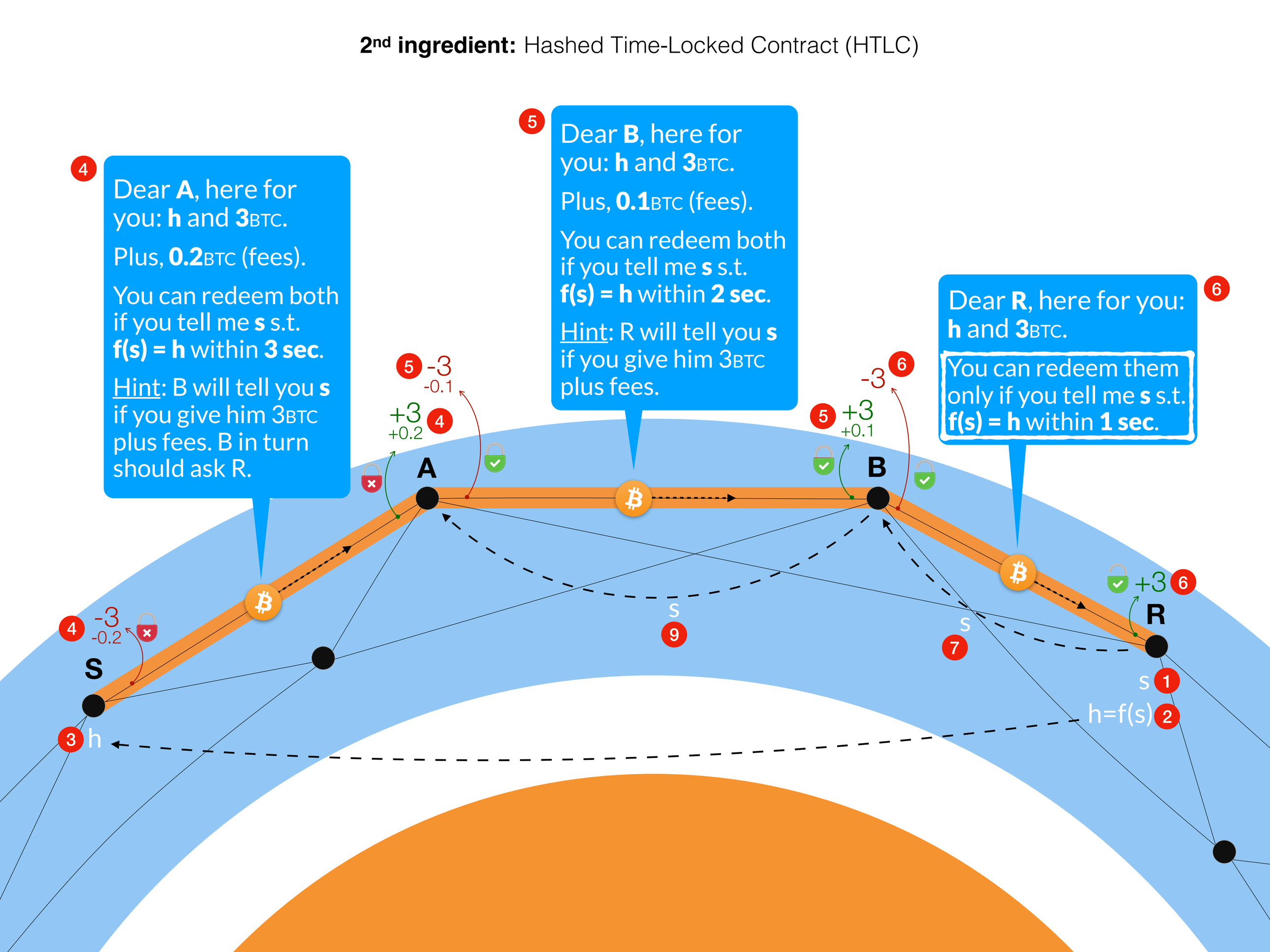
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

6 Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

5 Dear **B**, here for you: **h** and **3**BTC.

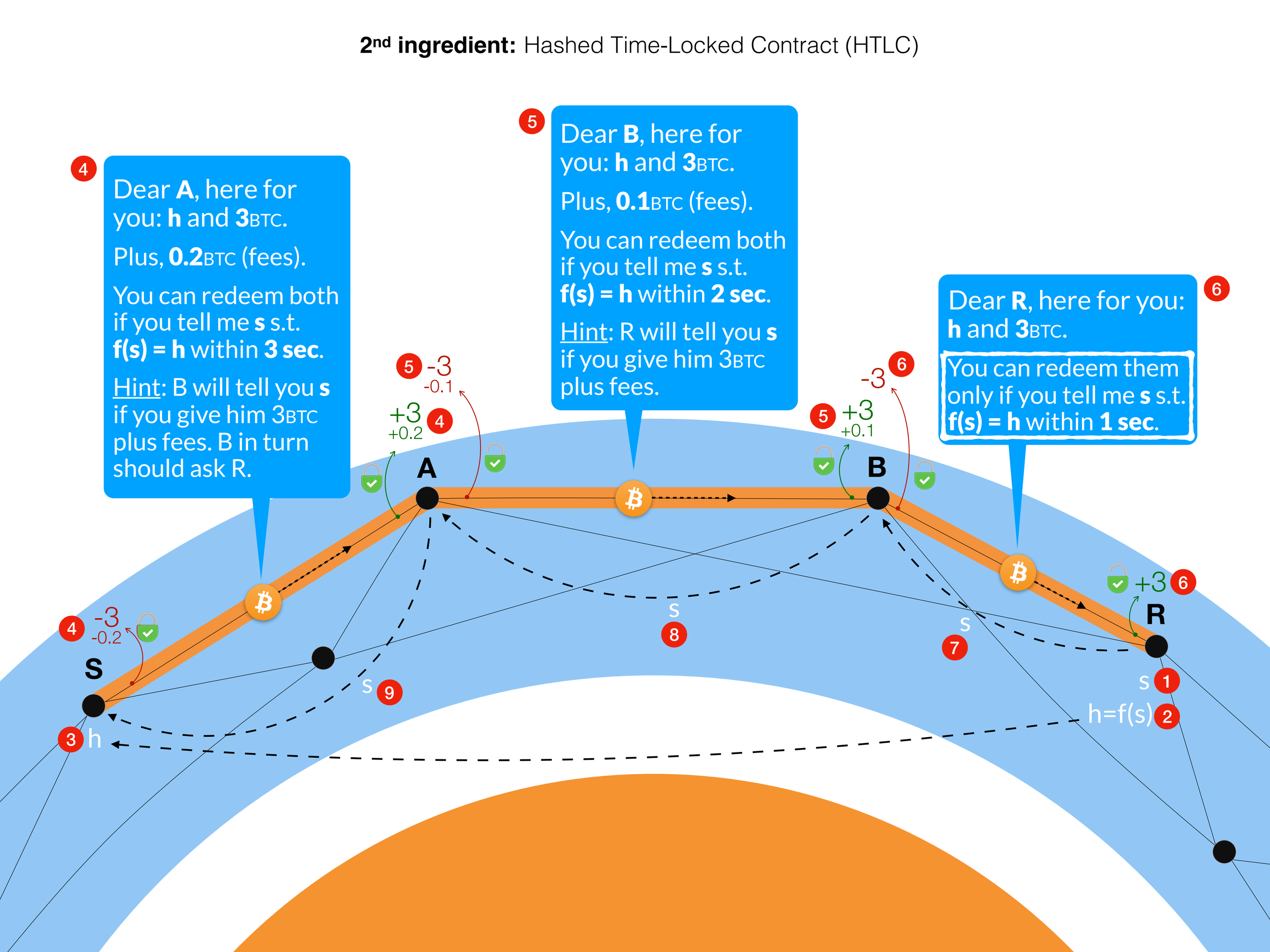
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

6 Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



2nd ingredient: Hashed Time-Locked Contract (HTLC)

4 Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

5 Dear **B**, here for you: **h** and **3**BTC.

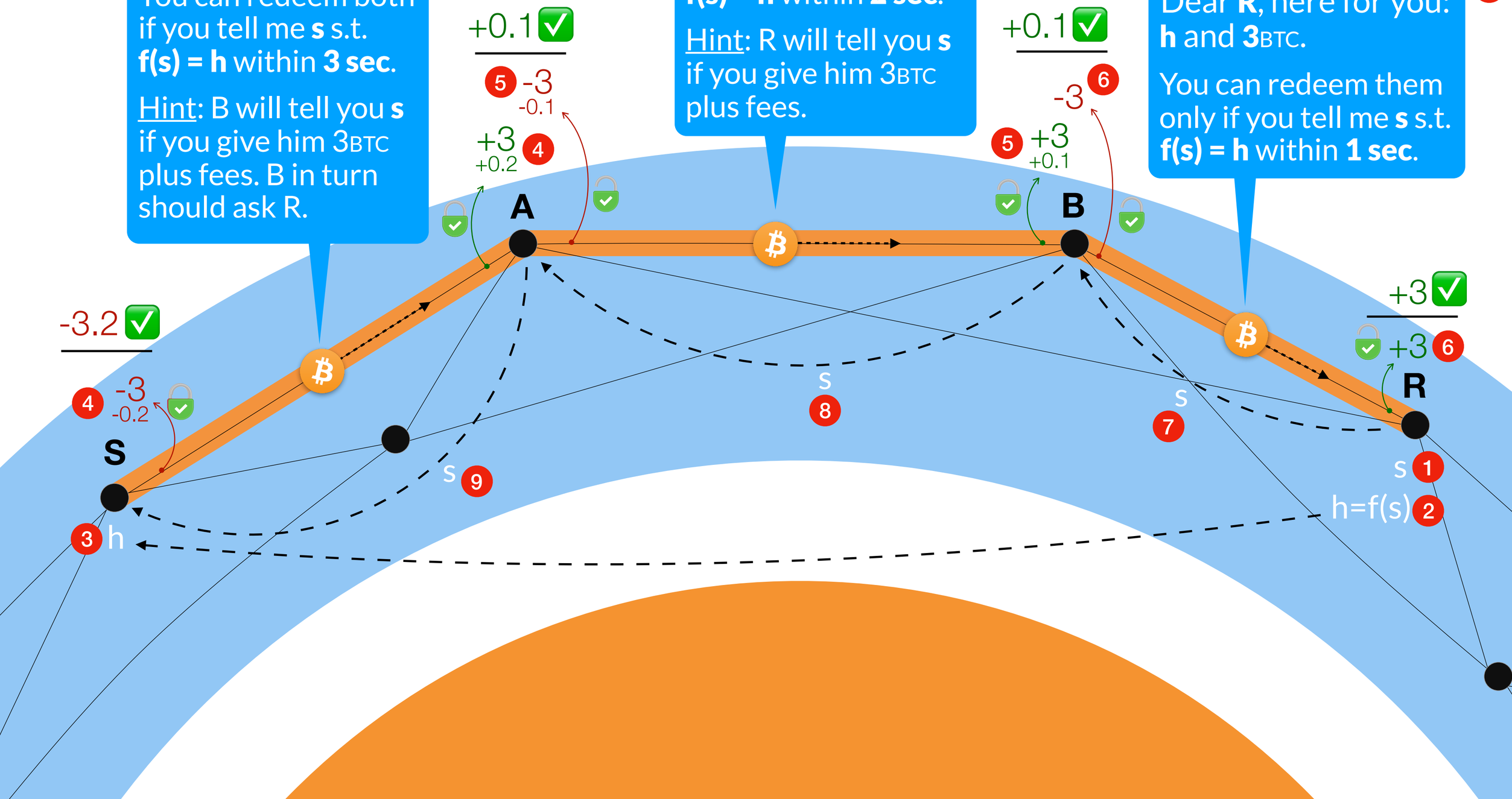
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

6 Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Hint: B will tell you **s** if you give him 3BTC plus fees. B in turn should ask R.

source-based **routing** information

Dear **B**, here for you: **h** and **3**BTC.

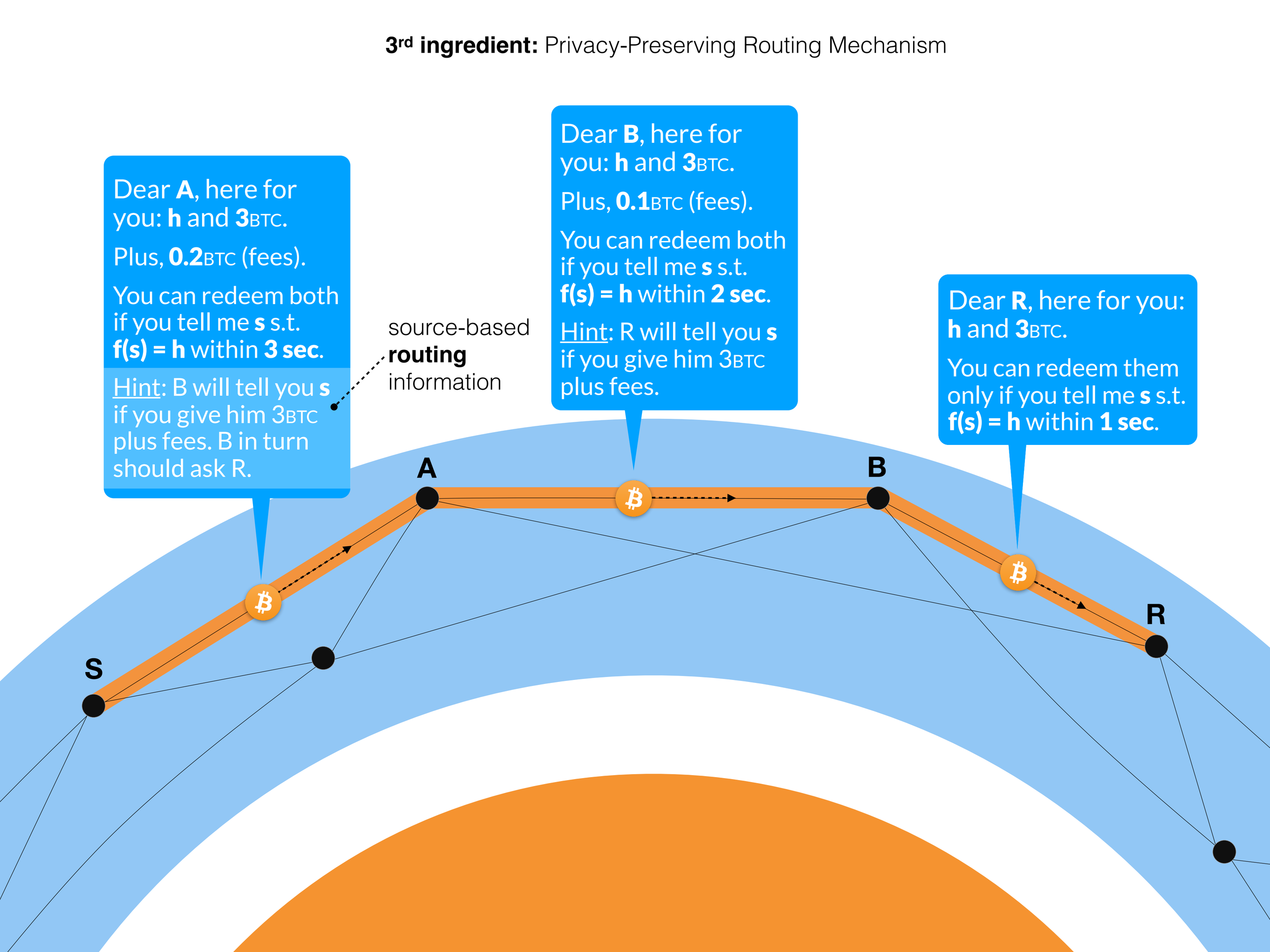
Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Hint: R will tell you **s** if you give him 3BTC plus fees.

Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.



3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:

for A: ask B about **s**
for B: ask R about **s**
for R: destination

source-based **routing** information

Dear **B**, here for you: **h** and **3**BTC.

Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Routing:

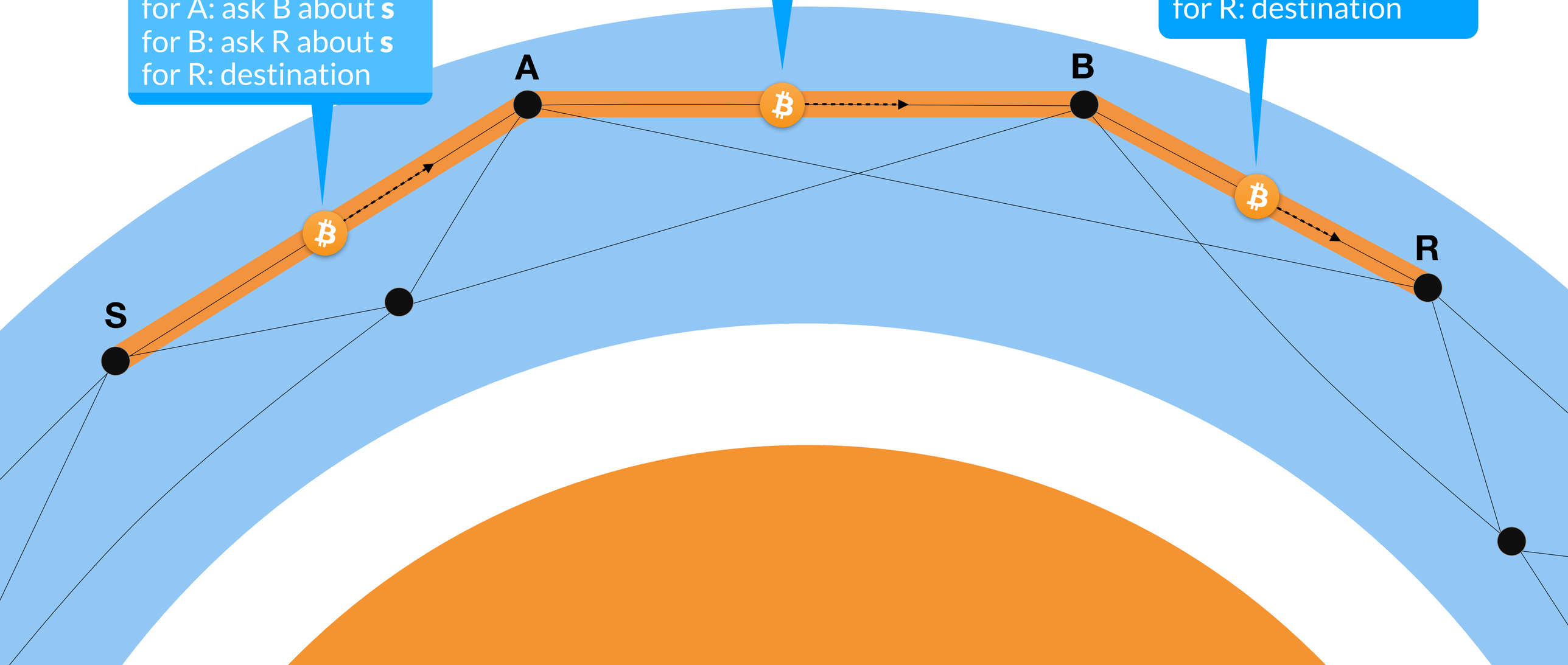
for B: ask R about **s**
for R: destination

Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.

Routing:

for R: destination



3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:

for A: ask B about **s**
for B: ask R about **s**

k_R { G348fw7oeGG1RzZ4wmJphUEwxxsfQlDov
MOISexKTbRpd6a+/NnJkHmqPZu/Y5O++F

Dear **B**, here for you: **h** and **3**BTC.

Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Routing:

for B: ask R about **s**
for R: destination

Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.

Routing:

for R: destination

S

A

B

R

k_A $k_A^{private}$

k_B $k_B^{private}$

k_R $k_R^{private}$

3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:

for A: ask B about **s**

k_B { lg8P2aS6ExODxtODlR/bTLikYb9V3S20+Os
k_R { yJSHwsTb673YDVo0RI3BQ9xaiLnM8+9YvVl
DgzYeDETC4xsRMjRogbl/rXa2N9VOwpdRbg
Vxm49aC8UTduCOYU2UbOrIz9AOrLgwfhn3f

Dear **B**, here for you: **h** and **3**BTC.

Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Routing:

for B: ask R about **s**
for R: destination

Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.

Routing:

for R: destination

S

A

B

R

k_A $k_A^{private}$

k_B $k_B^{private}$

k_R $k_R^{private}$

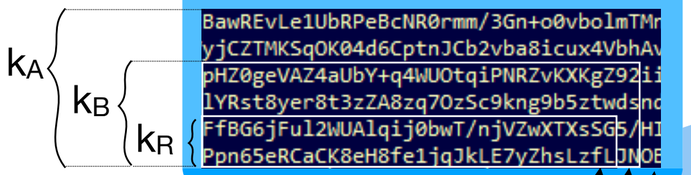
3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:



Dear **B**, here for you: **h** and **3**BTC.

Plus, **0.1**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **2 sec**.

Routing:

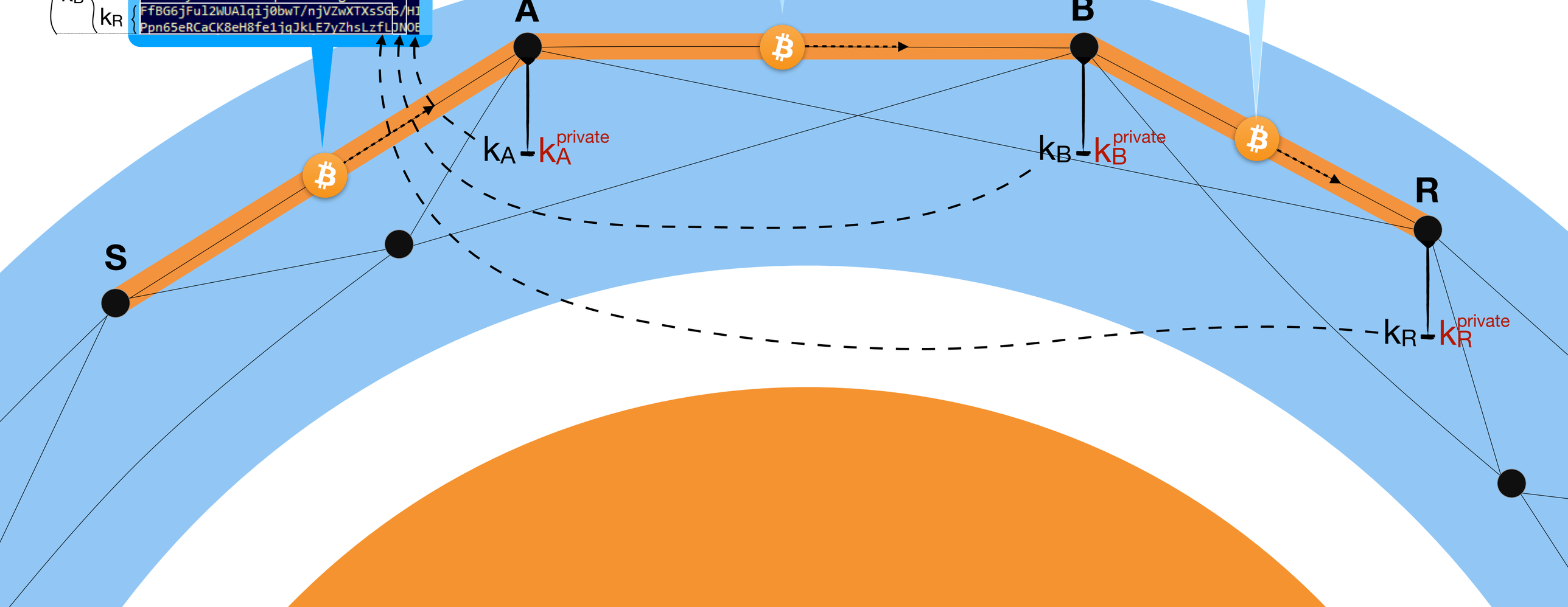
for B: ask R about **s**
for R: destination

Dear **R**, here for you: **h** and **3**BTC.

You can redeem them only if you tell me **s** s.t. **f(s) = h** within **1 sec**.

Routing:

for R: destination



3rd ingredient: Privacy-Preserving Routing Mechanism

Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:

k_A { BawREvLe1UbRPeBcNR0rmm/3Gn+o0vbolmTMr
yjCZTMKSqOK04d6CptnJCb2vba8icux4VbhAv
k_B { pHZ0geVAZ4aUbY+q4WUOtqiPNRZvKXKgZ92ii
k_R { lYRst8yer8t3zZA8zq7OzSc9kng9b5ztwdsnc
FfBG6jFul2WUAlqij0bwT/njVZwXTXsSG5/HI
Ppn65eRCaCK8eH8fe1jqJkLE7yZhsLzfLJNOE

A

B

S

R

k_A $k_A^{private}$

k_B $k_B^{private}$

k_R $k_R^{private}$

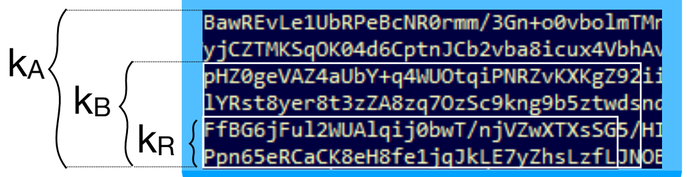
3rd ingredient: Privacy-Preserving Routing Mechanism

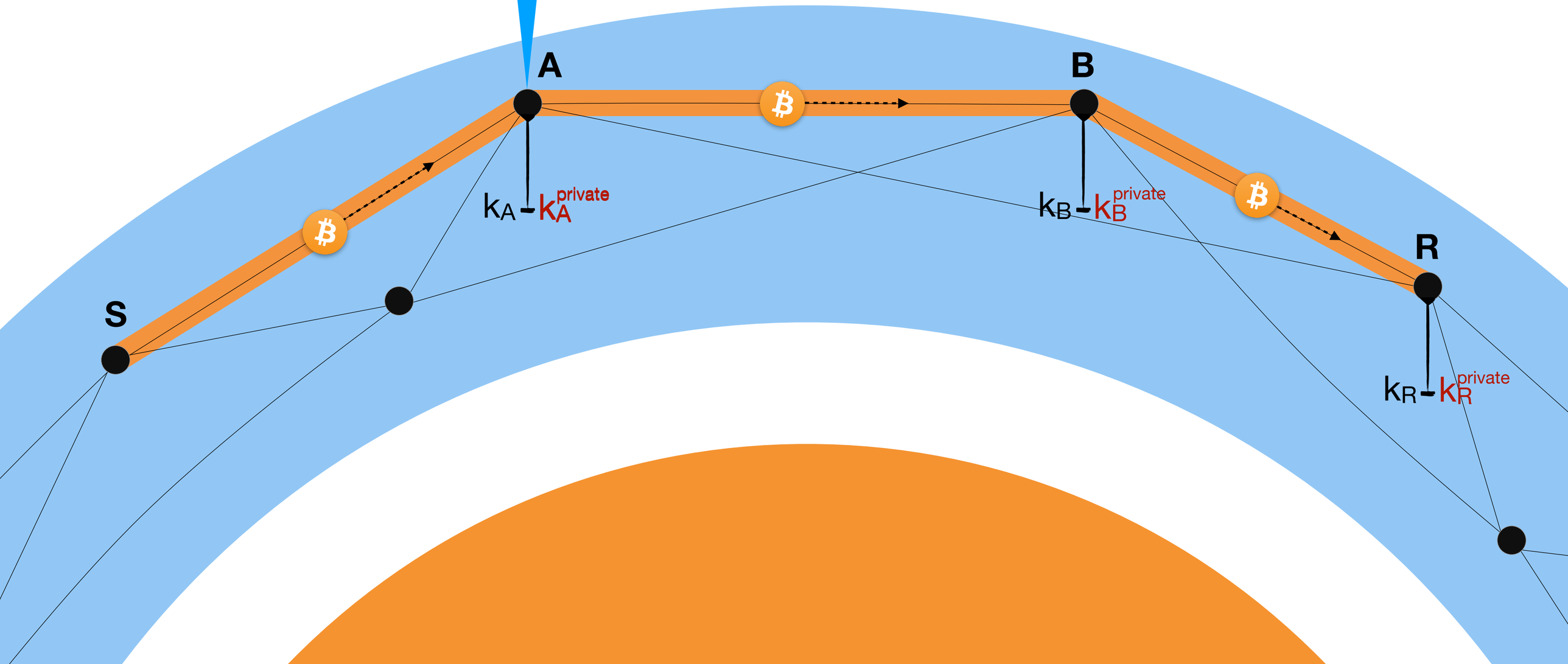
Dear **A**, here for you: **h** and **3**BTC.

Plus, **0.2**BTC (fees).

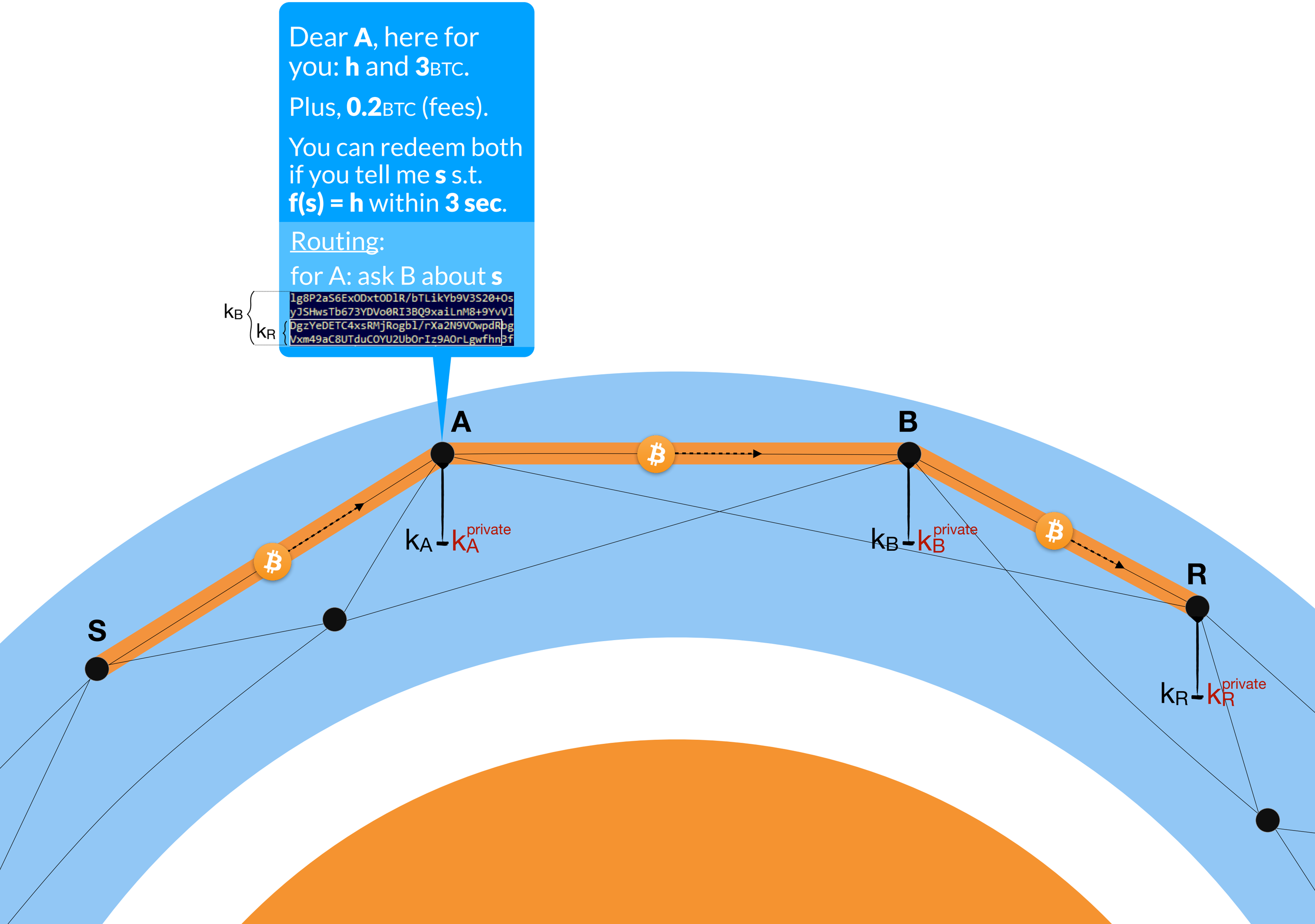
You can redeem both if you tell me **s** s.t. **f(s) = h** within **3 sec**.

Routing:

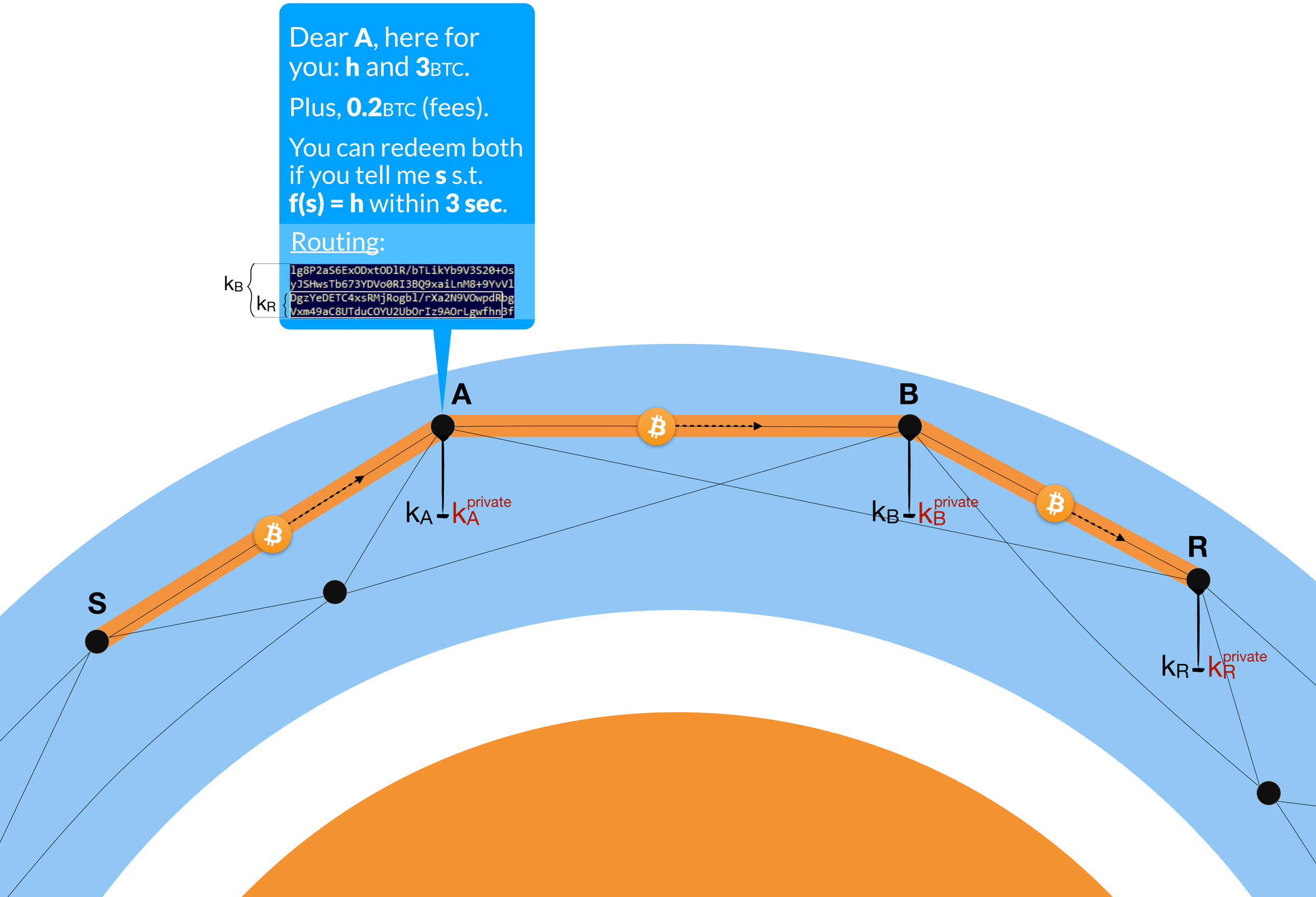




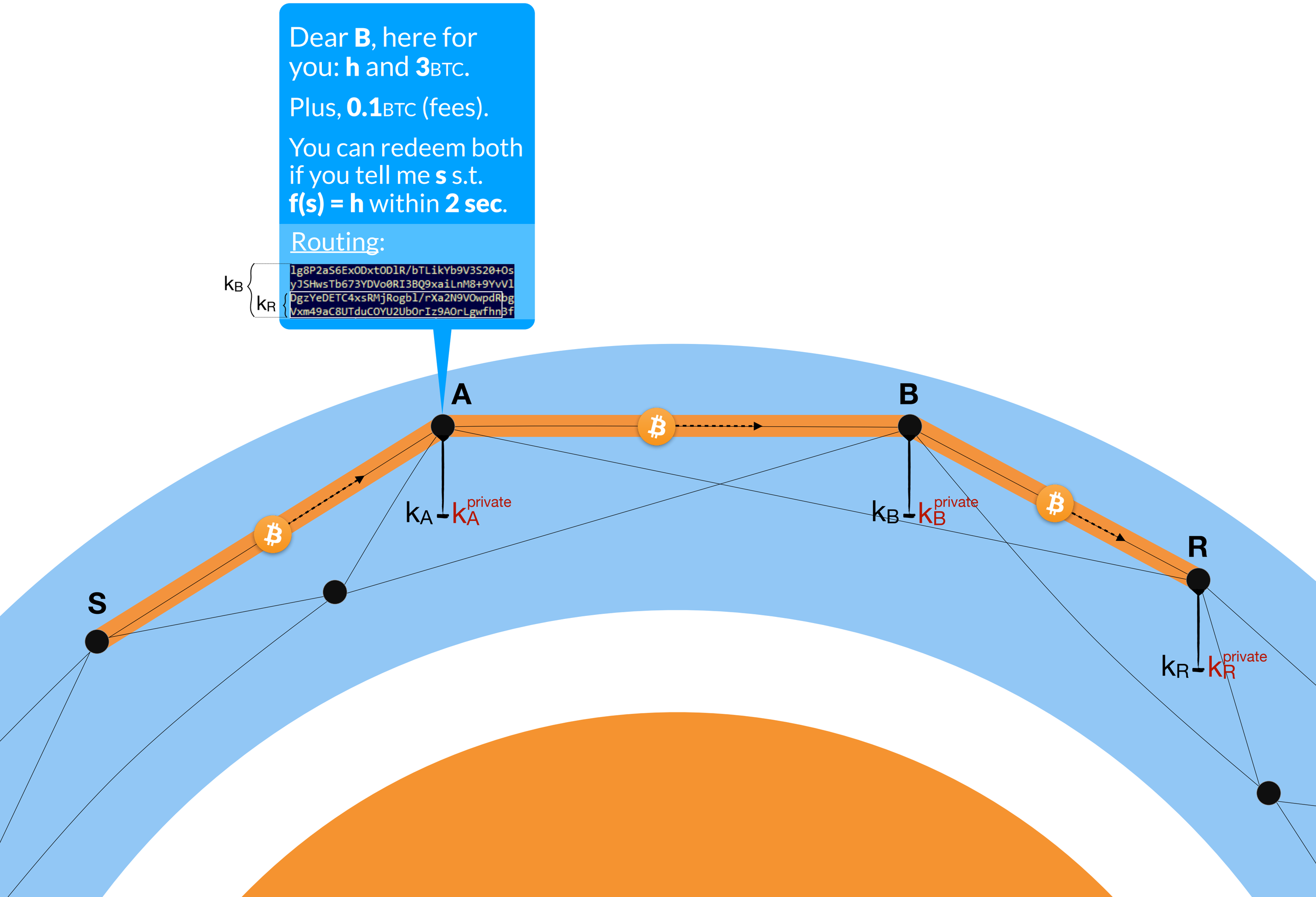
3rd ingredient: Onion Routing Mechanism



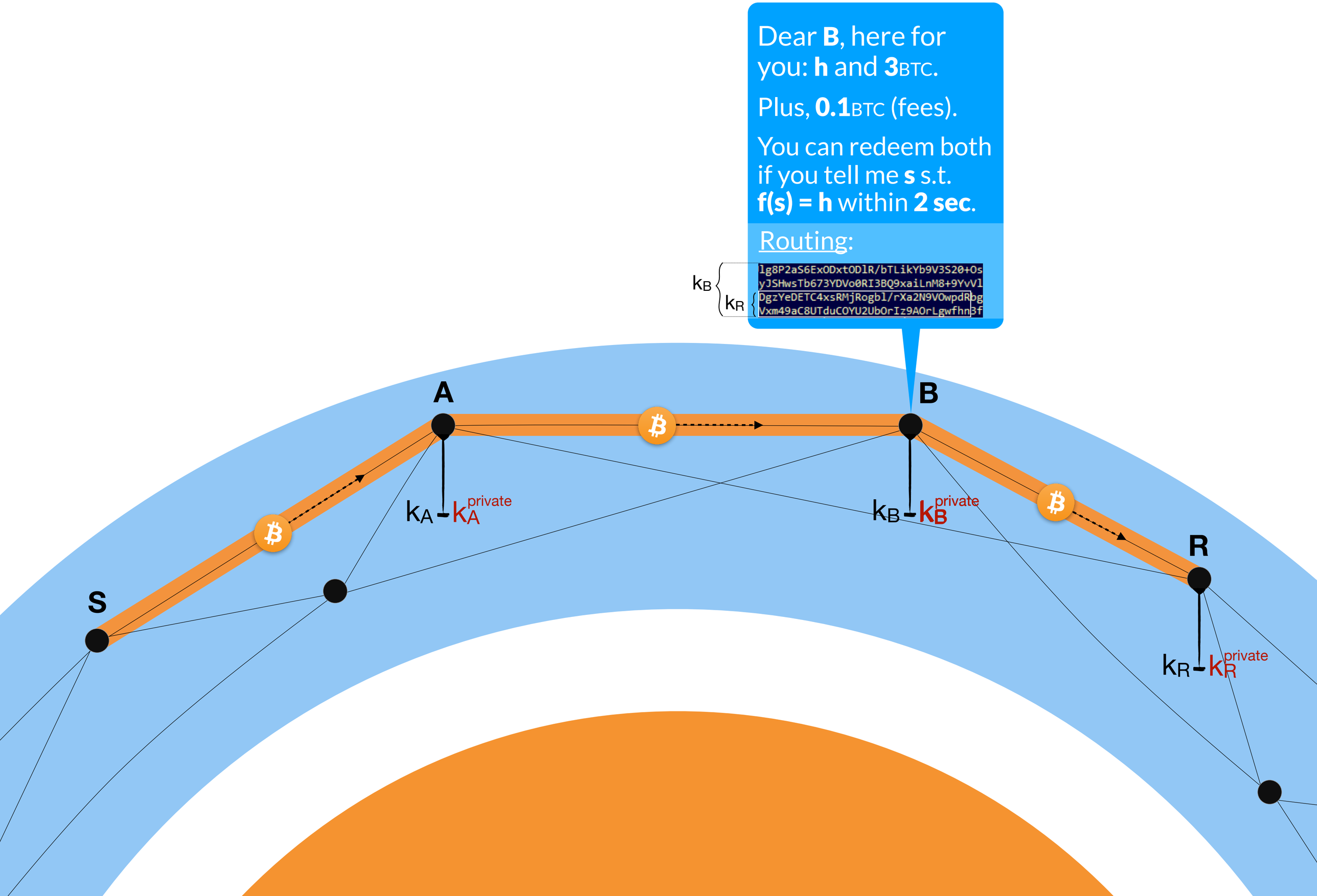
3rd ingredient: Onion Routing Mechanism



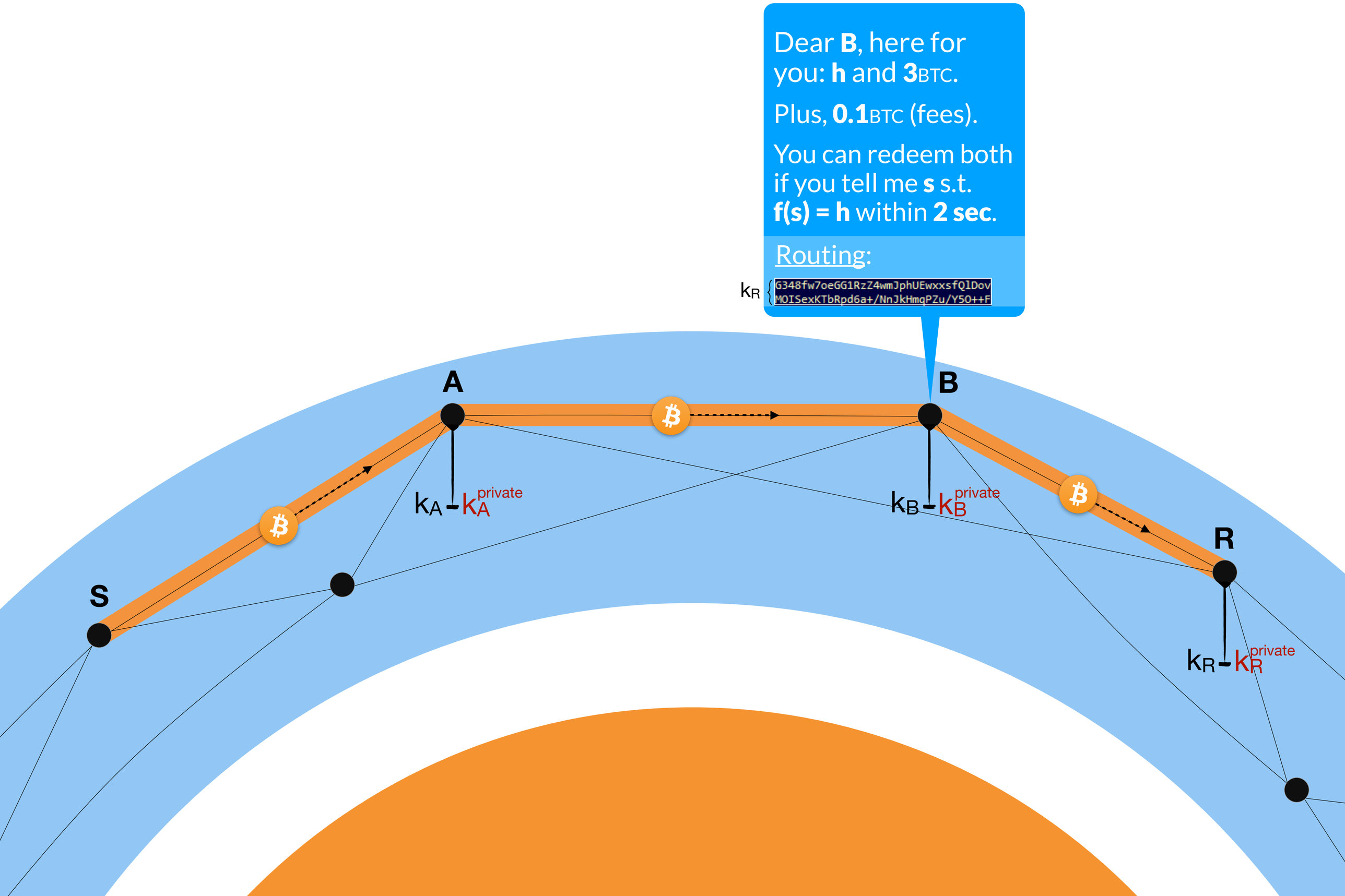
3rd ingredient: Onion Routing Mechanism



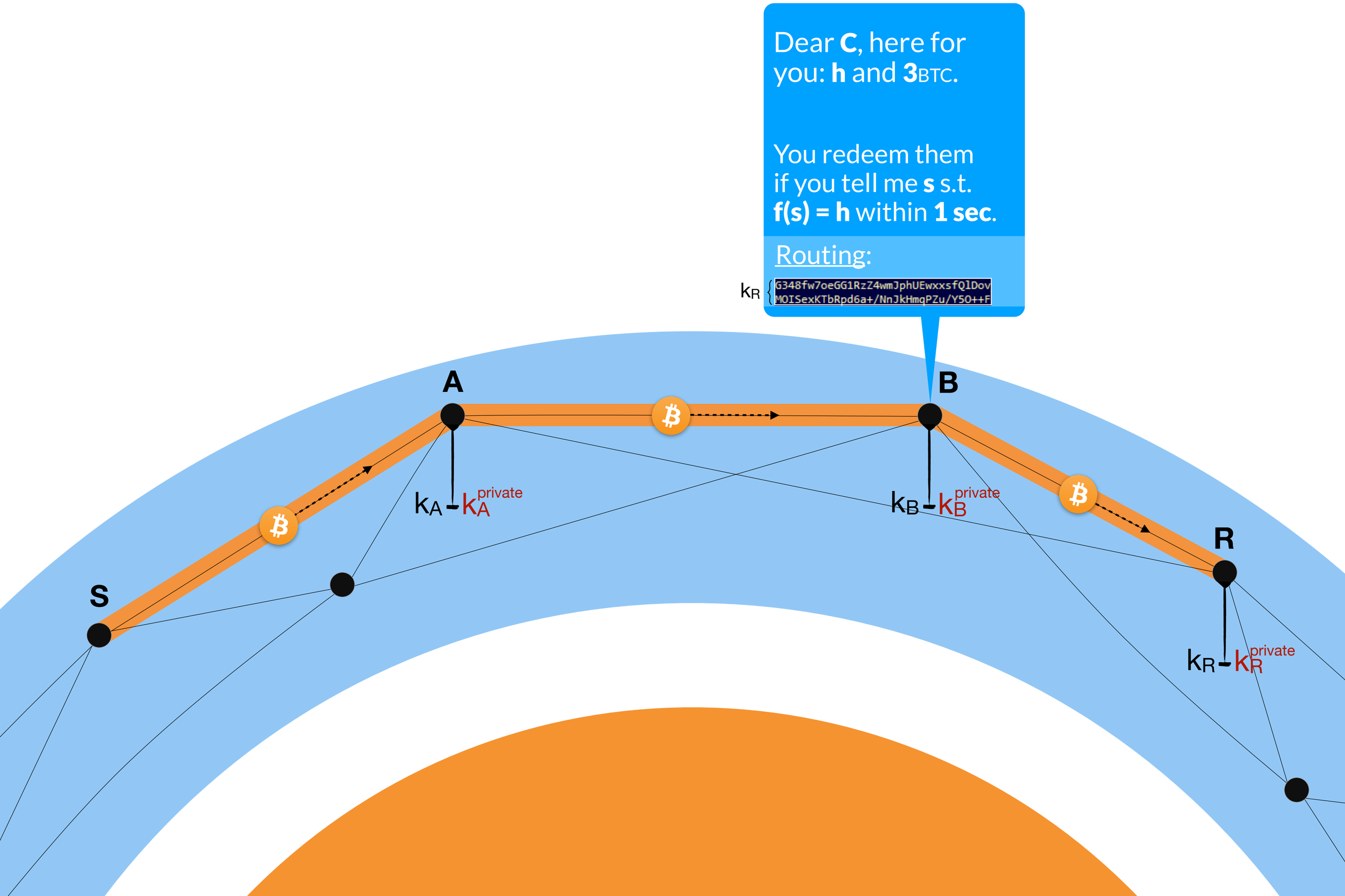
3rd ingredient: Onion Routing Mechanism



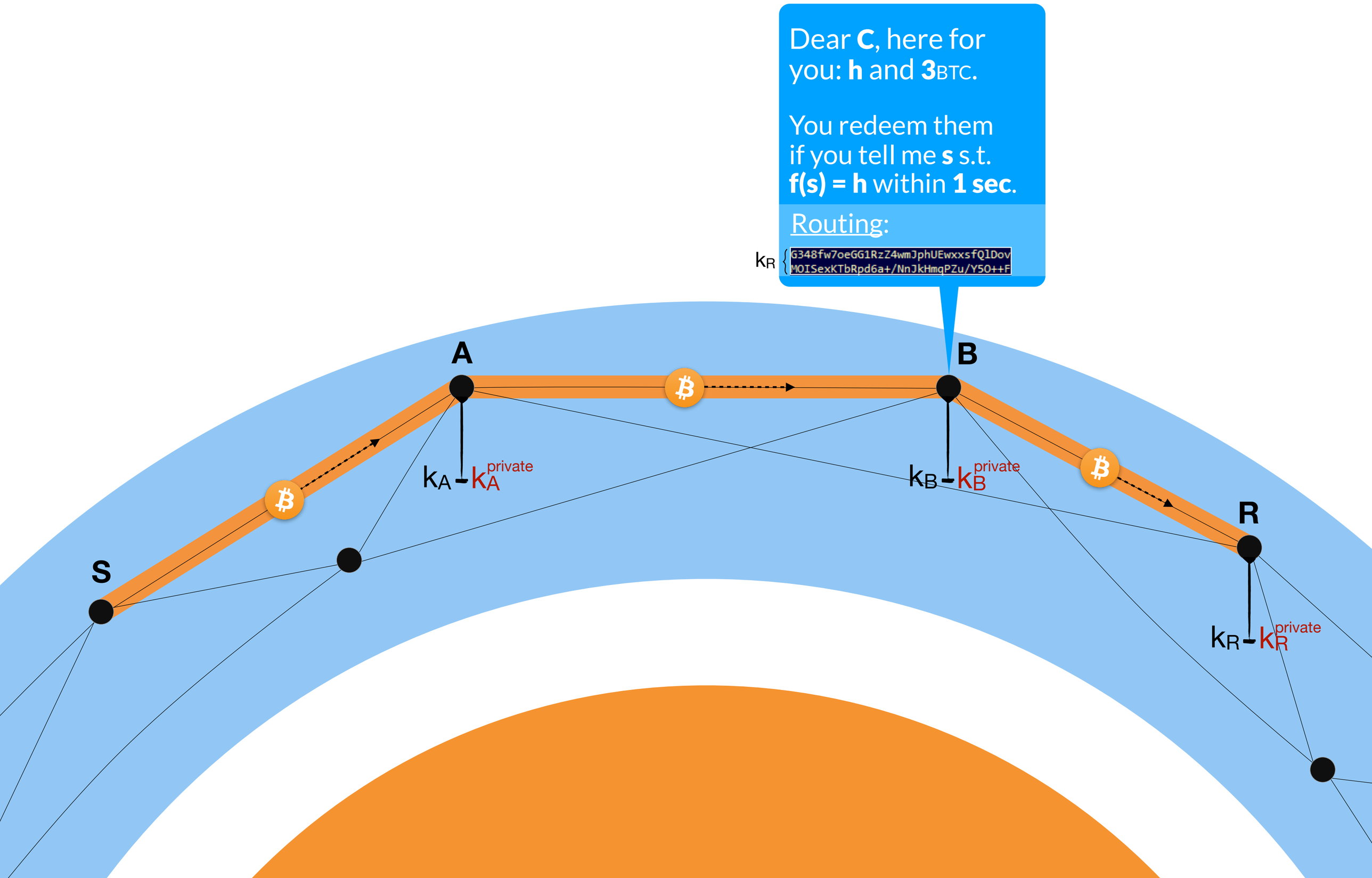
3rd ingredient: Onion Routing Mechanism



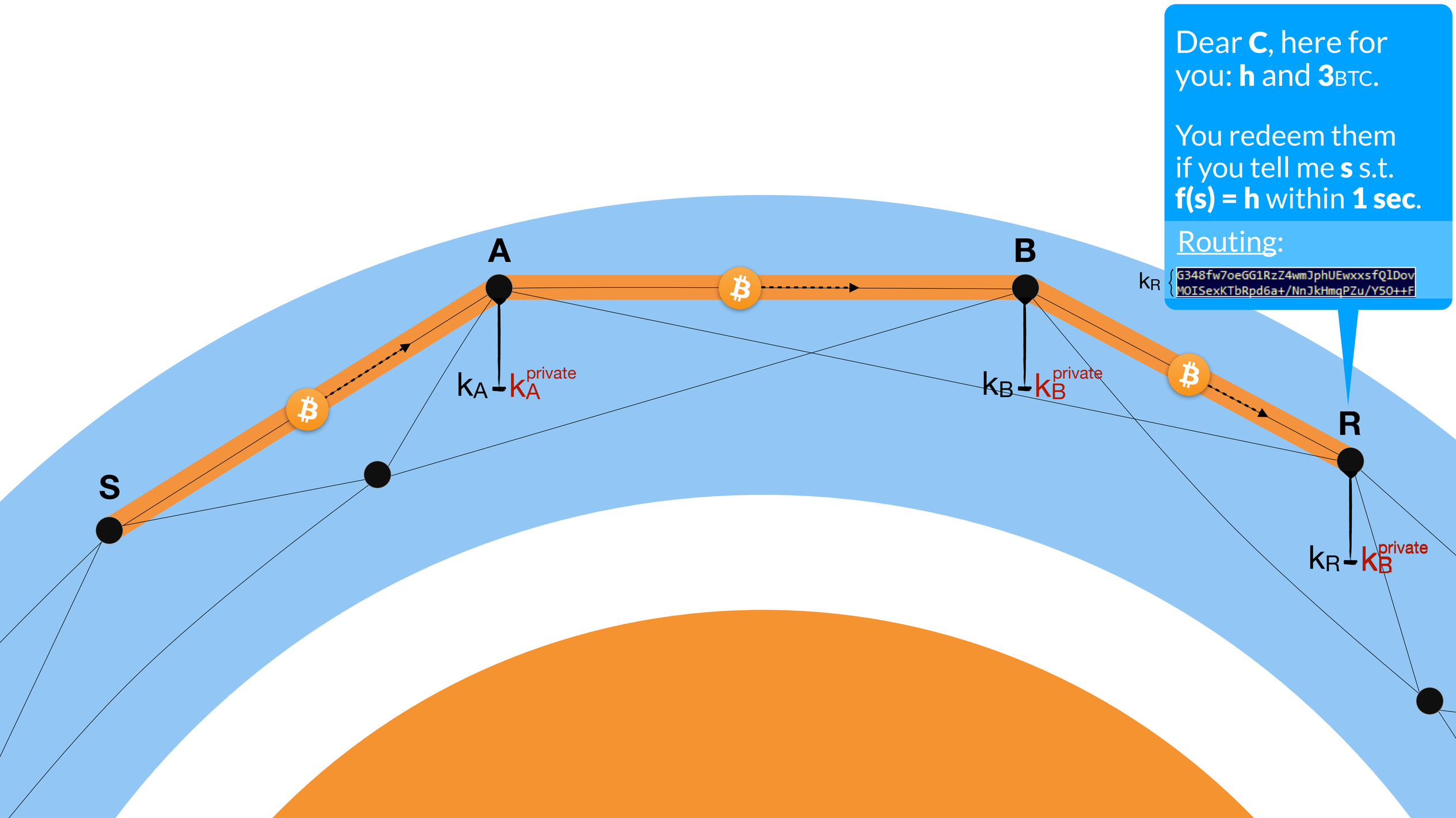
3rd ingredient: Onion Routing Mechanism



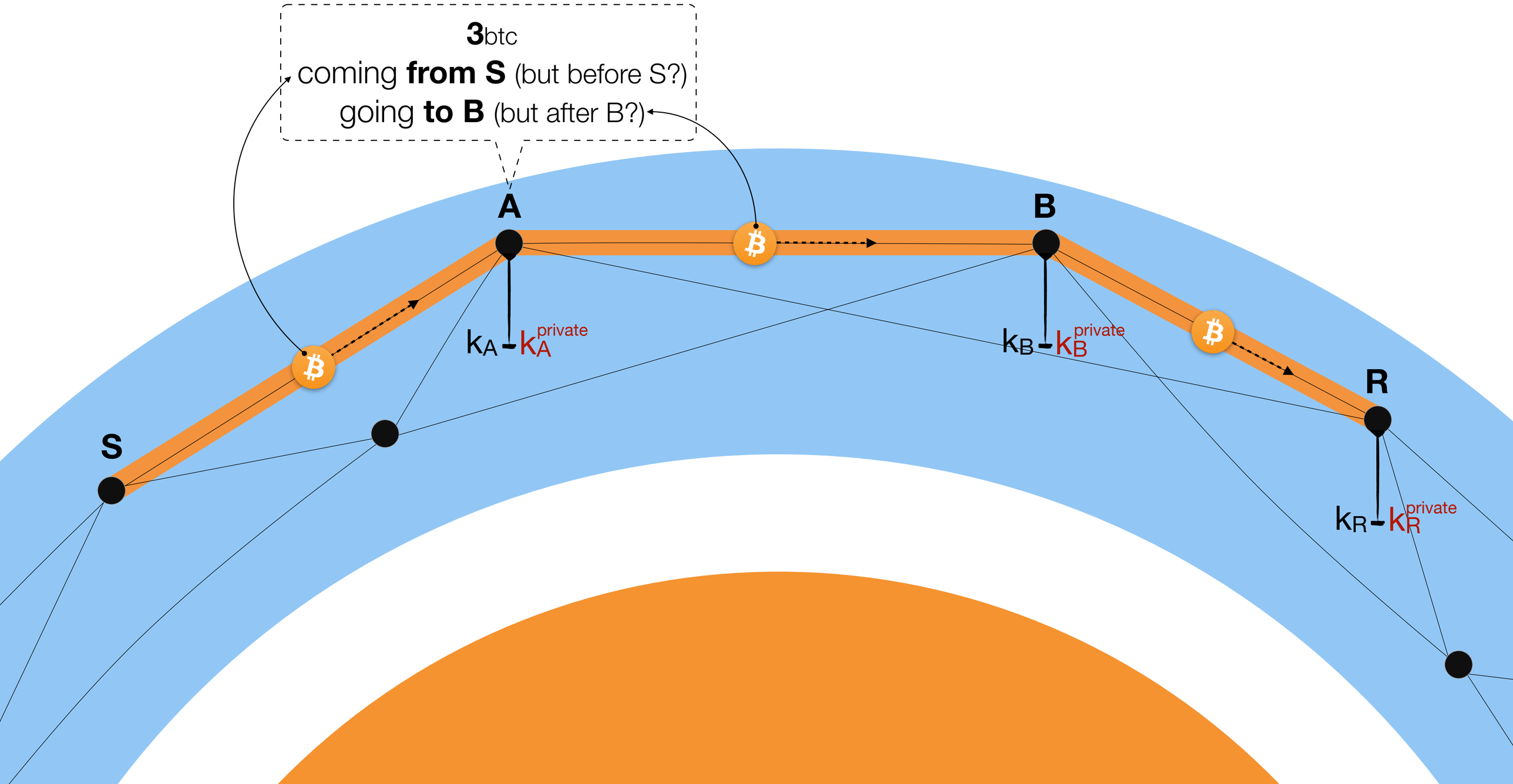
3rd ingredient: Onion Routing Mechanism



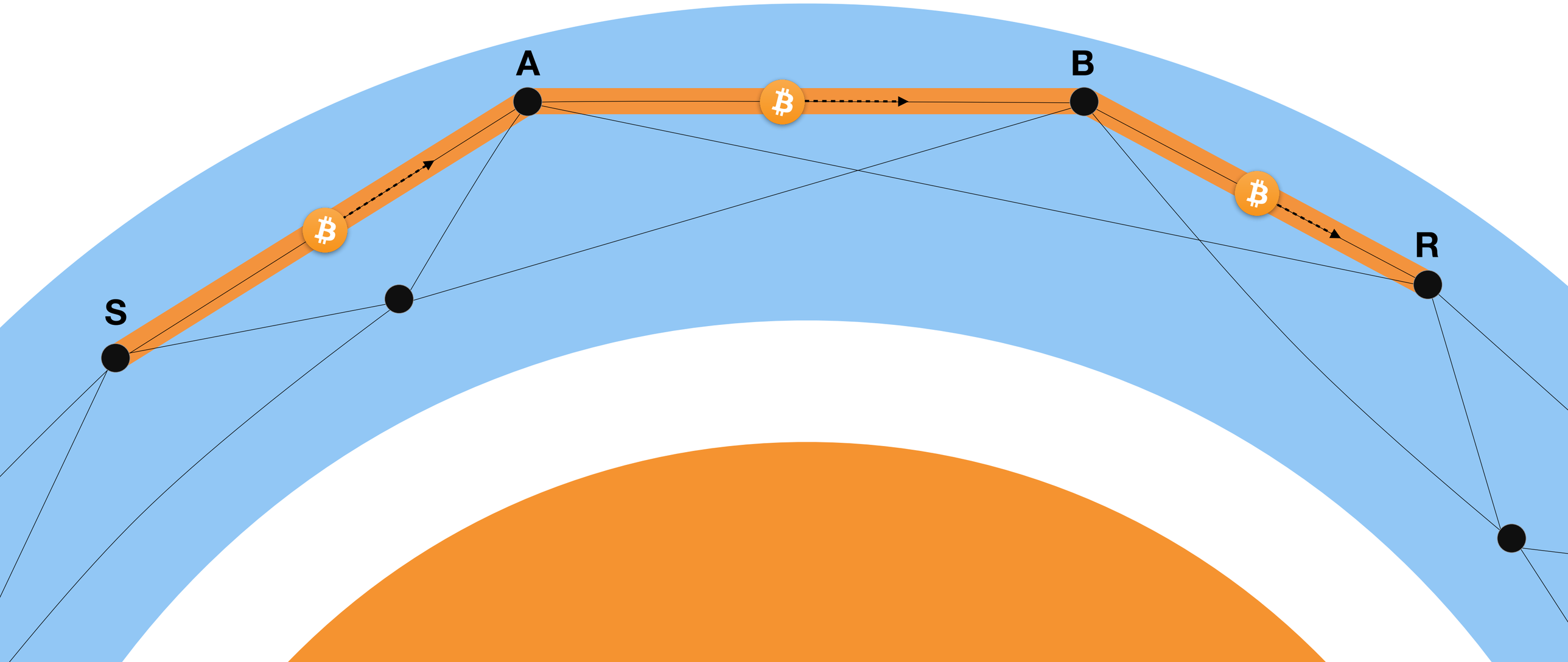
3rd ingredient: Onion Routing Mechanism

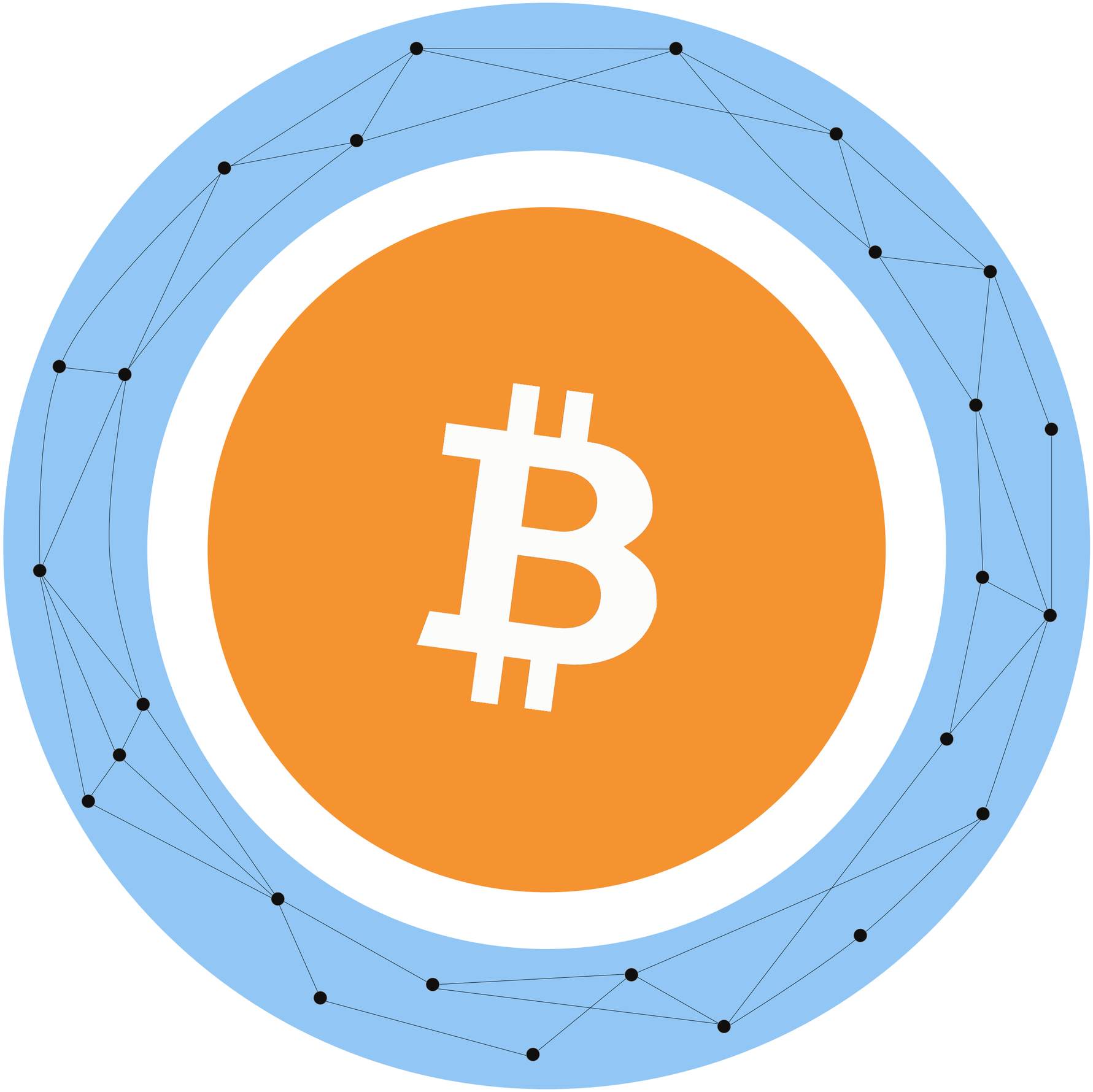


3rd ingredient: Onion Routing Mechanism



3rd ingredient: Onion Routing Mechanism





Lightning Network:

Cryptography to the Rescue of Cryptography?

Marco **Benedetti**, Roberto **Favaroni**,
Giuseppe **Galano,** Andrea **Gentili**,
Davide **Magnanimi**, Michela **Santangelo***

[NAME].[SURNAME]@bancaditalia.it

ART
www.bankit.art

*Intern at ART

Thank you for your attention

Any questions?

BANCA D'ITALIA
EUROSISTEMA

The **opinions** expressed and conclusions drawn are those of the **authors** and do **not** necessarily reflect the views of the **Bank of Italy**.