

  **CAR Pool & More** 

A Pragmatic Improvement to Sync Performance

github.com/fission-suite/spec/car-pool



The limitation of ***local knowledge***
is the ***fundamental fact***
about the setting in which we work,
and it is ***a very powerful limitation***

– Nancy Lynch, A Hundred Impossibility Proofs for Distributed Computing

Some peers are "**more peer**" **than others**
(especially in an operator context)

BROOKLYN ZELENKA

@expede
CTO @ Fission



JUSTIN JOHNSON

@justincjohnson
IPFS Engineer @ Fission



Won't have time to cover everything in depth.
Come talk to us after, or read the spec ;)

Caveat

Trusted, Point-to-Point IPLD Transfer

The Problem

Pain In Practice 

The Problem

Why?

The Problem

Why?

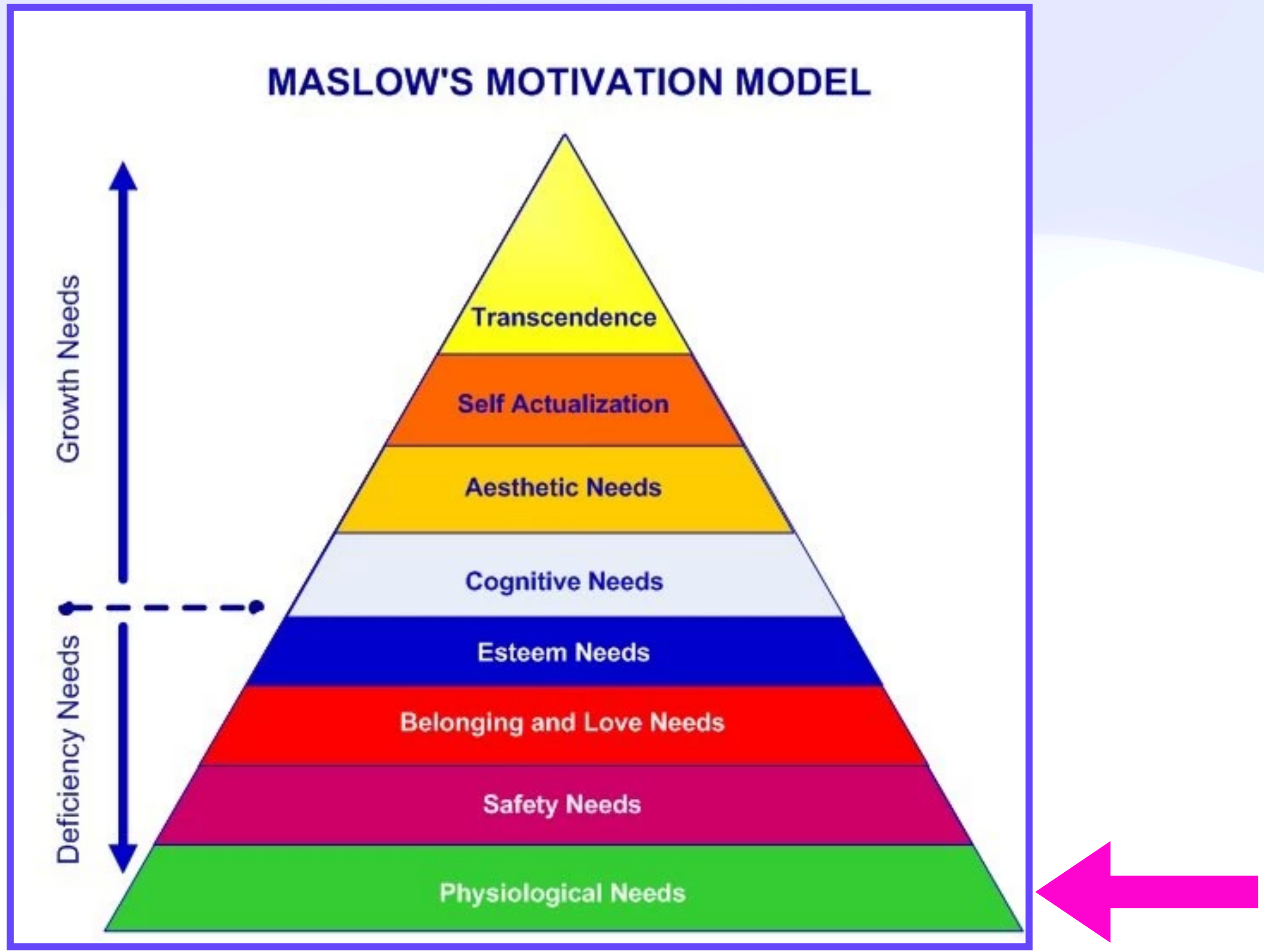
- ◆ We ♥ IPLD
- ◆ Fission has deeply nested, cross-linked data that
 - ◆ Cannot fit into browser storage
 - ◆ Benefits from deduplication
 - ◆ Has multiple writers
- ◆ Using mature, universally supported transports would be awesome
- ◆ "Light clients"
- ◆ Speed of light concerns

The Problem

Food. Water. Shelter.

The Problem

Food. Water. Shelter.

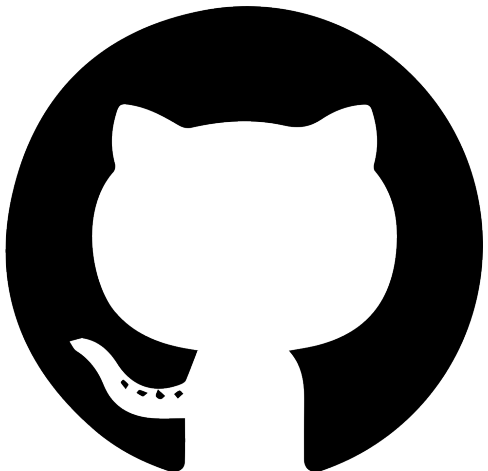


The Problem

Flakiness in Browsers & CLI & GitHub Actions

The Problem

Flakiness in Browsers & CLI & GitHub Actions



The Problem

Are You There? 📱

The Problem

Are You There? 📱

- ◆ Keeping nodes reliably connected in production is extremely difficult (browser, server, desktop, mobile)
 - ◆ Private networks don't make sense for our use case
 - ◆ (We'd love to share an operator node architecture wish list with any maintainers, though)

The Problem

Are You There? 📱

- ◆ Keeping nodes reliably connected in production is extremely difficult (browser, server, desktop, mobile)
 - ◆ Private networks don't make sense for our use case
 - ◆ (We'd love to share an operator node architecture wish list with any maintainers, though)
- ◆ Up/down over a HTTP Gateway is just LARPing decentralization 🌶️

The Problem

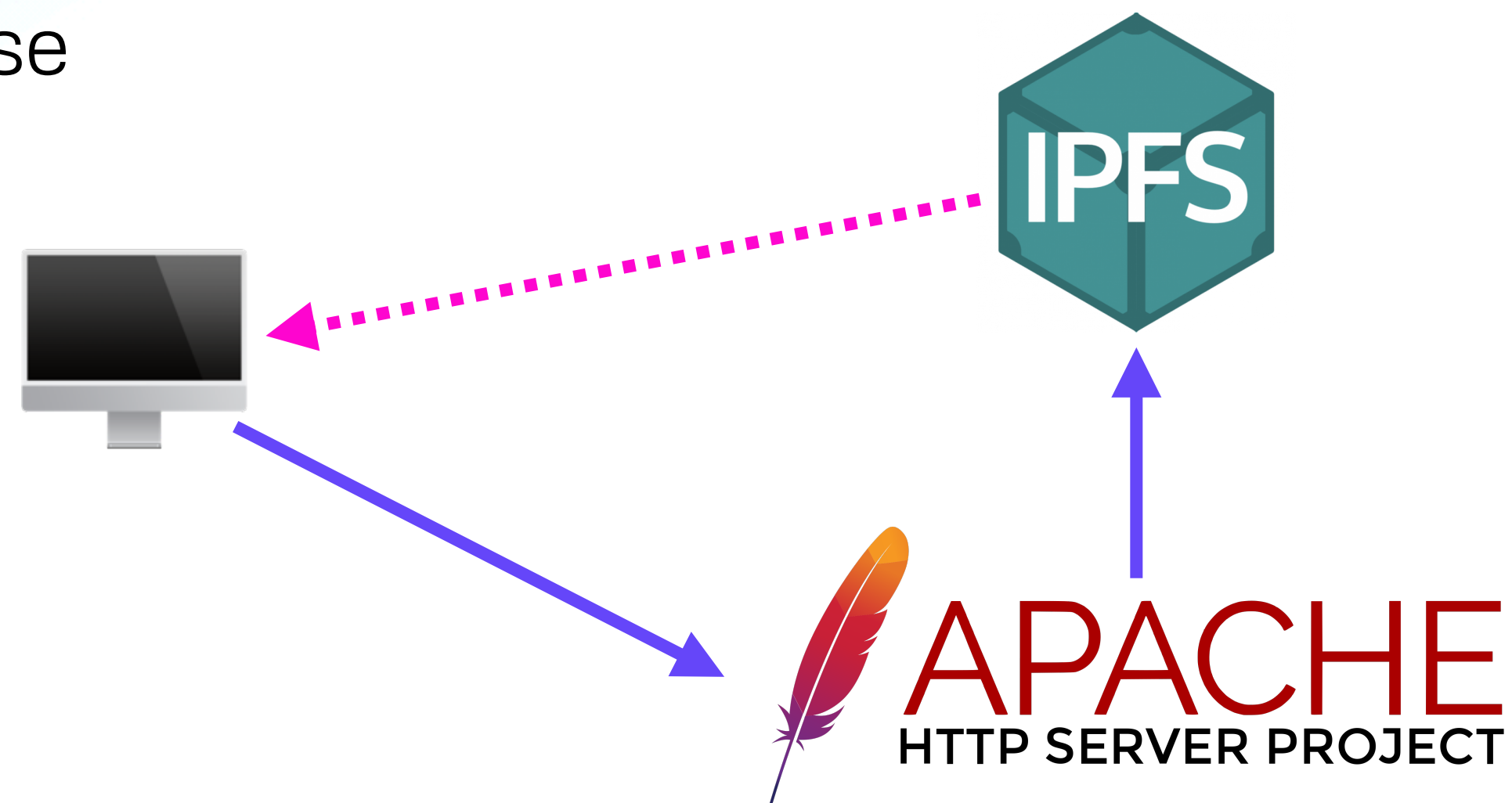
Are You There? 📱

- ◆ Keeping nodes reliably connected in production is extremely difficult (browser, server, desktop, mobile)
 - ◆ Private networks don't make sense for our use case
 - ◆ (We'd love to share an operator node architecture wish list with any maintainers, though)
- ◆ Up/down over a HTTP Gateway is just LARPing decentralization 🌶️
- ◆ Native push doesn't make sense

The Problem

Are You There? 📱

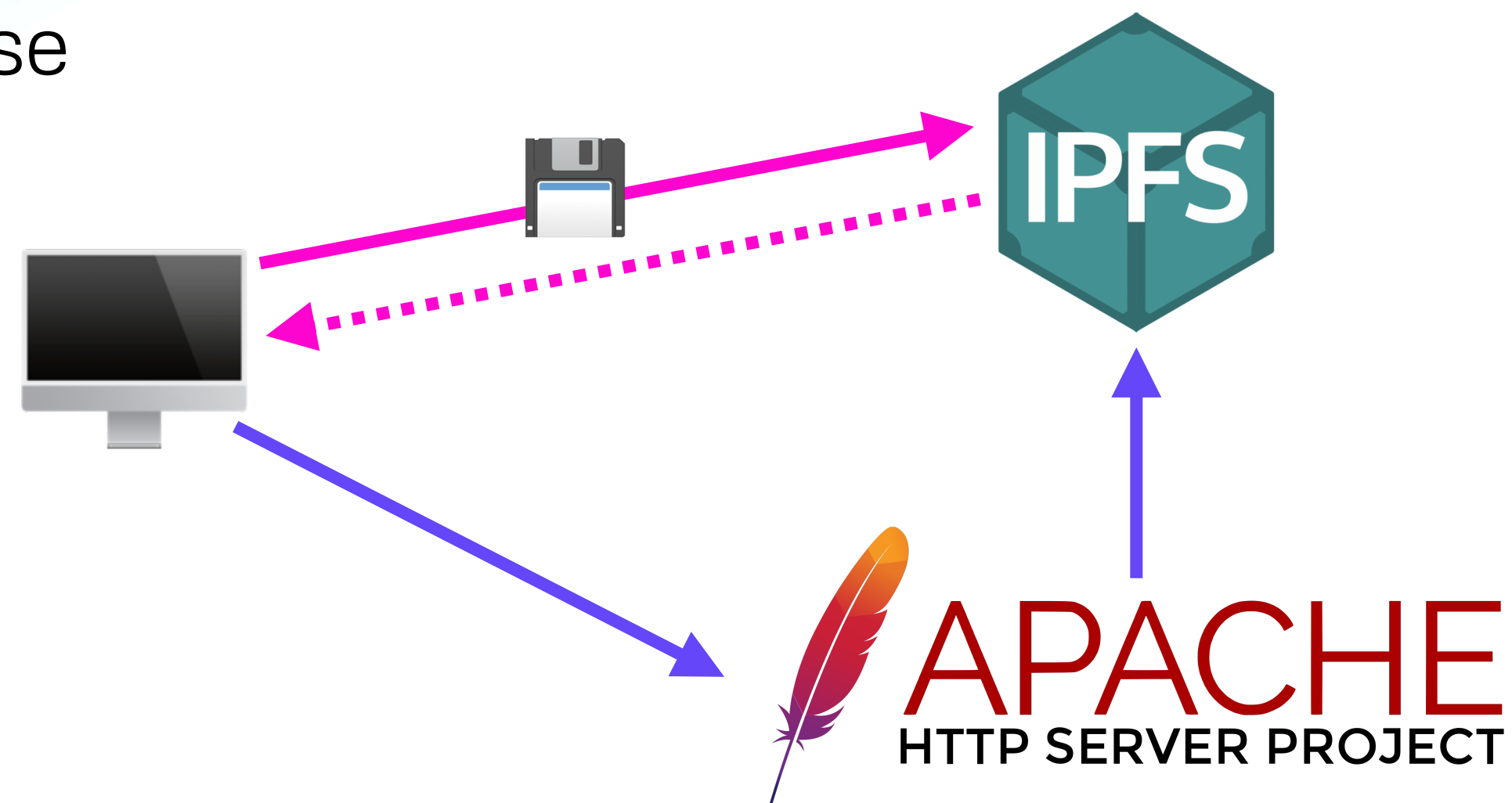
- ◆ Keeping nodes reliably connected in production is extremely difficult (browser, server, desktop, mobile)
 - ◆ Private networks don't make sense for our use case
 - ◆ (We'd love to share an operator node architecture wish list with any maintainers, though)
- ◆ Up/down over a HTTP Gateway is just LARPing decentralization 🌶️
- ◆ Native push doesn't make sense



The Problem

Are You There? 📱

- ◆ Keeping nodes reliably connected in production is extremely difficult (browser, server, desktop, mobile)
 - ◆ Private networks don't make sense for our use case
 - ◆ (We'd love to share an operator node architecture wish list with any maintainers, though)
- ◆ Up/down over a HTTP Gateway is just LARPing decentralization 🌶️
- ◆ Native push doesn't make sense



The Problem

Bitswap Redux

The Problem

Bitswap Redux



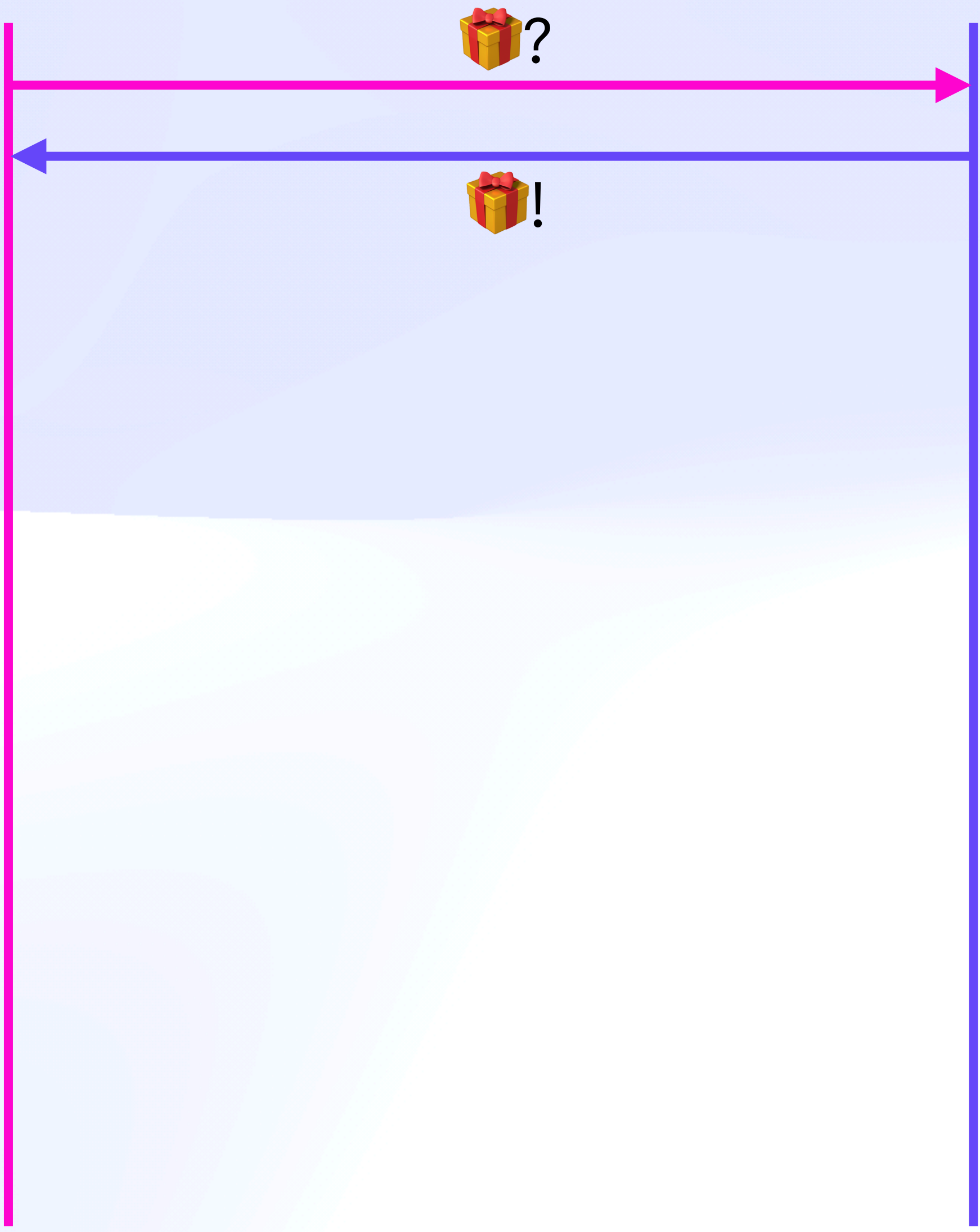
The Problem

Bitswap Redux



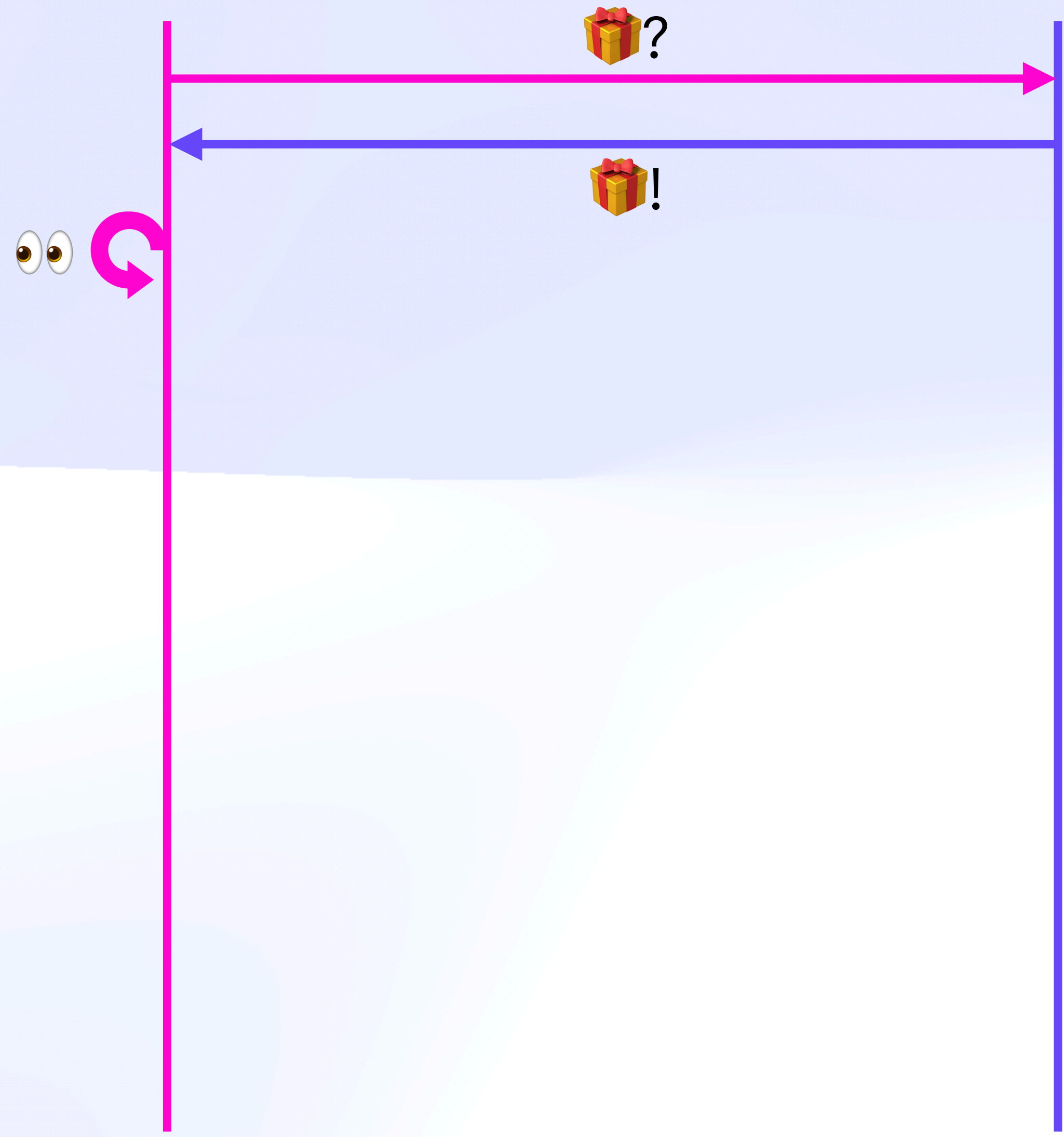
The Problem

Bitswap Redux



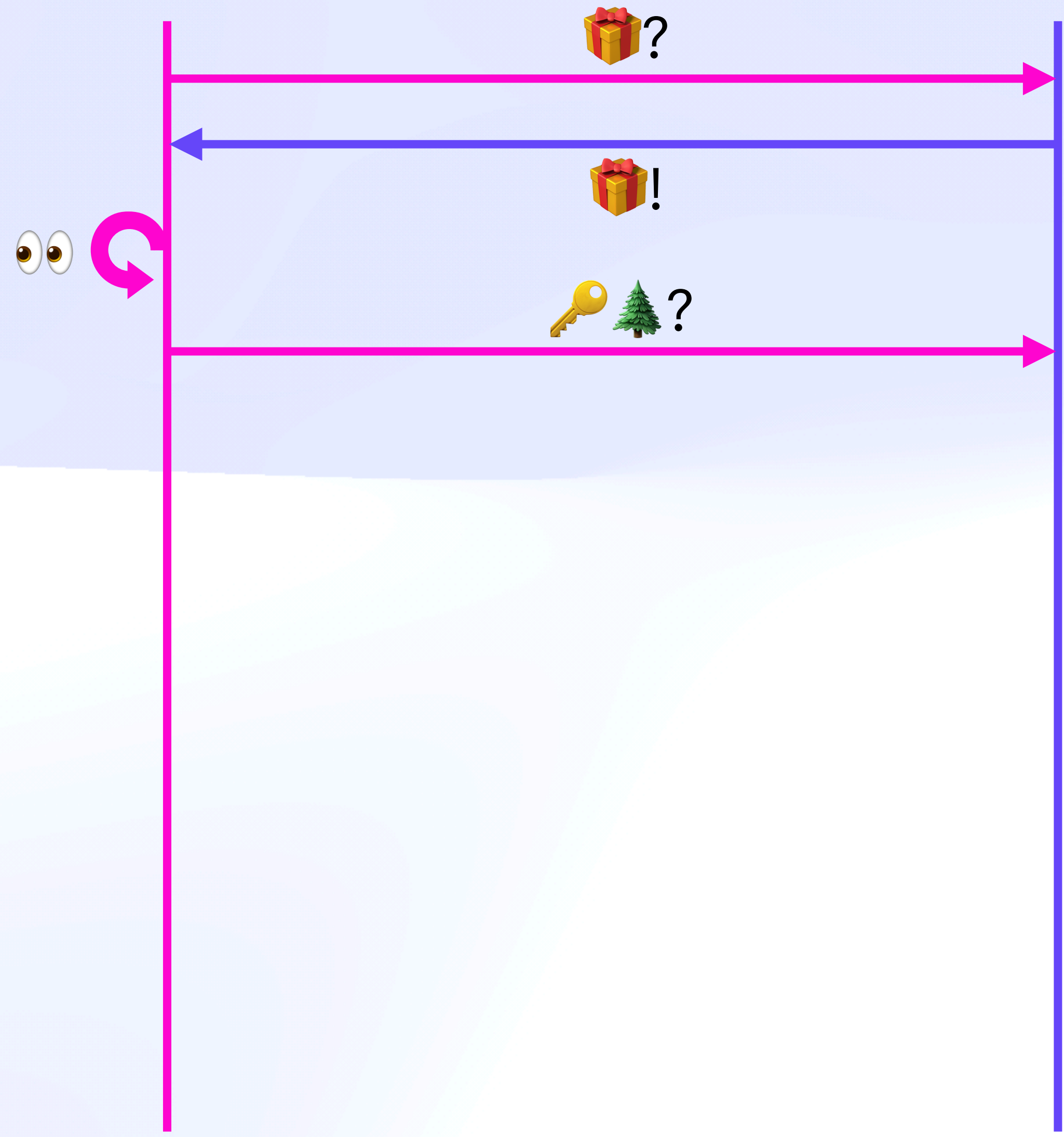
The Problem

Bitswap Redux



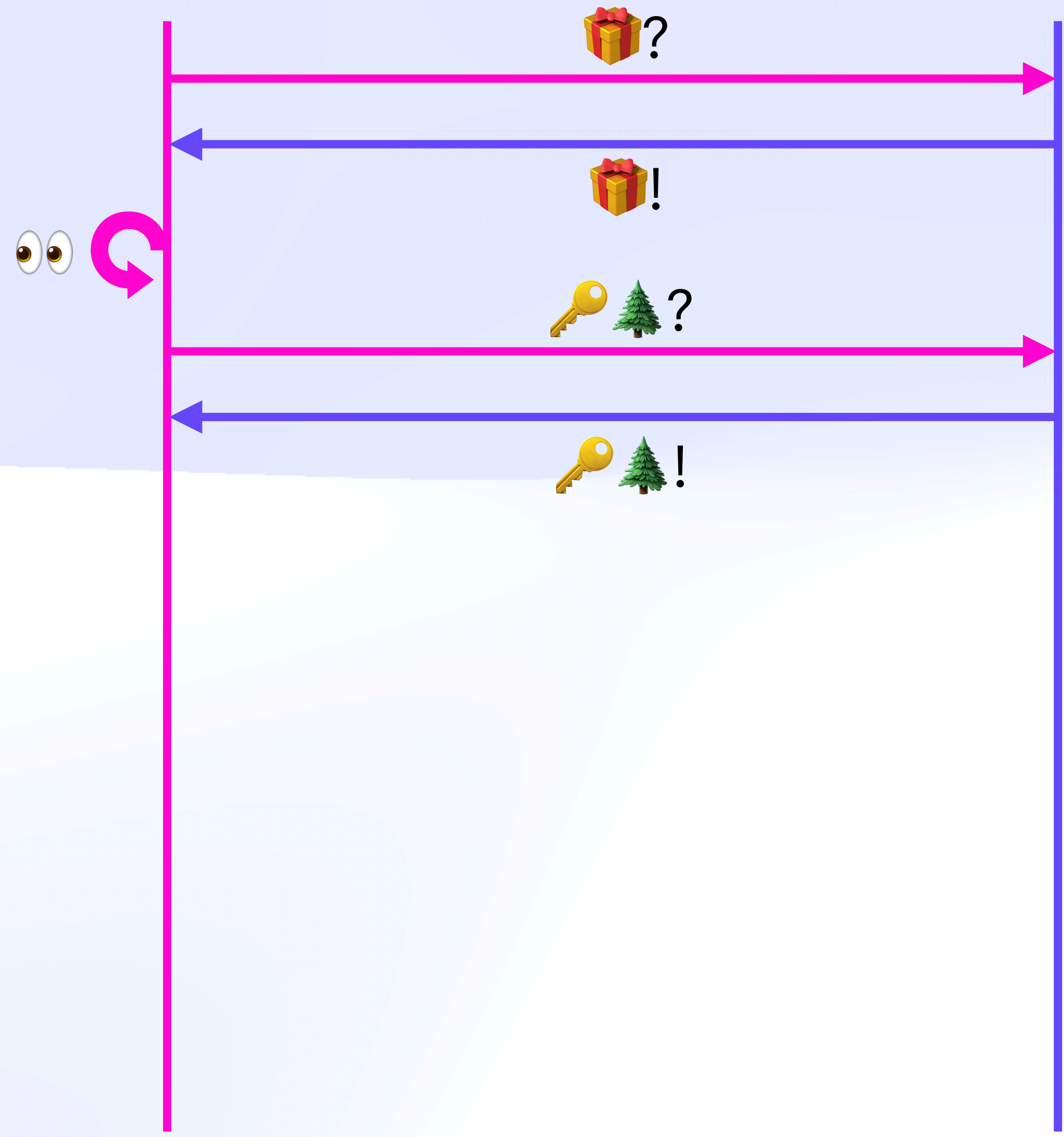
The Problem

Bitswap Redux



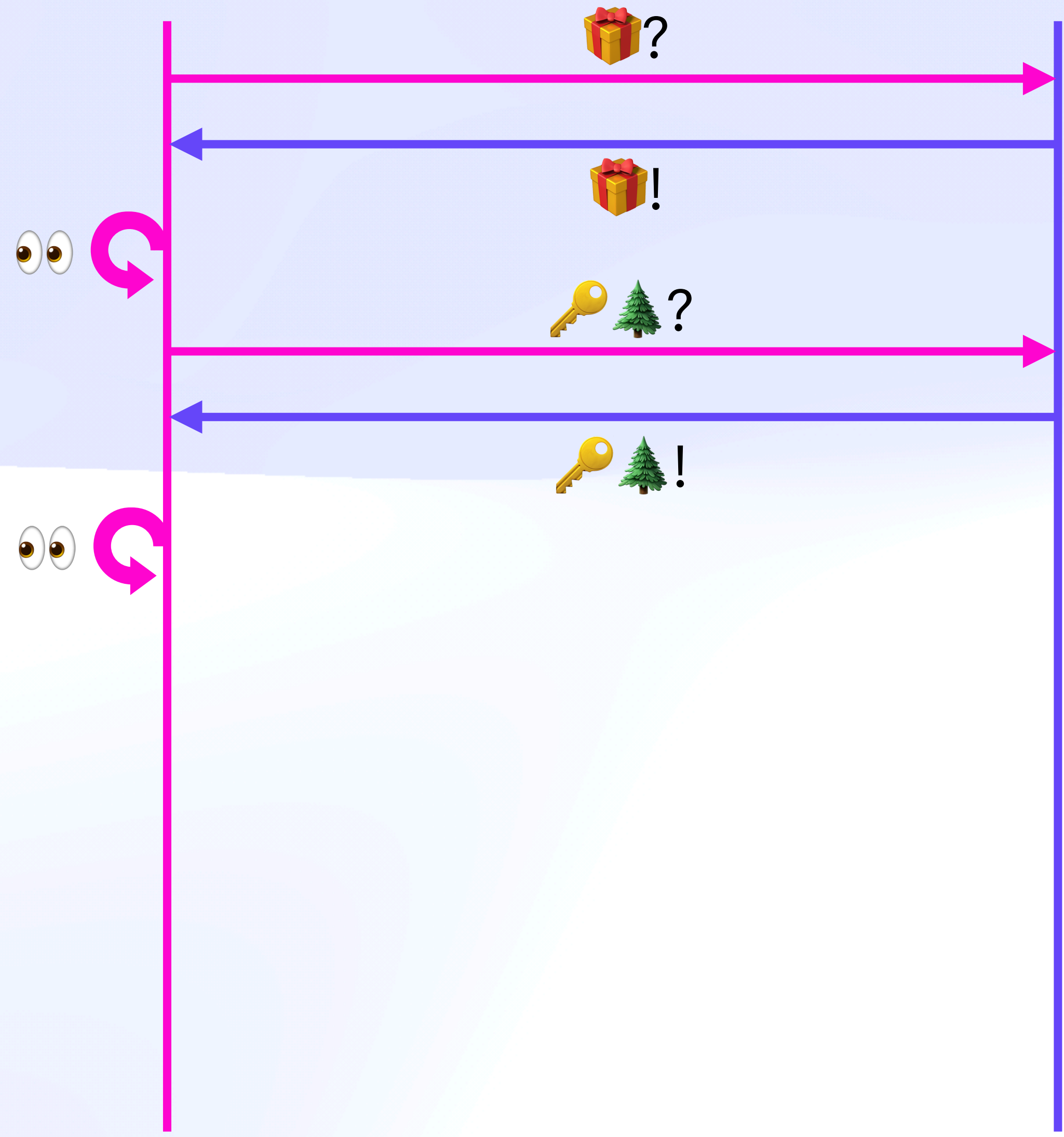
The Problem

Bitswap Redux



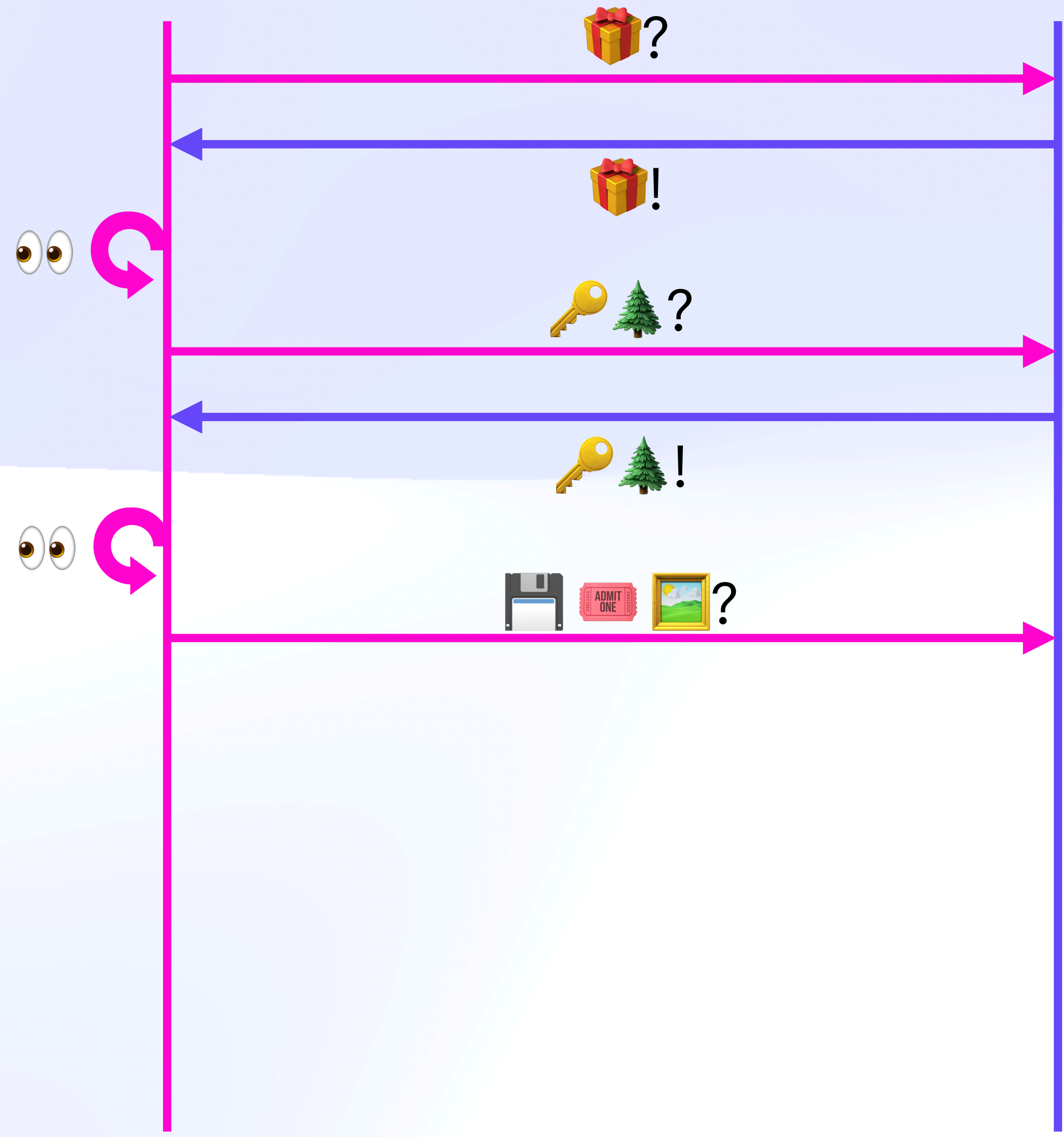
The Problem

Bitswap Redux



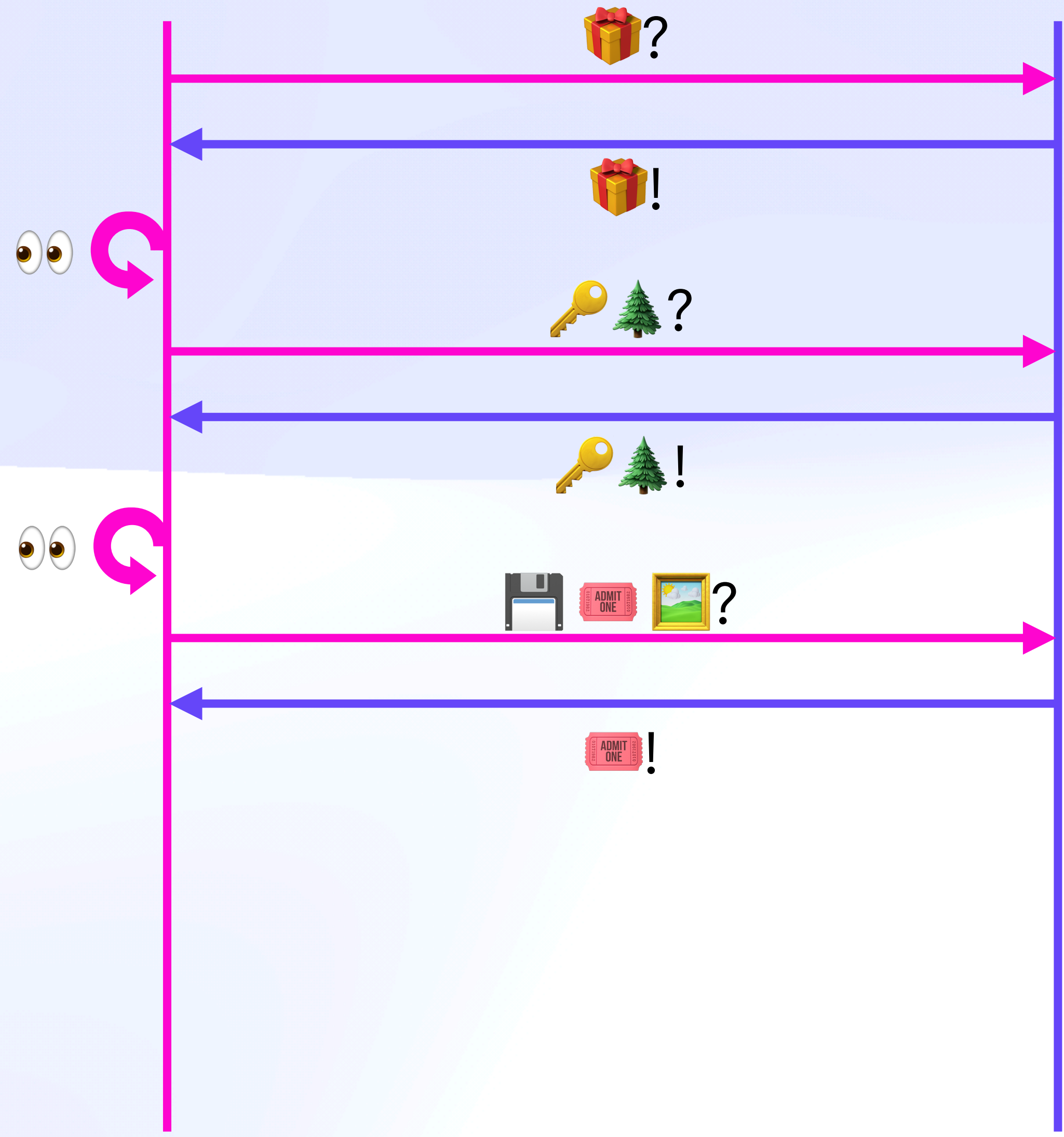
The Problem

Bitswap Redux



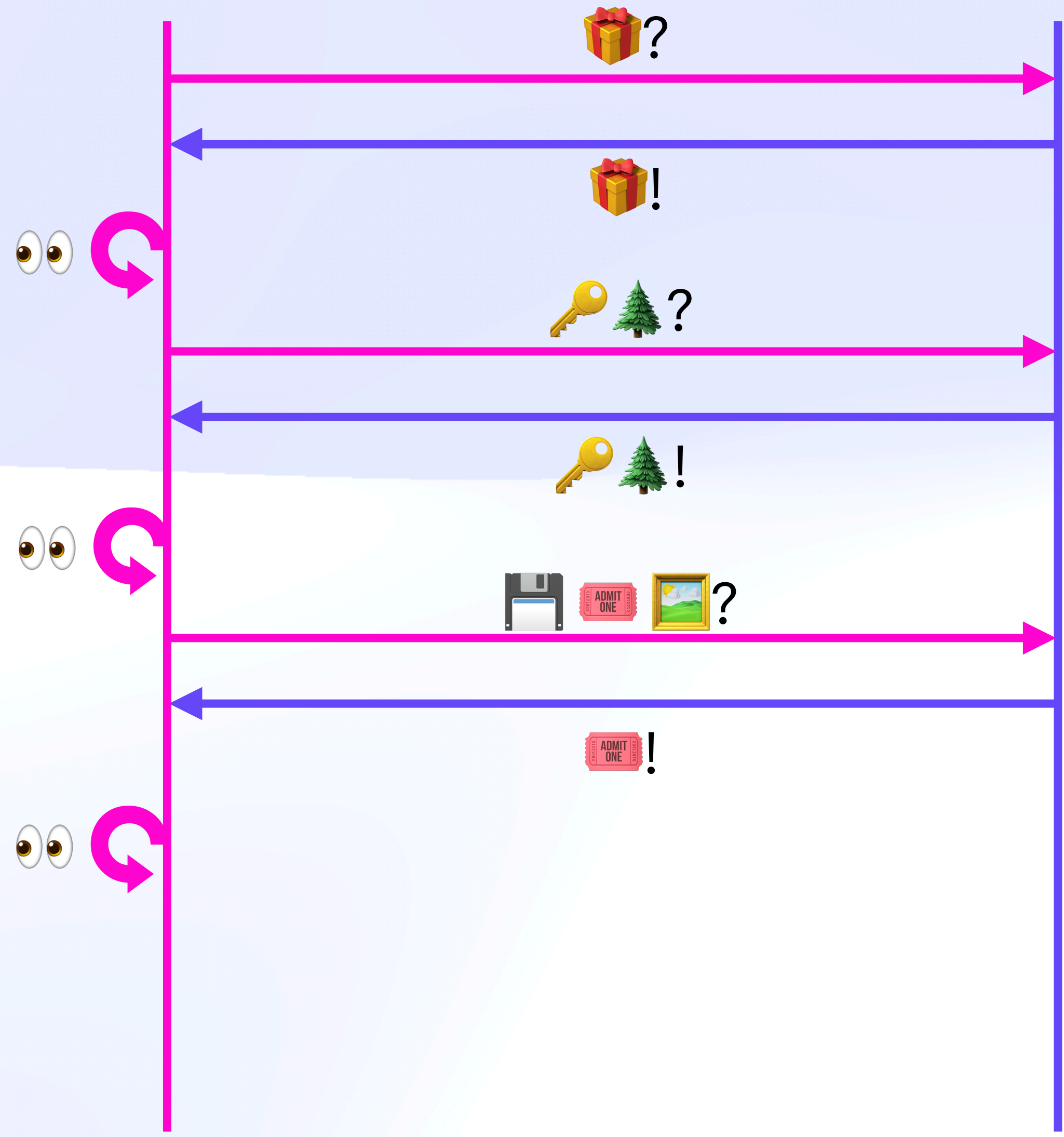
The Problem

Bitswap Redux



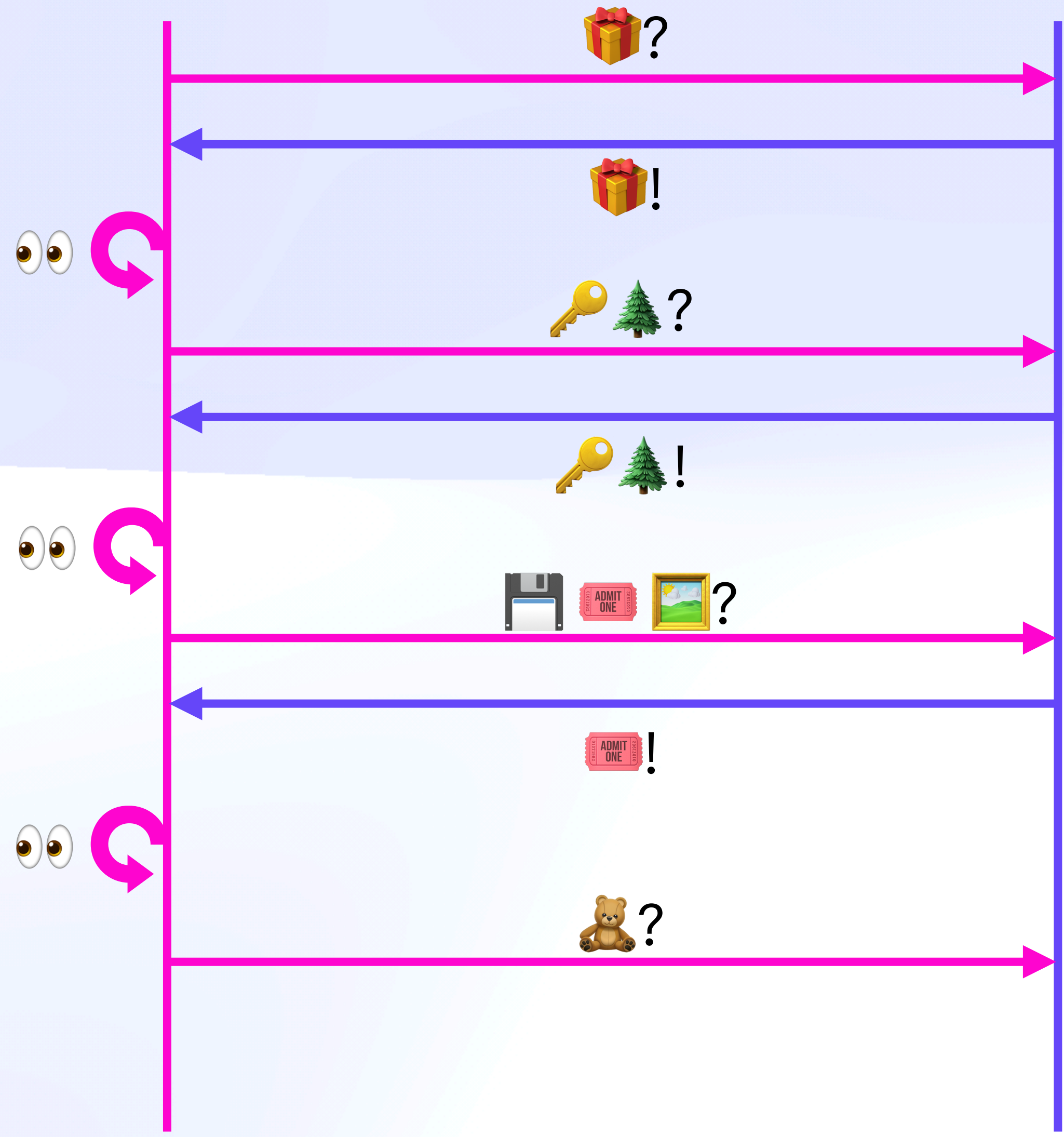
The Problem

Bitswap Redux



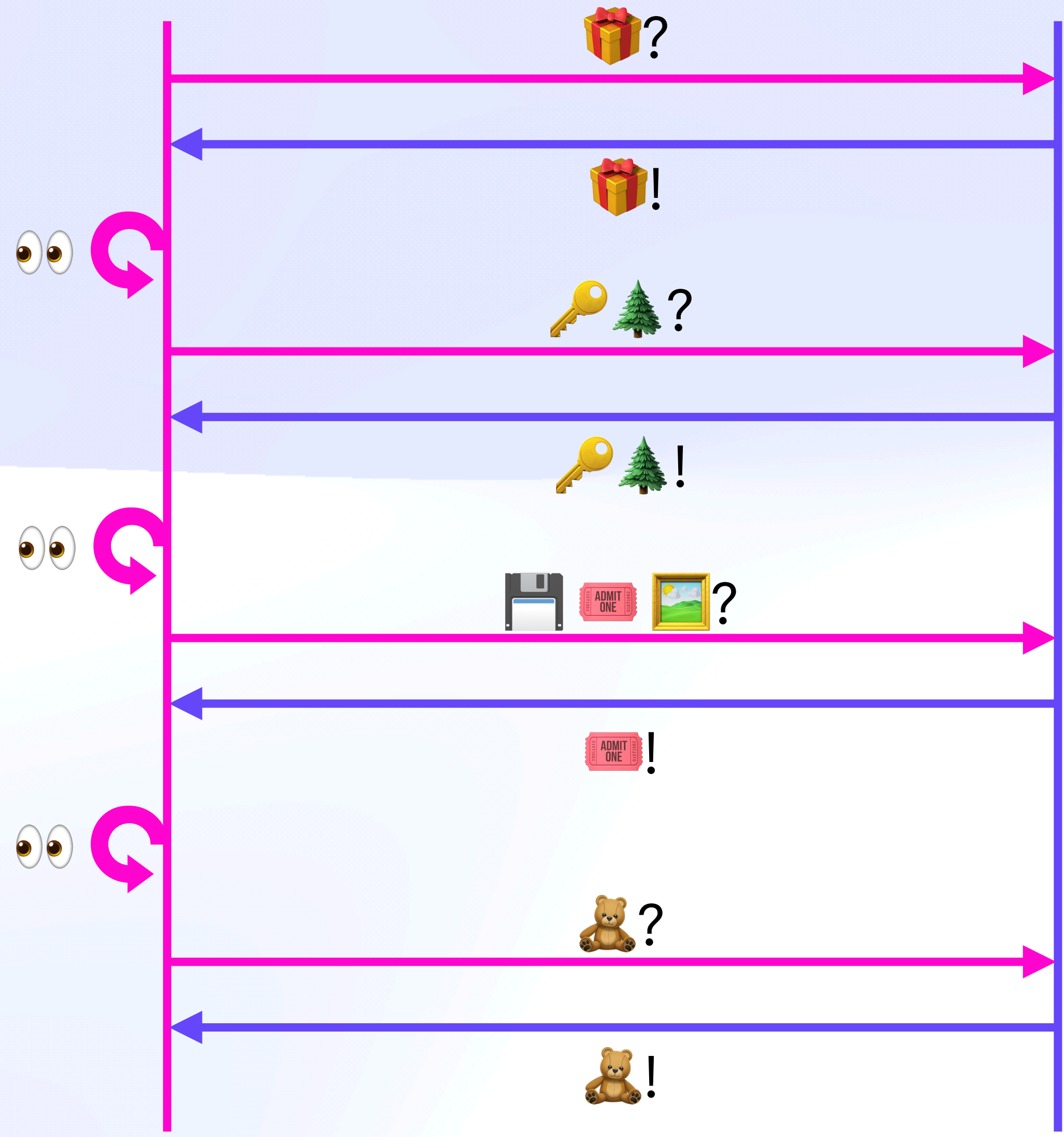
The Problem

Bitswap Redux



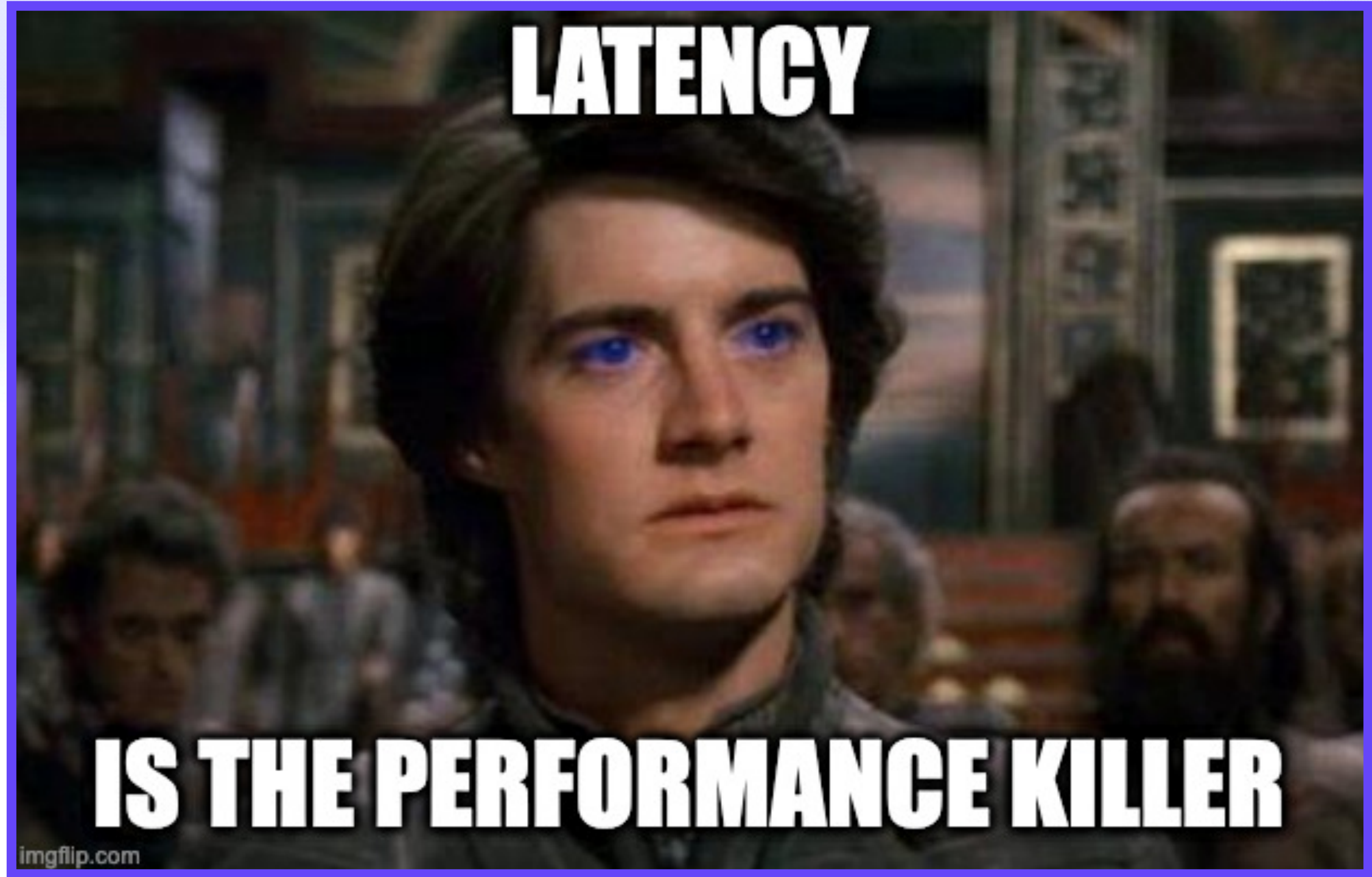
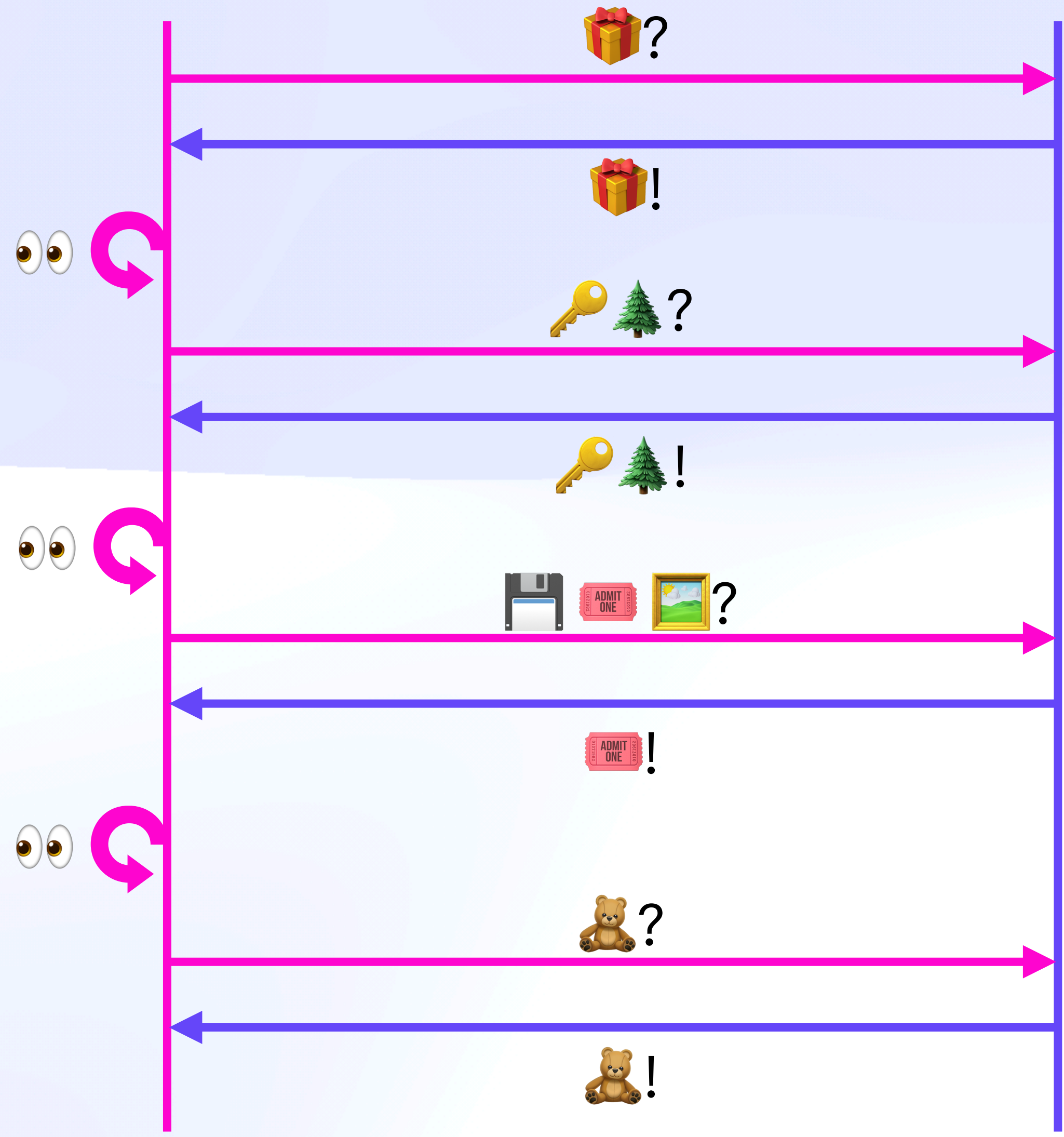
The Problem

Bitswap Redux



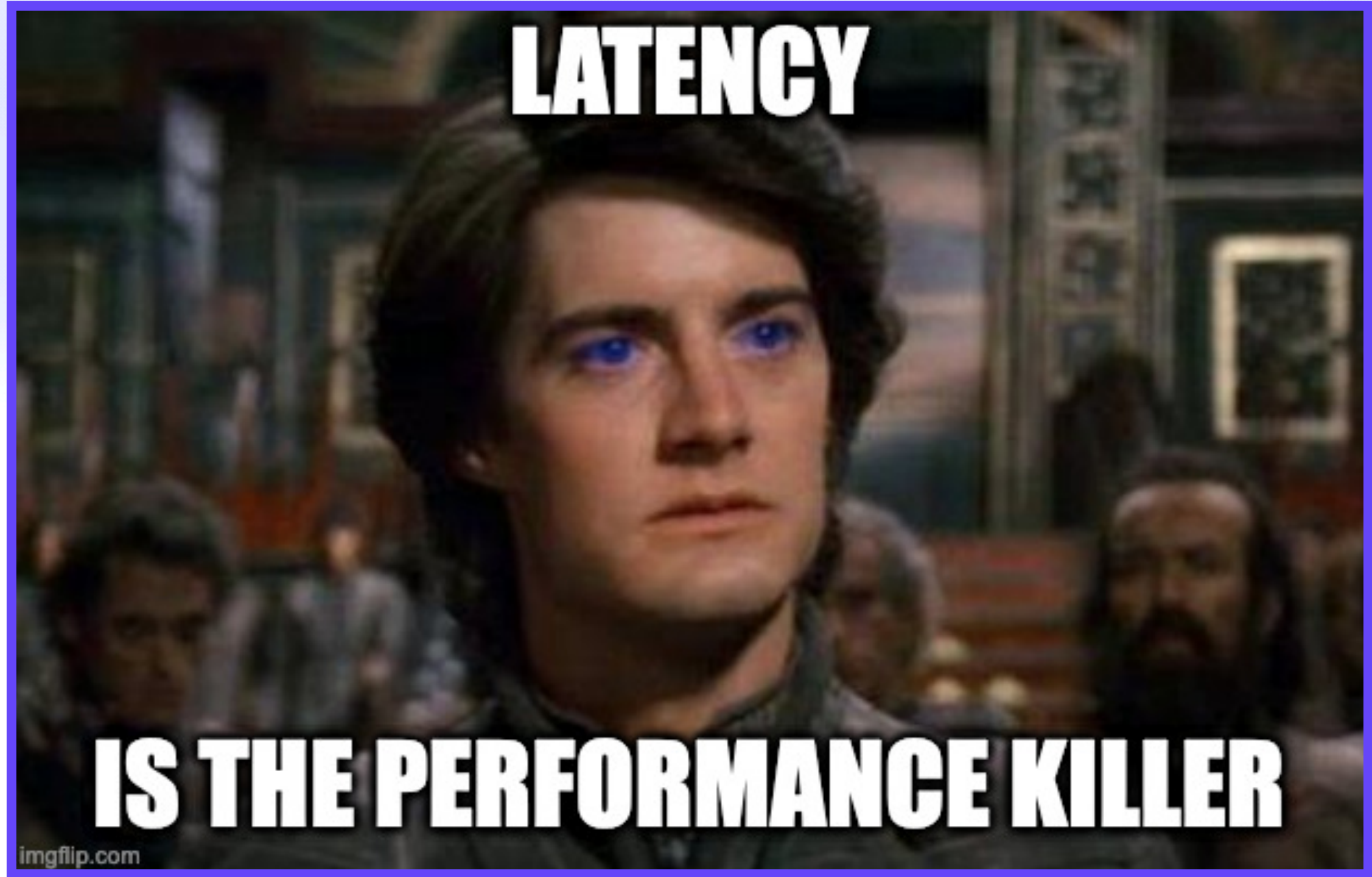
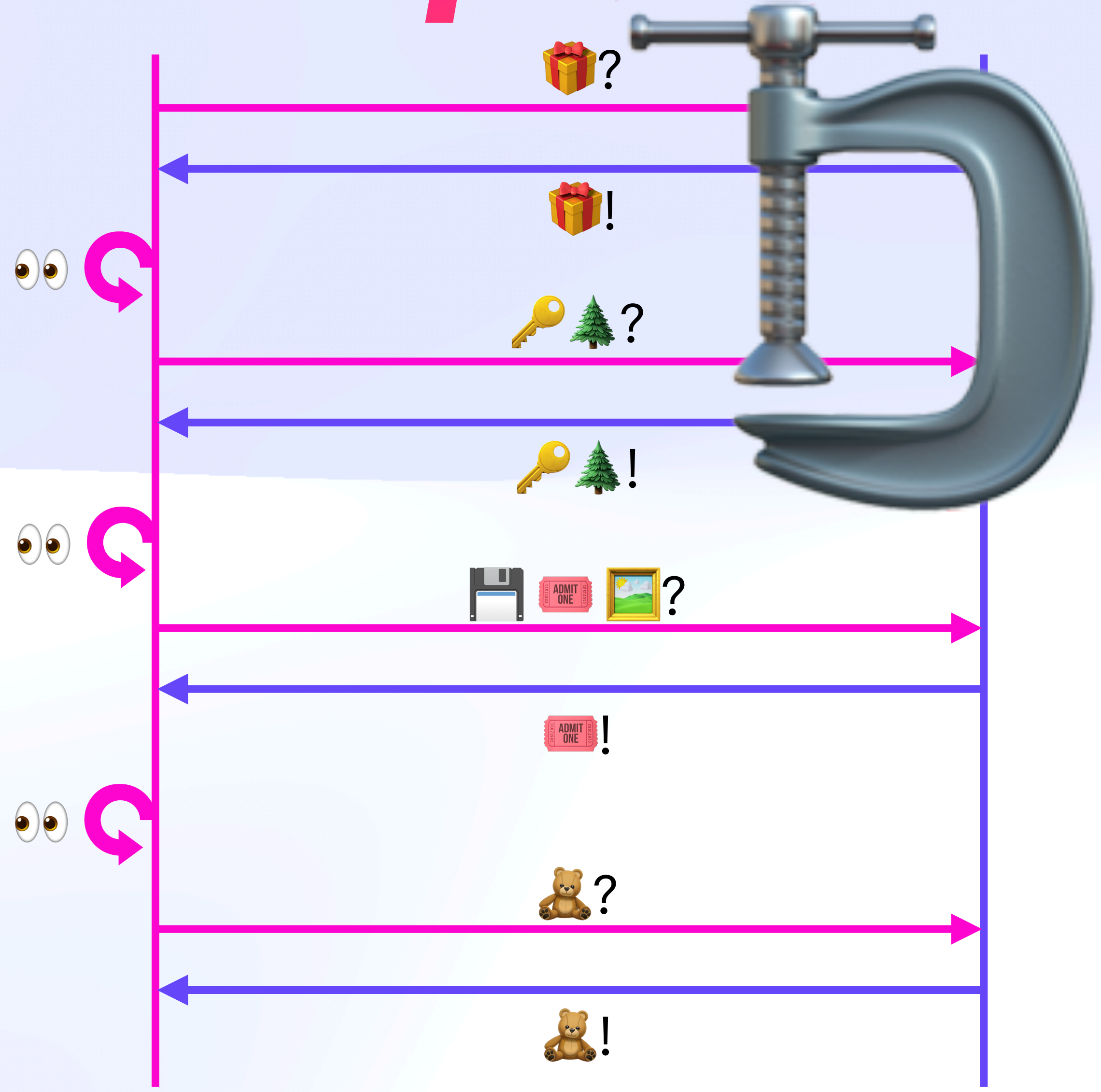
The Problem

Bitswap Redux



The Problem

Bitswap Redux

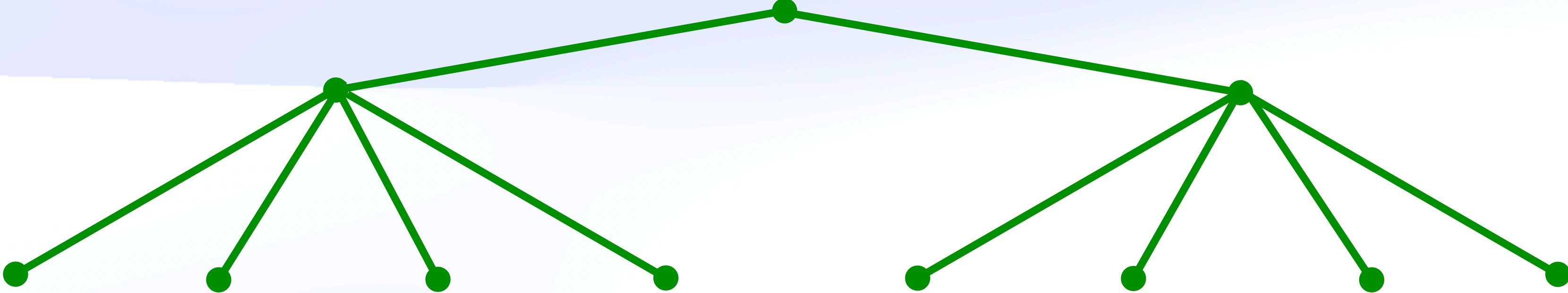


The Problem

Breadth vs Depth

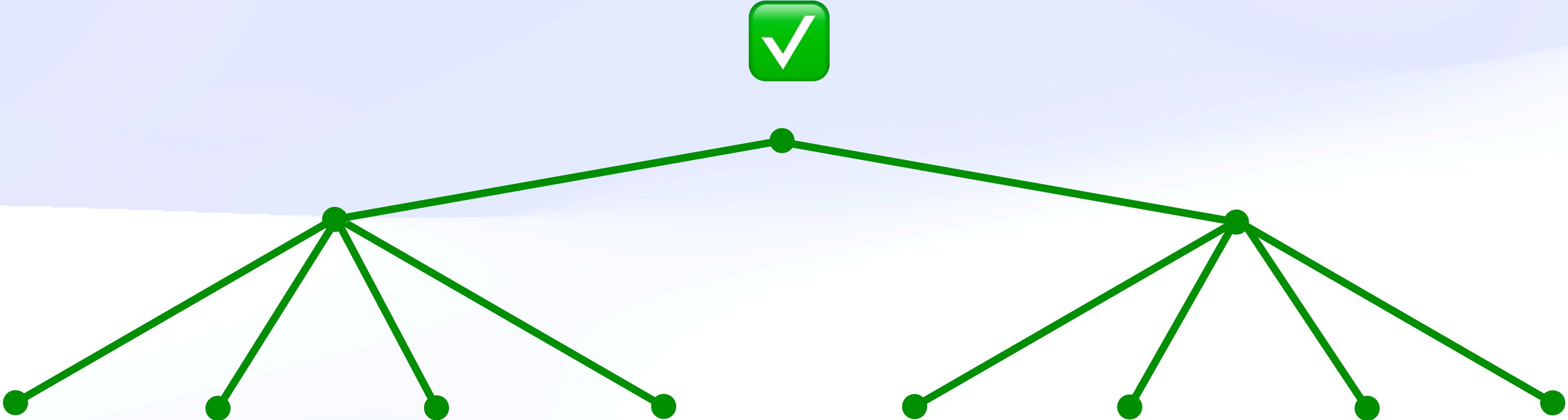
The Problem

Breadth vs Depth



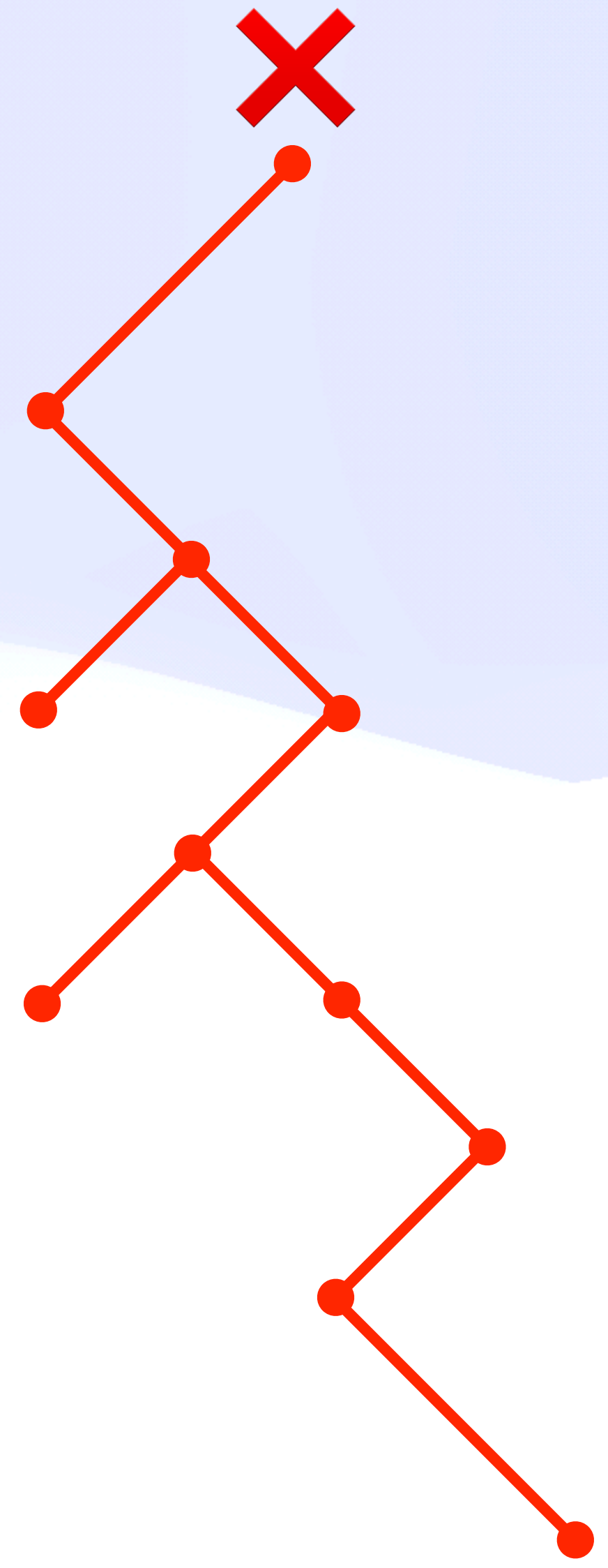
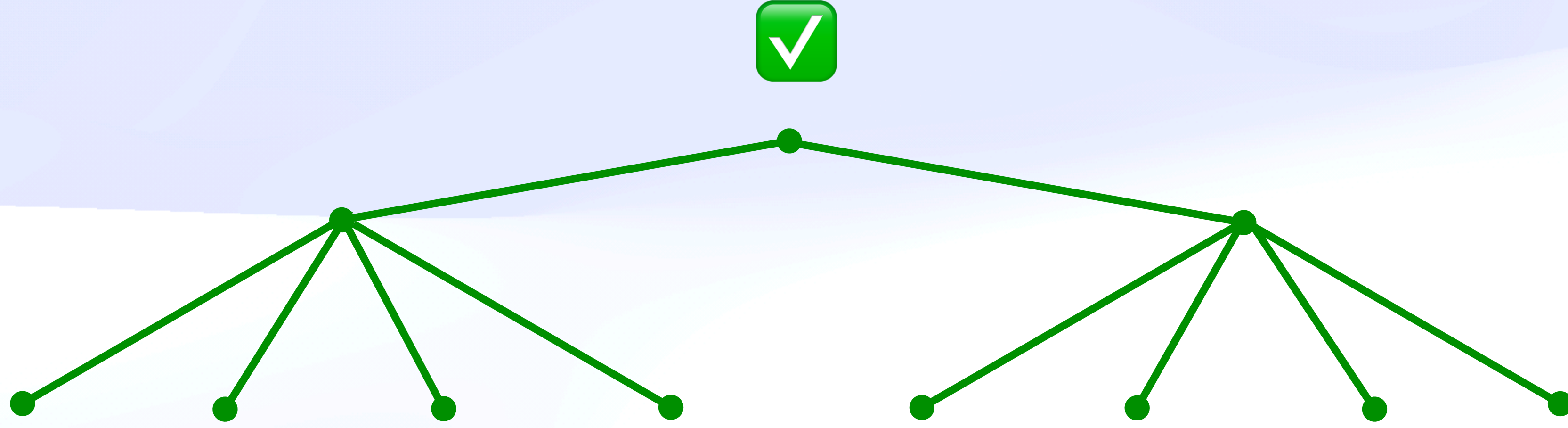
The Problem

Breadth vs Depth



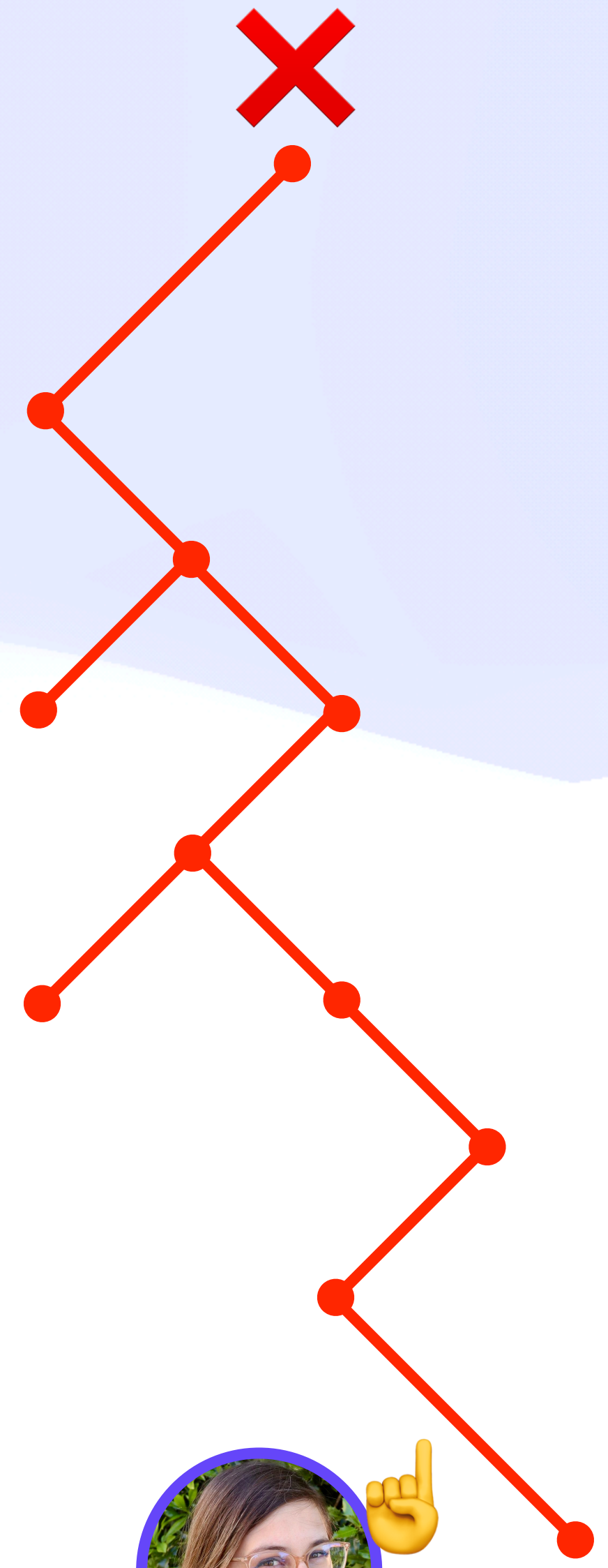
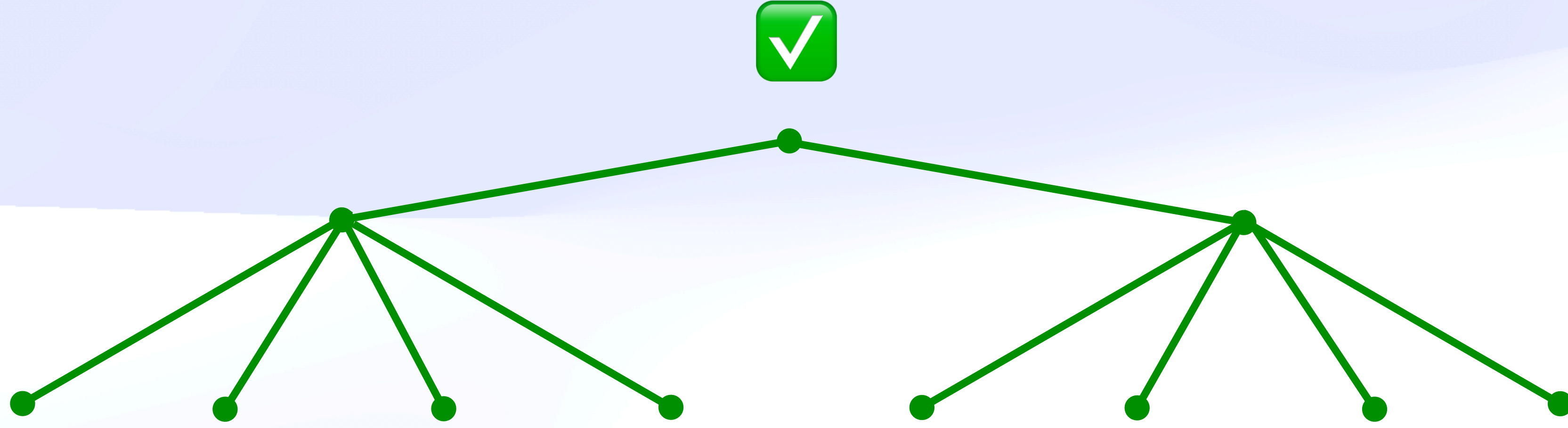
The Problem

Breadth vs Depth



The Problem

Breadth vs Depth



Prior Art

On the Shoulders of Giants

Prior Art

IPFS Ecosystem

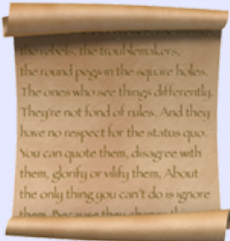
Prior Art

IPFS Ecosystem

- ◆ Qri's DSync
- ◆ GraphSync, Selectors (orthogonal)
- ◆ WNFS(v1) Hashed History Table
- ◆ Merkle Bloom

The Problem

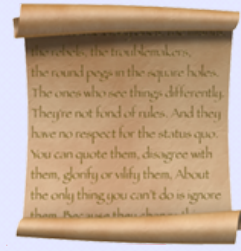
Papers



There are two kinds of papers:
the good papers and the bad papers.
The good papers are things that
people are fond of. And they
have no respect for the status quo.
You can quote them, disagree with
them, glorify or vilify them. About
the only thing you can't do is ignore
them. Because they are things.

The Problem

Papers



- ◆ **To Push or Pull: On Reducing Communication and Synchronization in Graph Computations**
- ◆ **What's The Difference? Efficient Set Reconciliation Without Prior Context**
- ◆ **Synchronizing Namespaces with Invertible Bloom Filters**
- ◆ **The Hash History Approach for Reconciling Mutual Inconsistency**
- ◆ **HEX BLOOM: An Efficient Method for Authenticity and Integrity Verification in Privacy-preserving Computing**
- ◆ **The Distributed Bloom Filter**

CAR Mirror

One-to-One

CAR Mirror

Strategy

CAR Mirror

Strategy

- ◆ Reduce graph scope
- ◆ Send index with request
- ◆ Use heuristics to find deduplication

CAR Mirror



CAR Mirror

Balancing Factors

CAR Mirror

Balancing Factors

Latency

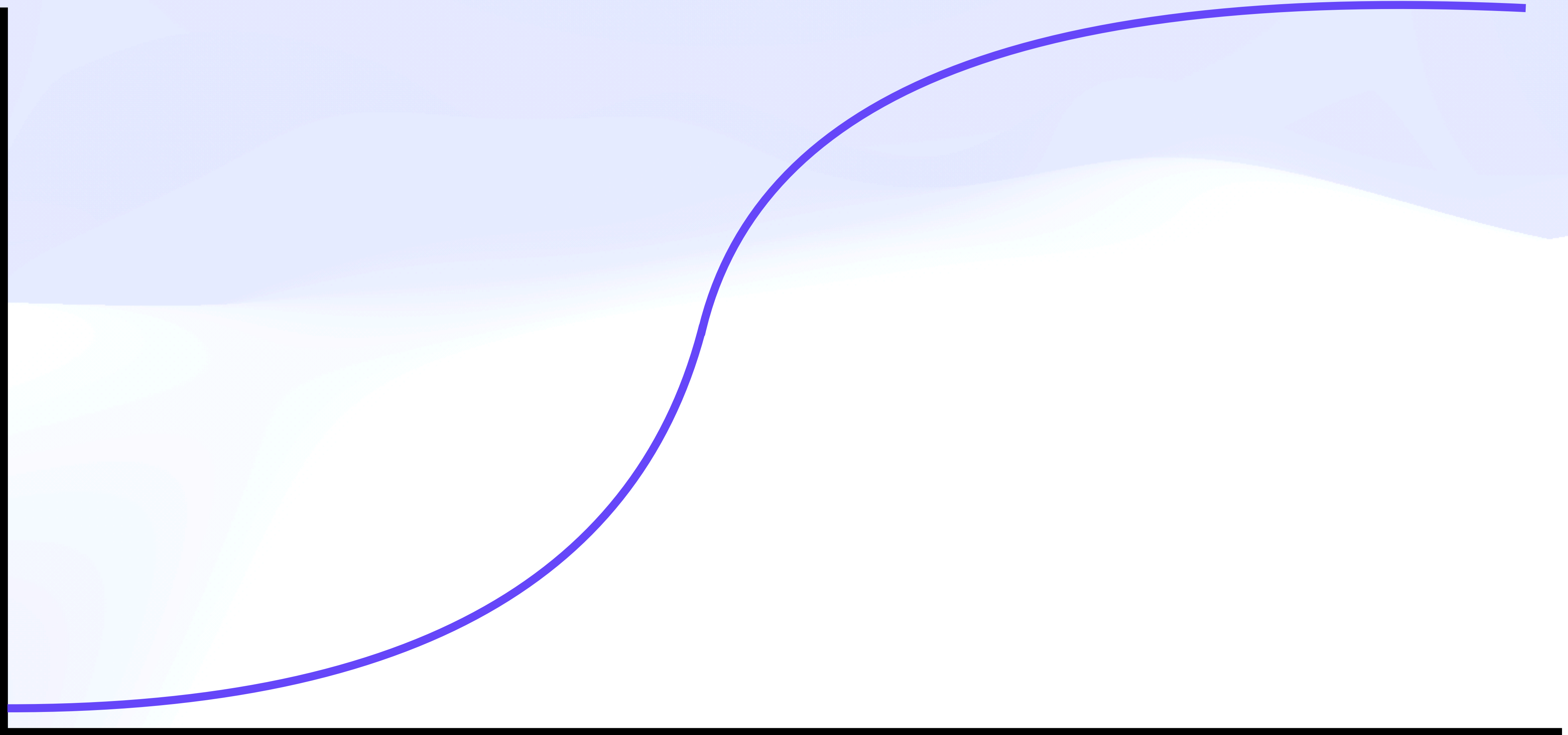
Accuracy/Deduplication



CAR Mirror

Balancing Factors

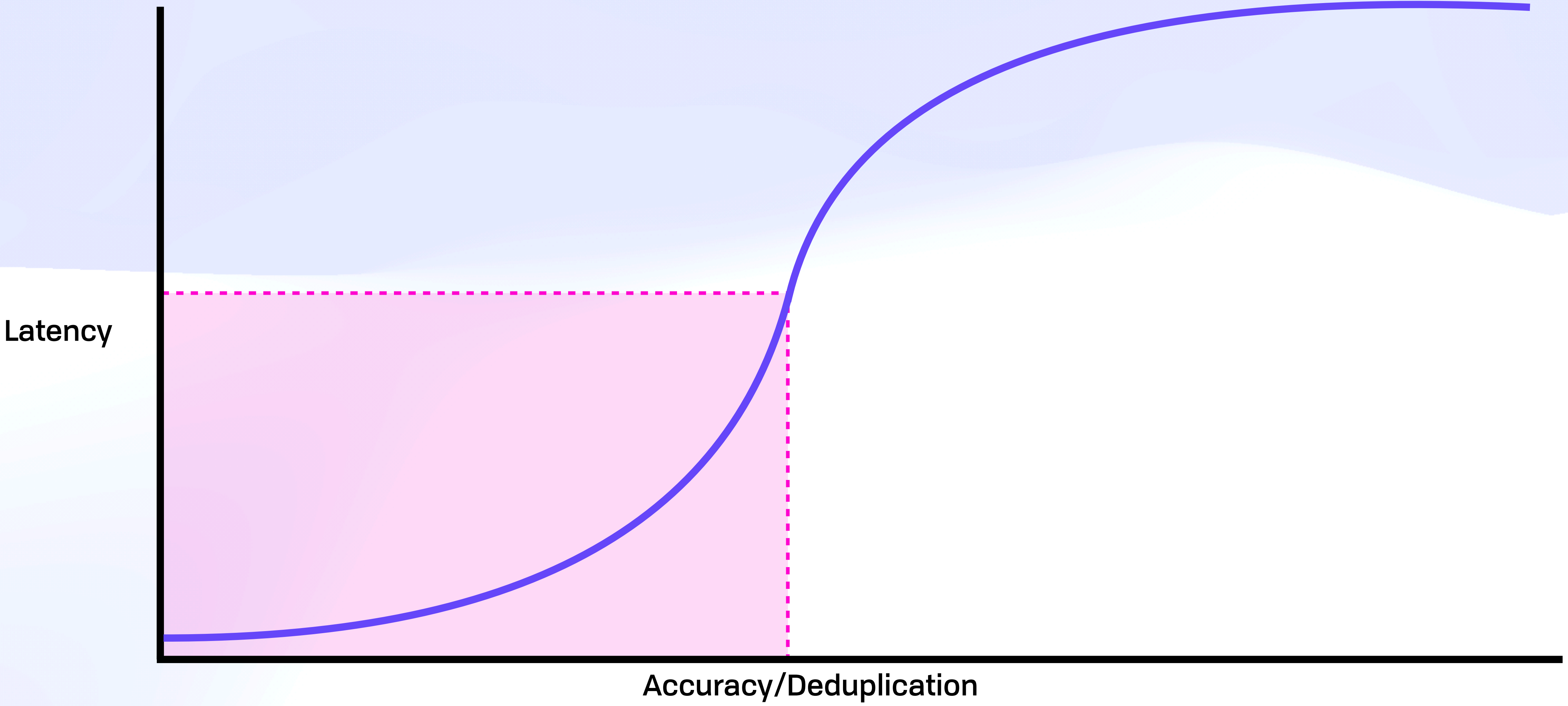
Latency



Accuracy/Deduplication

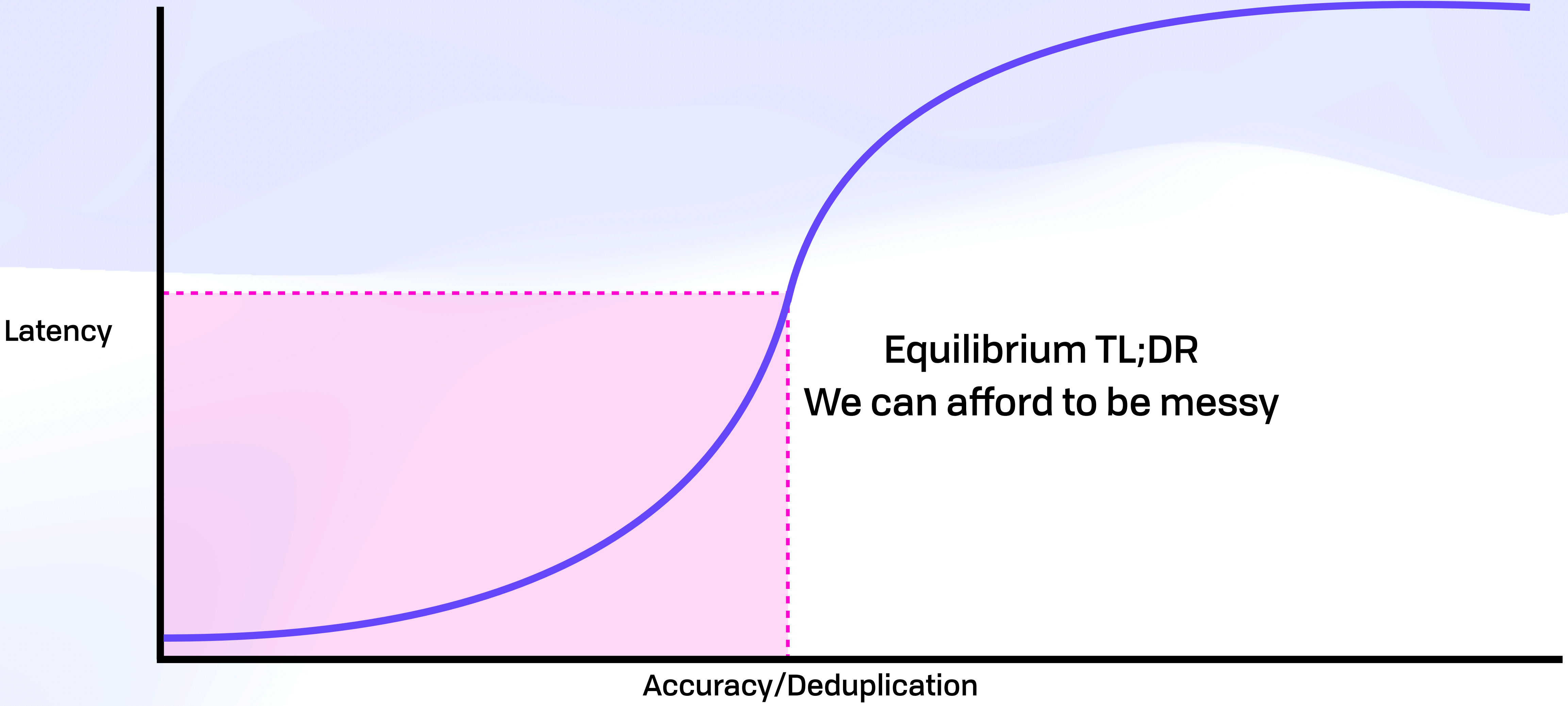
CAR Mirror

Balancing Factors



CAR Mirror

Balancing Factors



CAR Mirror

First Reduce Scope

CAR Mirror

First Reduce Scope

- ◆ AOT

- ◆ Mutable pointers (IPNS, DNSLink, ENS, diffed CID, etc)
- ◆ Previous CAR Mirror **sessions**

CAR Mirror

First Reduce Scope

- ◆ AOT

- ◆ Mutable pointers (IPNS, DNSLink, ENS, diffed CID, etc)
- ◆ Previous CAR Mirror **sessions**

- ◆ Adaptive Optimization

- ◆ Previous CAR Mirror **rounds**

CAR Mirror

First Reduce Scope

- ◆ AOT

- ◆ Mutable pointers (IPNS, DNSLink, ENS, diffed CID, etc)
- ◆ Previous CAR Mirror **sessions**

- ◆ **Adaptive Optimization**

- ◆ Previous CAR Mirror **rounds**

- ◆ **No Overlap or Cold Start**

- ◆ If the set is small, use random nodes
- ◆ *"Worst case" looks very similar to Bitswap*

CAR Mirror

Back of the Napkin

CAR Mirror

Back of the Napkin

- ◆ 500k nodes

CAR Mirror

Back of the Napkin

- ◆ 500k nodes
- ◆ 46-59 chars + 1 delimiter each ~ average 53 chars each

CAR Mirror

Back of the Napkin

- ◆ 500k nodes
- ◆ 46-59 chars + 1 delimiter each ~ average 53 chars each
- ◆ 53 ASCII chars x 500k nodes = **26.5MB**

CAR Mirror

Back of the Napkin

- ◆ 500k nodes
- ◆ 46-59 chars + 1 delimiter each ~ average 53 chars each
- ◆ 53 ASCII chars x 500k nodes = **26.5MB**
- ◆ Gzipped ~ **11.5MB**

CAR Mirror

Back of the Napkin

- ◆ 500k nodes
- ◆ 46-59 chars + 1 delimiter each ~ average 53 chars each
- ◆ 53 ASCII chars x 500k nodes = **26.5MB**
- ◆ Gzipped ~ **11.5MB**

CAR Mirror

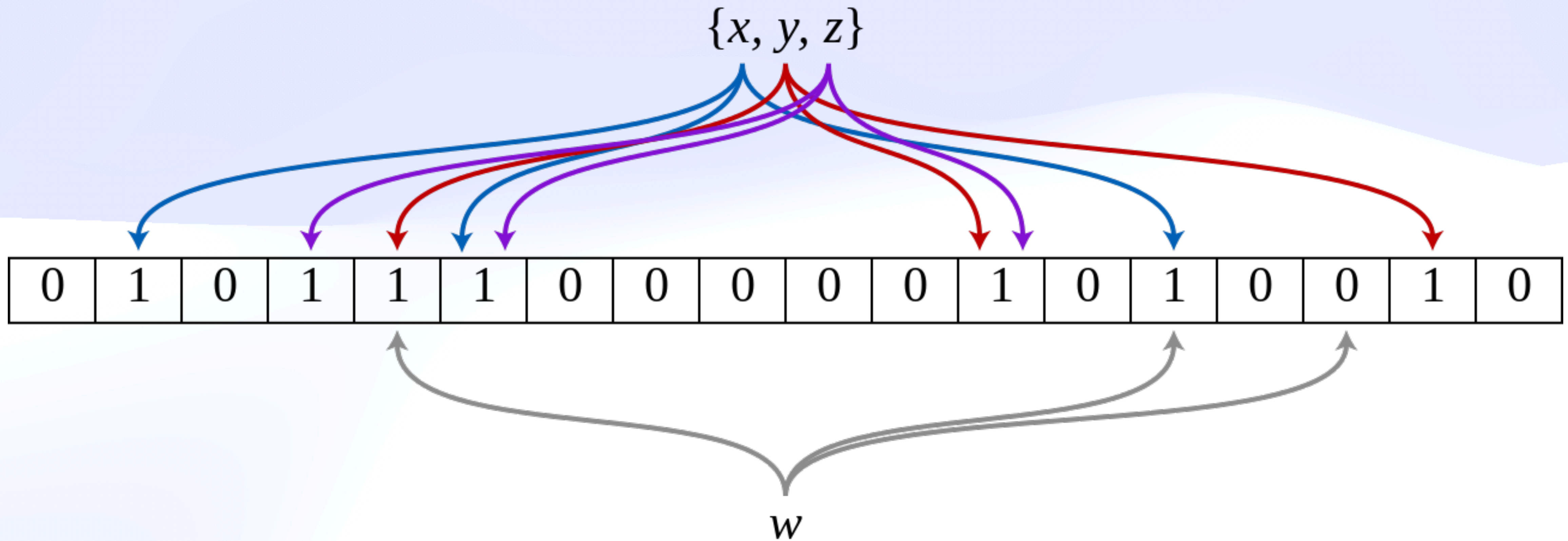
Back of the Napkin

- ◆ 500k nodes
- ◆ 46-59 chars + 1 delimiter each ~ average 53 chars each
- ◆ 53 ASCII chars x 500k nodes = **26.5MB**
- ◆ Gzipped ~ **11.5MB**

can we do better? 🤔

CAR Mirror

Bloom Filter Primer



CAR Mirror

Context and Heuristics

CAR Mirror

Context and Heuristics

- ◆ 500k nodes @ FPP 1/1M = **1.71MB**

CAR Mirror

Context and Heuristics

(Recommend +1 OOM)

◆ 500k nodes @ FPP 1/1M = **1.71MB**



CAR Mirror

Context and Heuristics

(Recommend +1 00M)

◆ 500k nodes @ FPP 1/1M = 1.71MB **93% savings!** 👍

CAR Mirror

Context and Heuristics

(Recommend +1 00M)

- ◆ 500k nodes @ FPP 1/1M = **1.71MB** **93% savings!** 👍
- ◆ Gzipped ~ **896 KB**

CAR Mirror

Context and Heuristics

(Recommend +1 OOM)

- ◆ 500k nodes @ FPP 1/1M = 1.71MB **93% savings!** 👍
- ◆ Gzipped ~ 896 KB **92% savings!** 👍

CAR Mirror

Context and Heuristics

(Recommend +1 OOM)

- ◆ 500k nodes @ FPP 1/1M = 1.71MB **93% savings!** 👍
- ◆ Gzipped ~ 896 KB **92% savings!** 👍

AKA ~1 x OOM 🎉

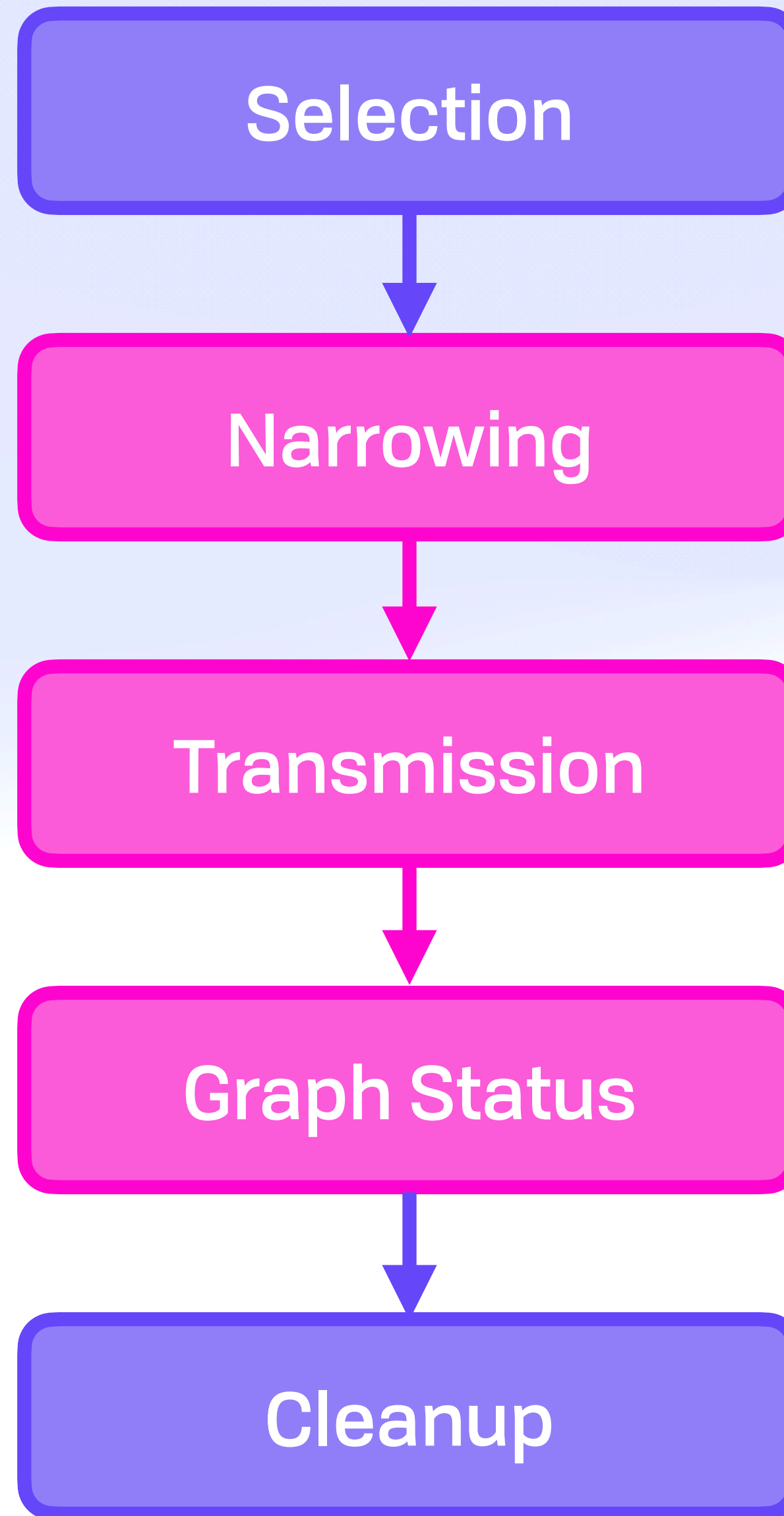
Zero false positives on average

CAR Mirror

Stages

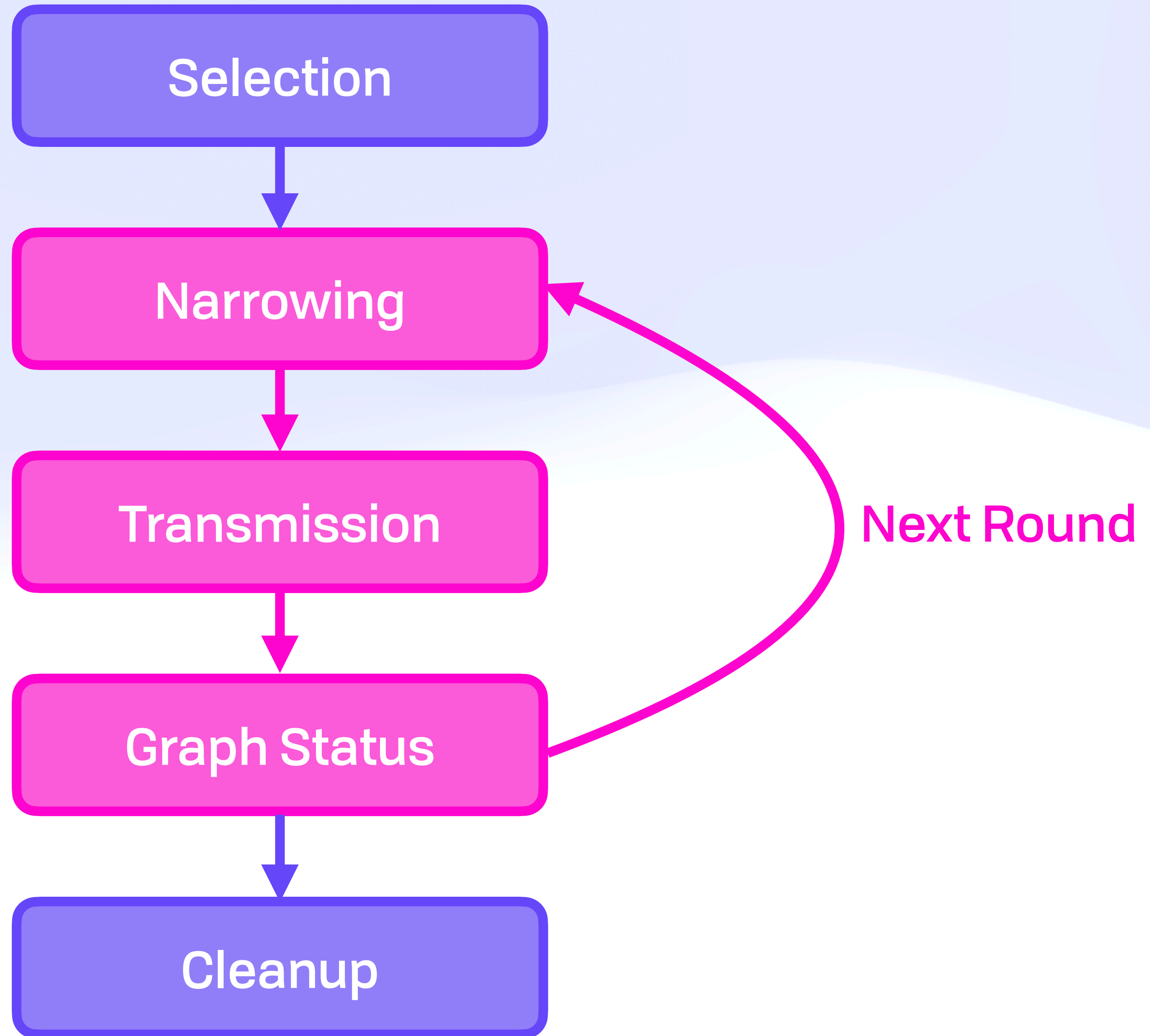
CAR Mirror

Stages



CAR Mirror

Stages



CAR Mirror

Over the Wire

CAR Mirror

Over the Wire

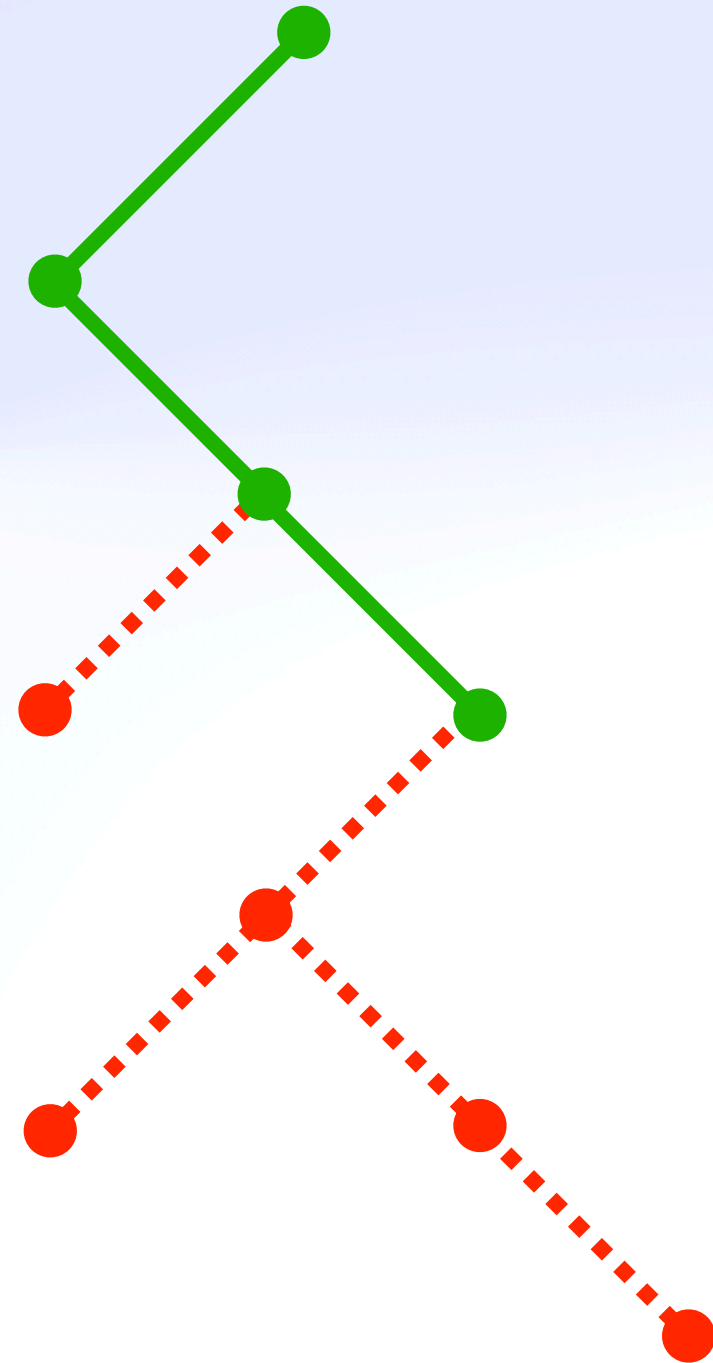
- ◆ Bloom
- ◆ CID Roots
- ◆ CAR

CAR Mirror

Over the Wire

Requestor Pull

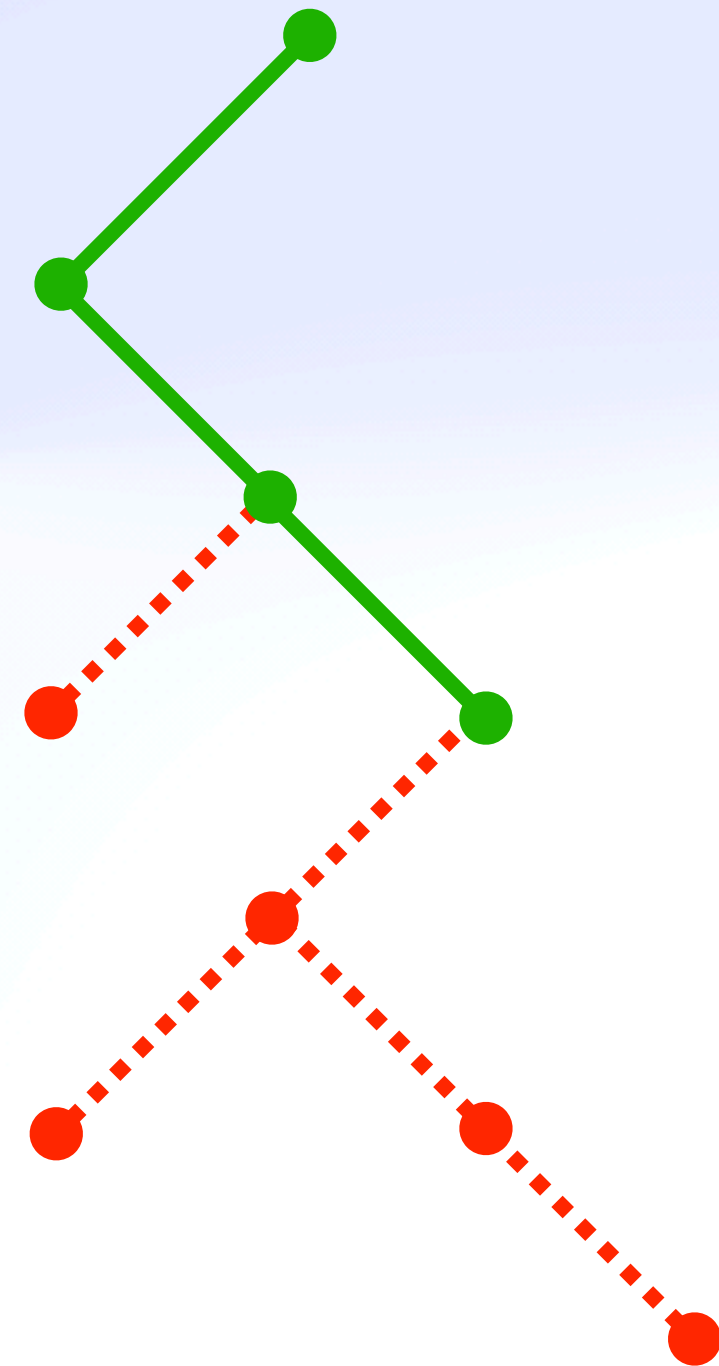
- ◆ Bloom
- ◆ CID Roots
- ◆ CAR



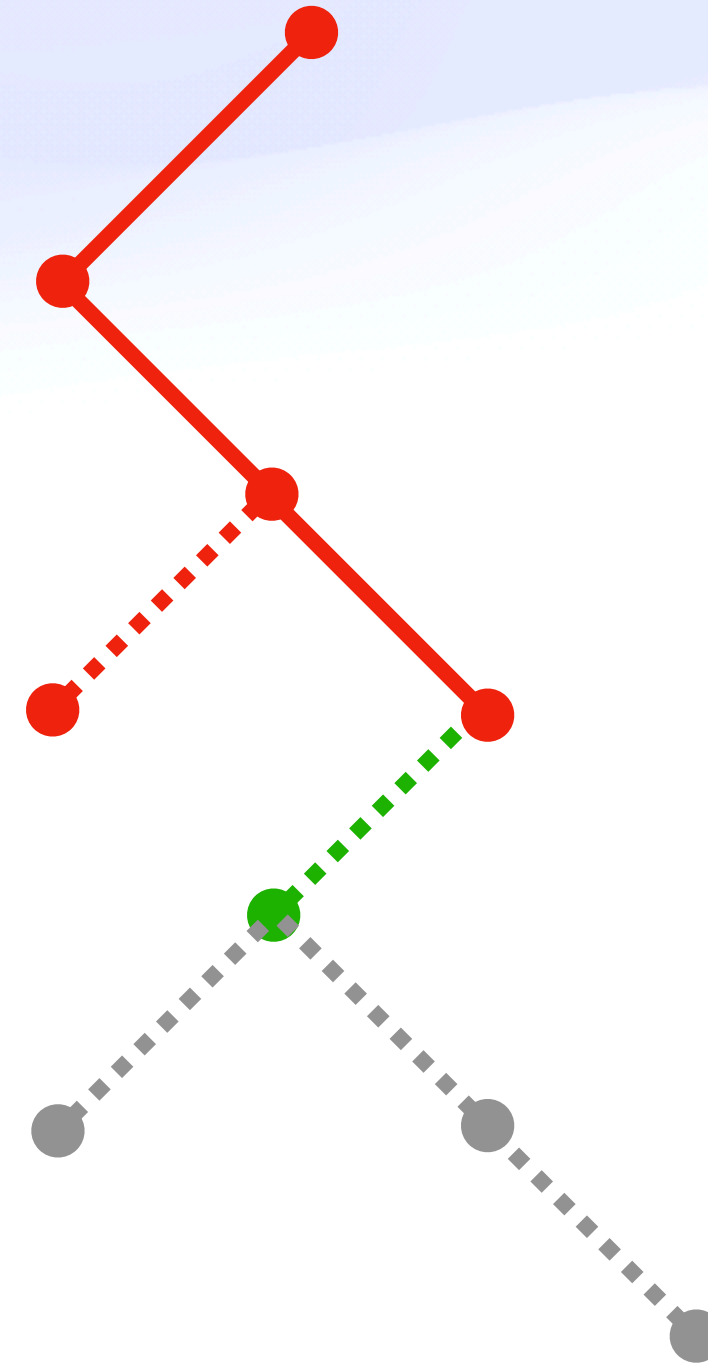
CAR Mirror

Over the Wire

Requestor Pull



Requestor Push

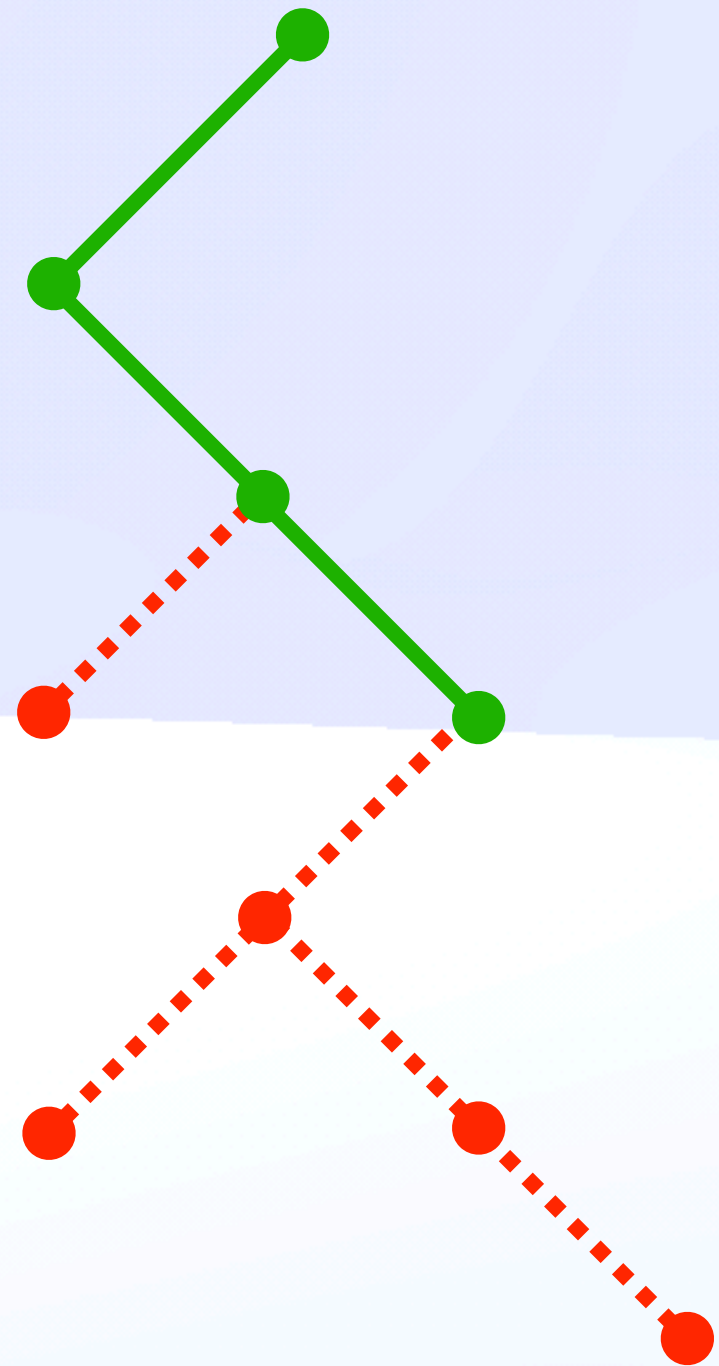


◆ Bloom

◆ CID Roots

◆ CAR

Meta Pull



Previous Round Bloom & Roots



Previous Round Hash Set

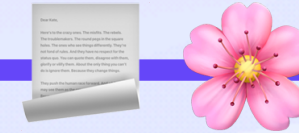
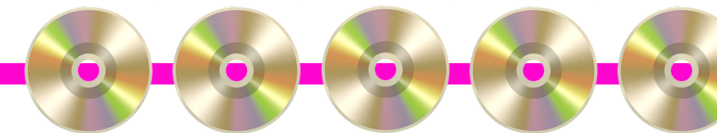
Analyze
New Roots
New Bloom

Analyze
& Fetch

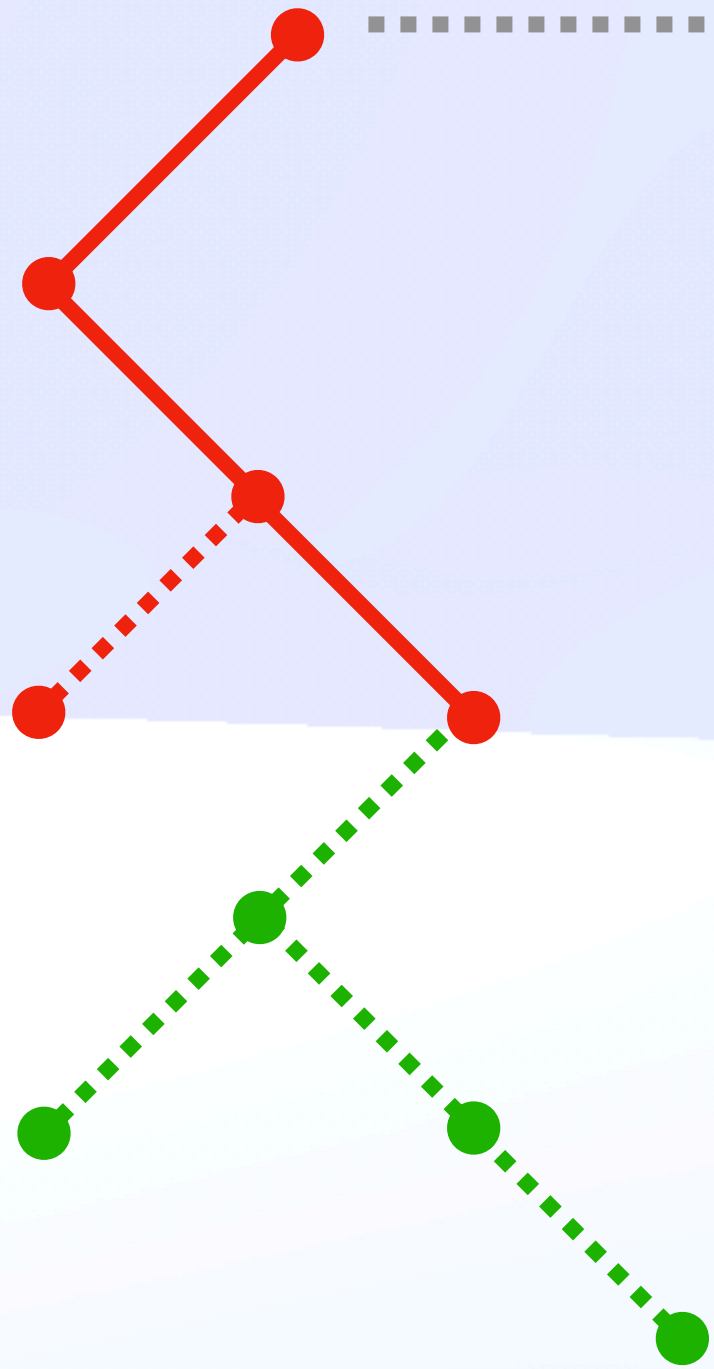
Response CAR

Updated Bloom & Roots

Updated Hash Set



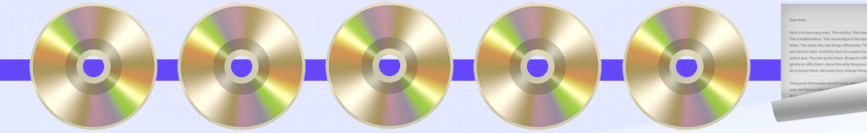
Meta *Push*



Previous Round
Bloom & Roots



Request CAR
DAG Bloom



Previous Round
Hash Set



Store &
Analyze

Bloom &
CID Roots

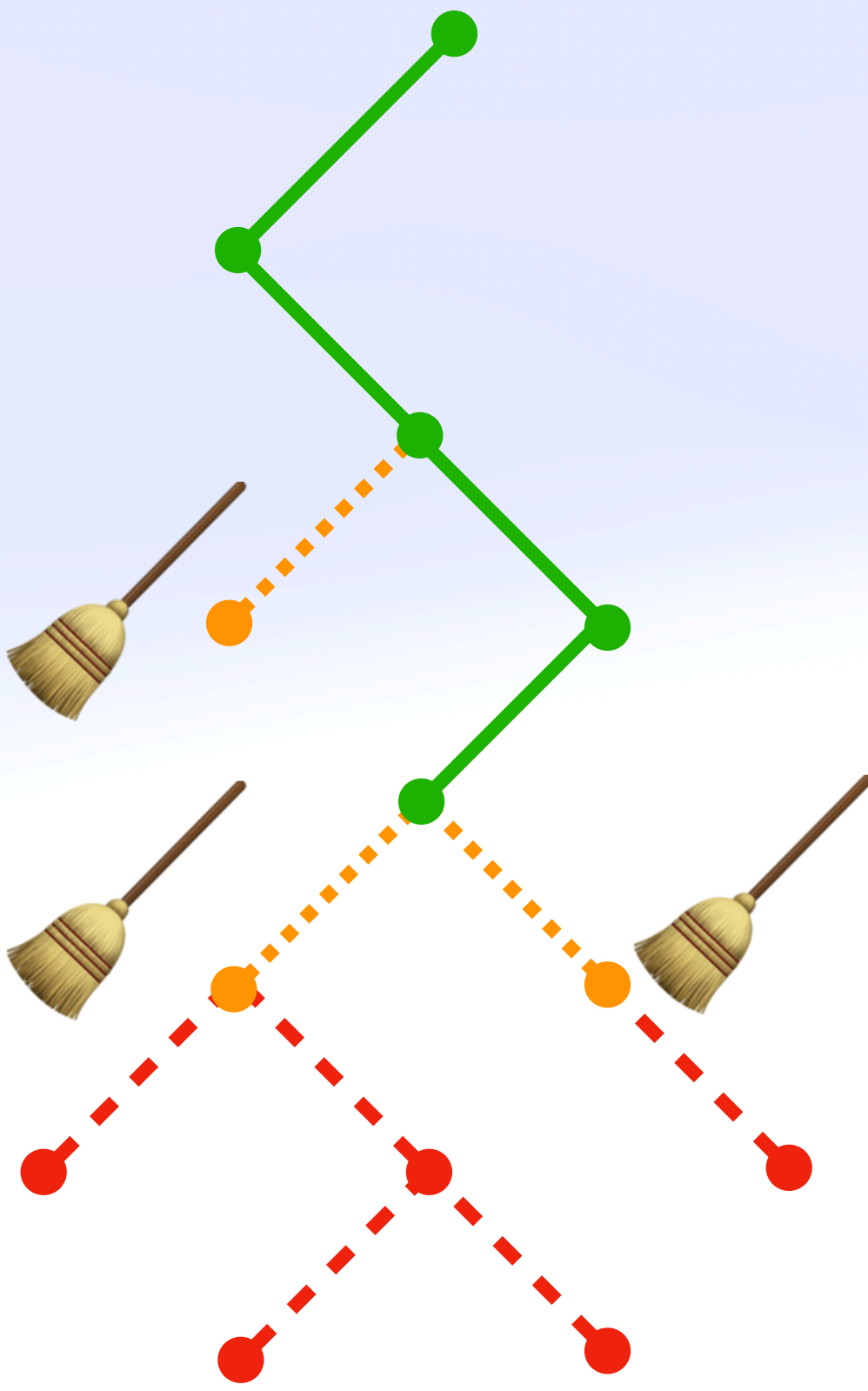
Updated
Bloom & Roots

Updated
Hash Set



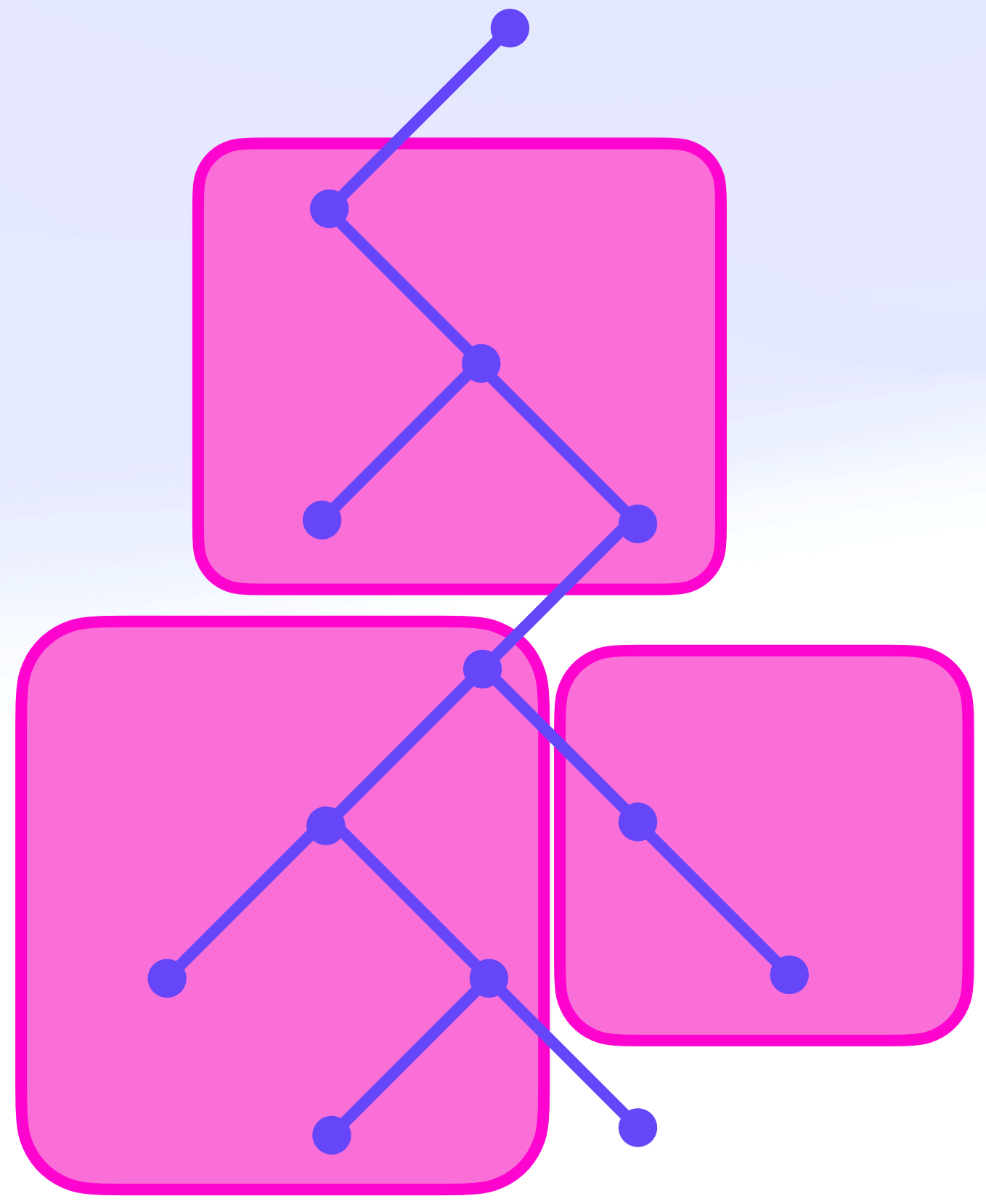
CAR Mirror

Straggler Cleanup



CAR Mirror

Graph Contraction



CAR Pool

Many-to-One

CAR Pool

Why

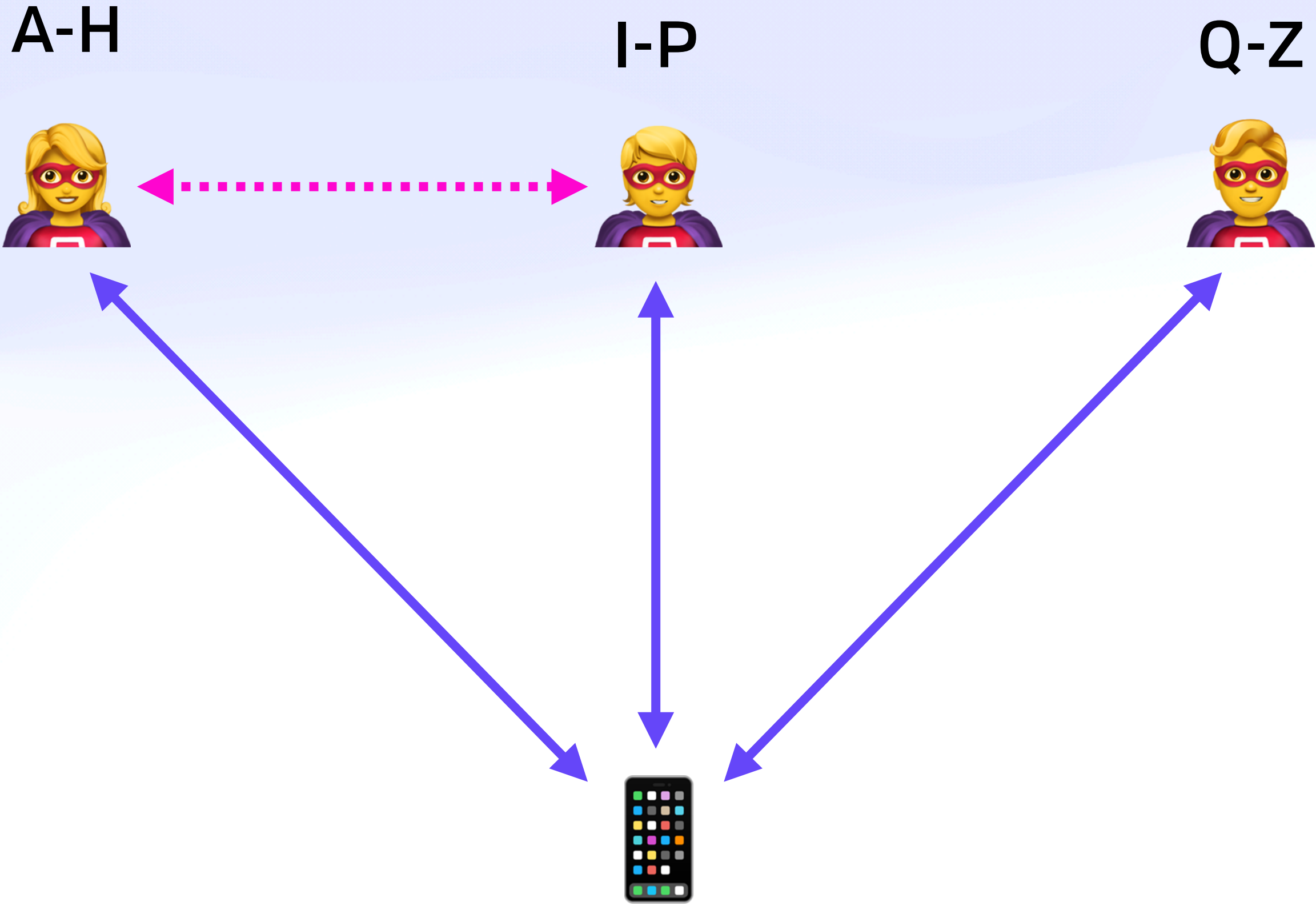
CAR Pool

Why

- ◆ Recover multiple providers
- ◆ Stream in true parallel
- ◆ Work sharing
- ◆ Stepping stone to trustless network seeding (more later)

CAR Pool

Requestor-Controlled Sharding



CAR Pool

Invertible Bloom Filter (IBF) Primer

CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

XOR	3	XOR	2	XOR	6	XOR	4	XOR	7	XOR	5	XOR	2
-----	---	-----	---	-----	---	-----	---	-----	---	-----	---	-----	---

CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

XORed CIDs



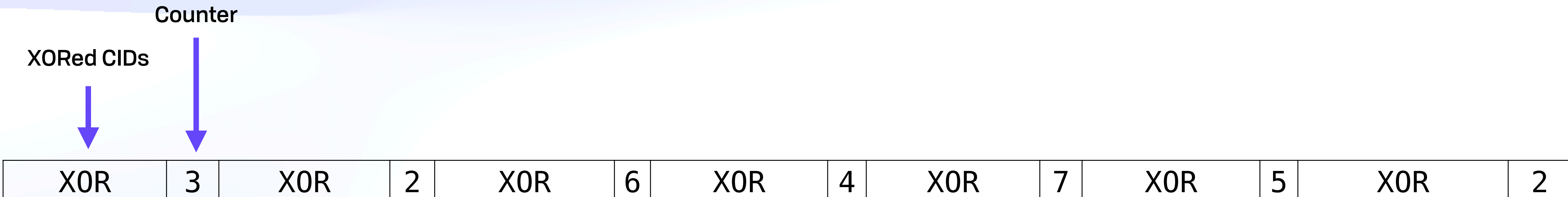
XOR	3	XOR	2	XOR	6	XOR	4	XOR	7	XOR	5	XOR	2
-----	---	-----	---	-----	---	-----	---	-----	---	-----	---	-----	---

CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

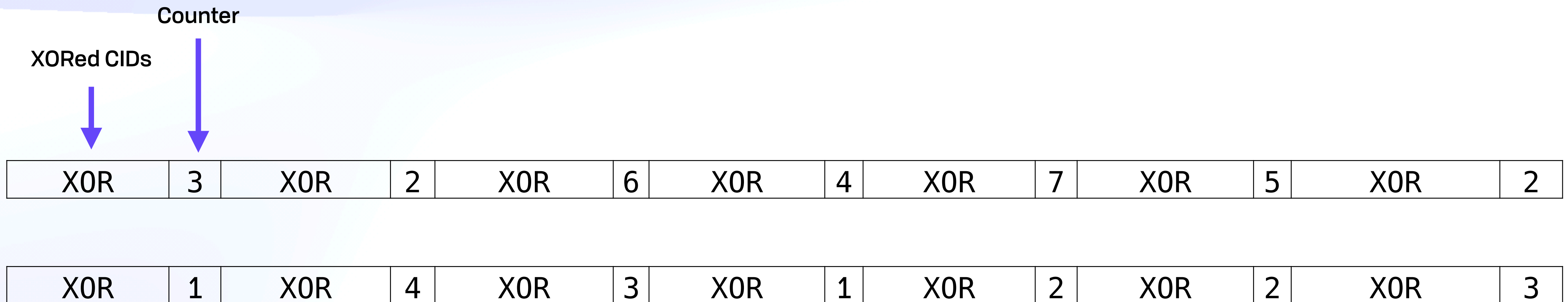


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

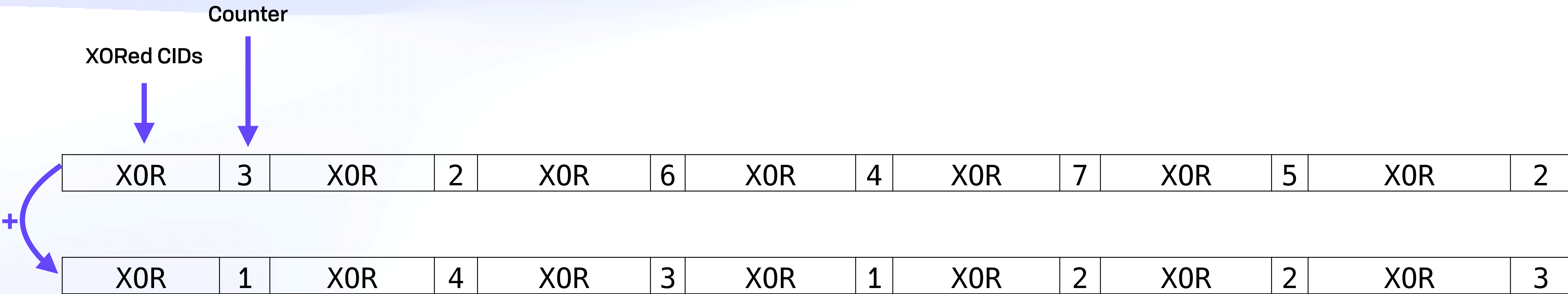


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

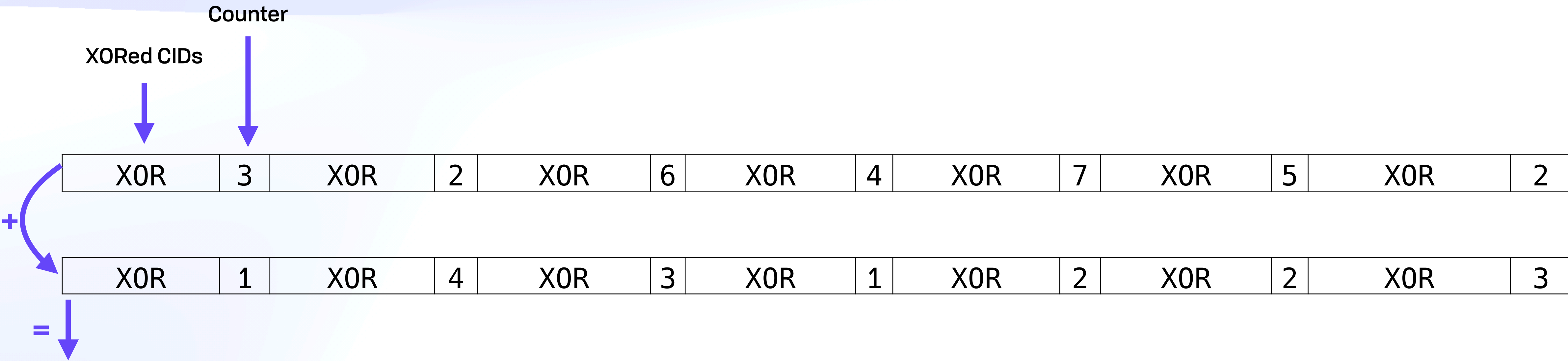


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

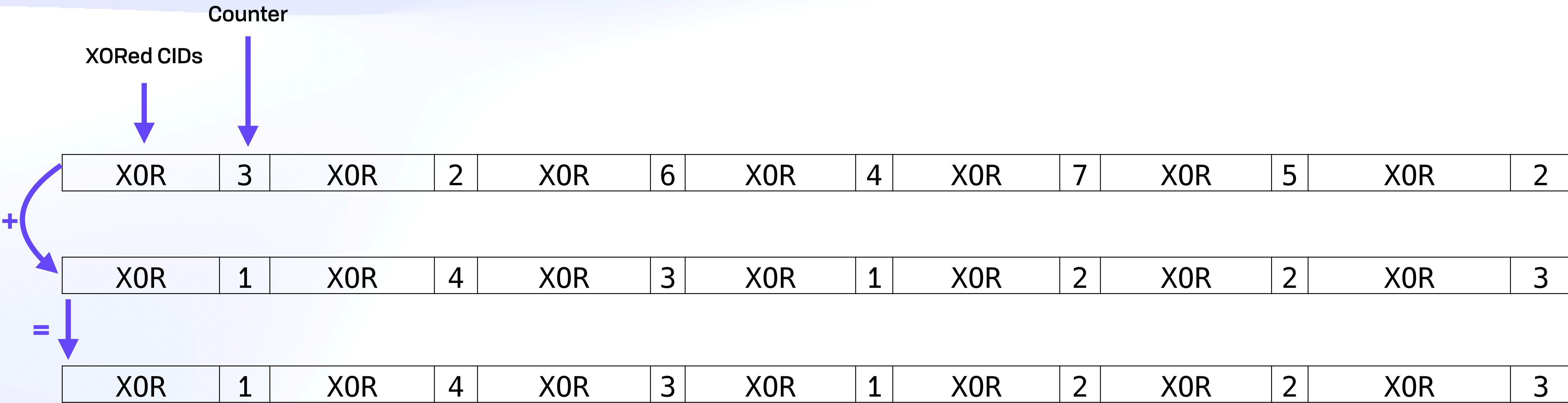


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

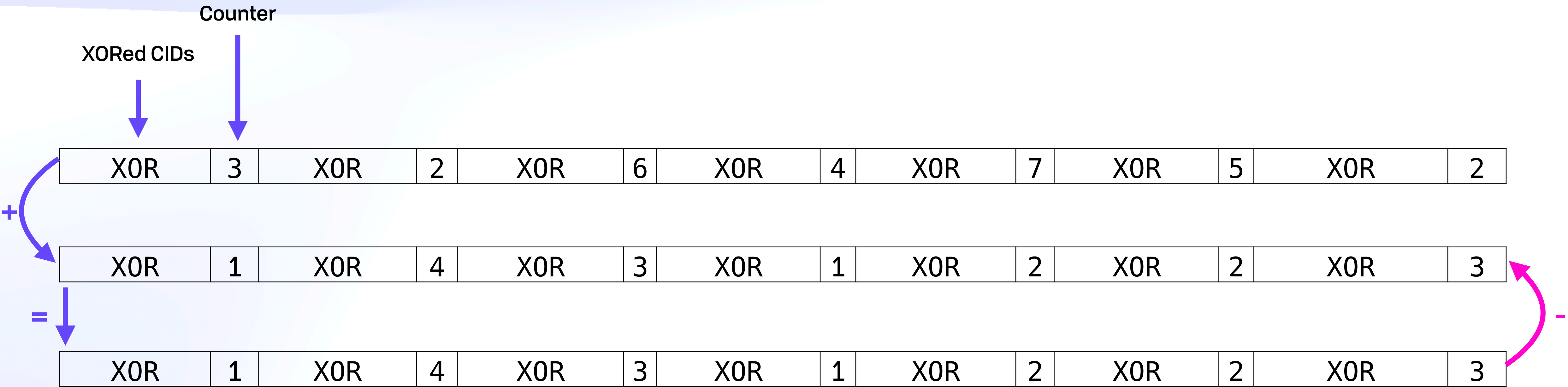


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

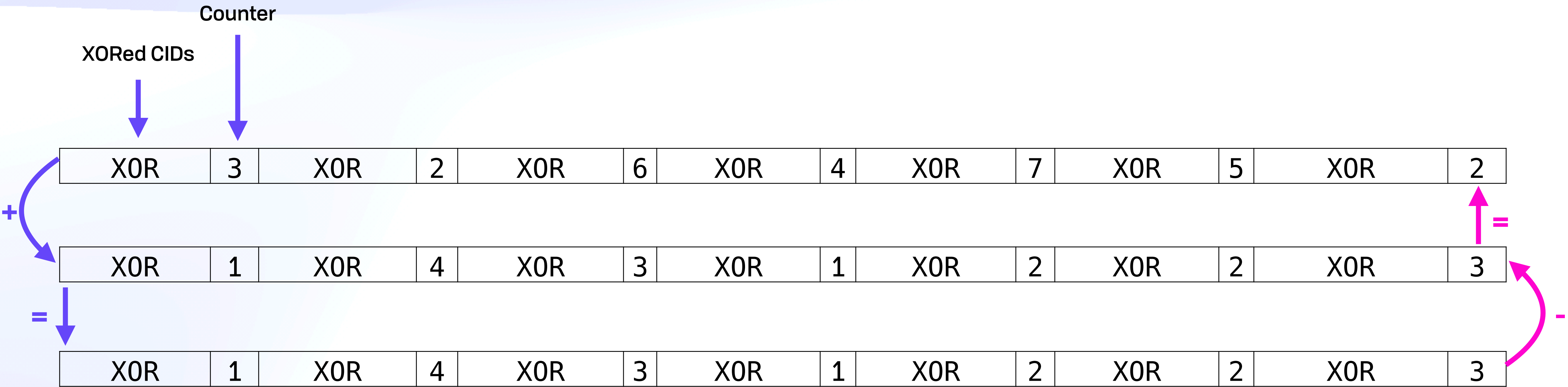


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

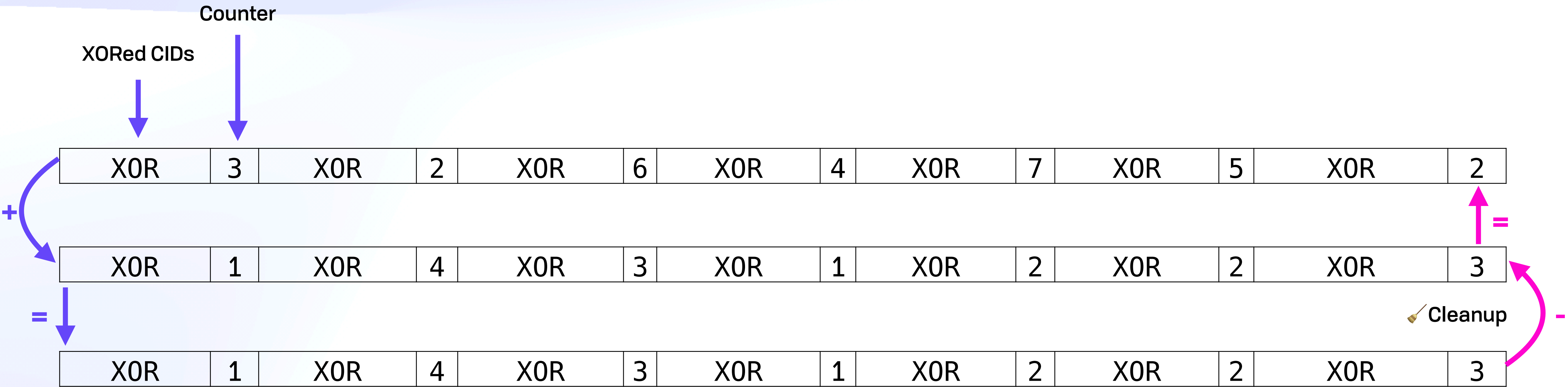


CAR Pool

Invertible Bloom Filter (IBF) Primer

1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )

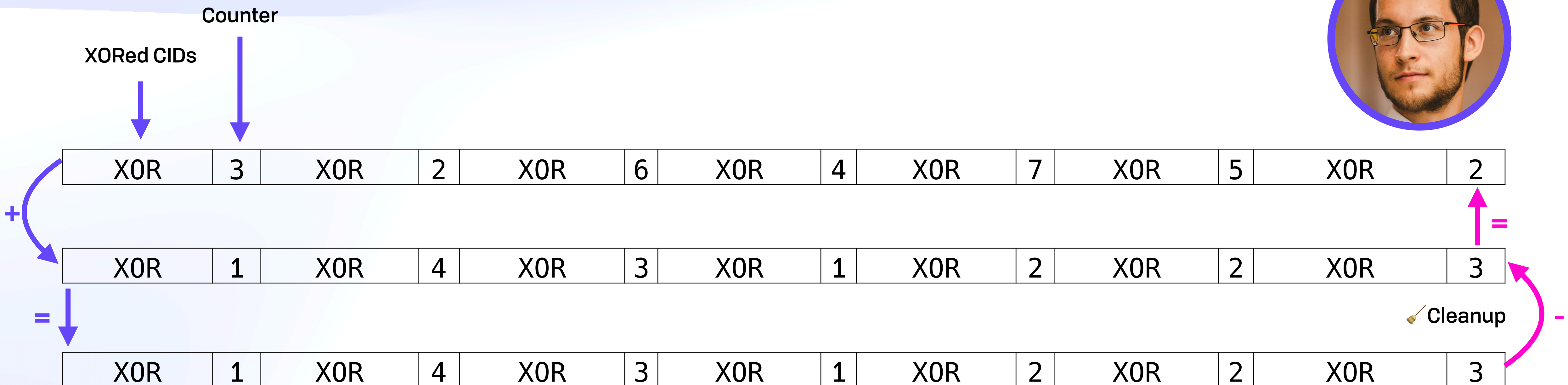


CAR Pool

Invertible Bloom Filter (IBF) Primer

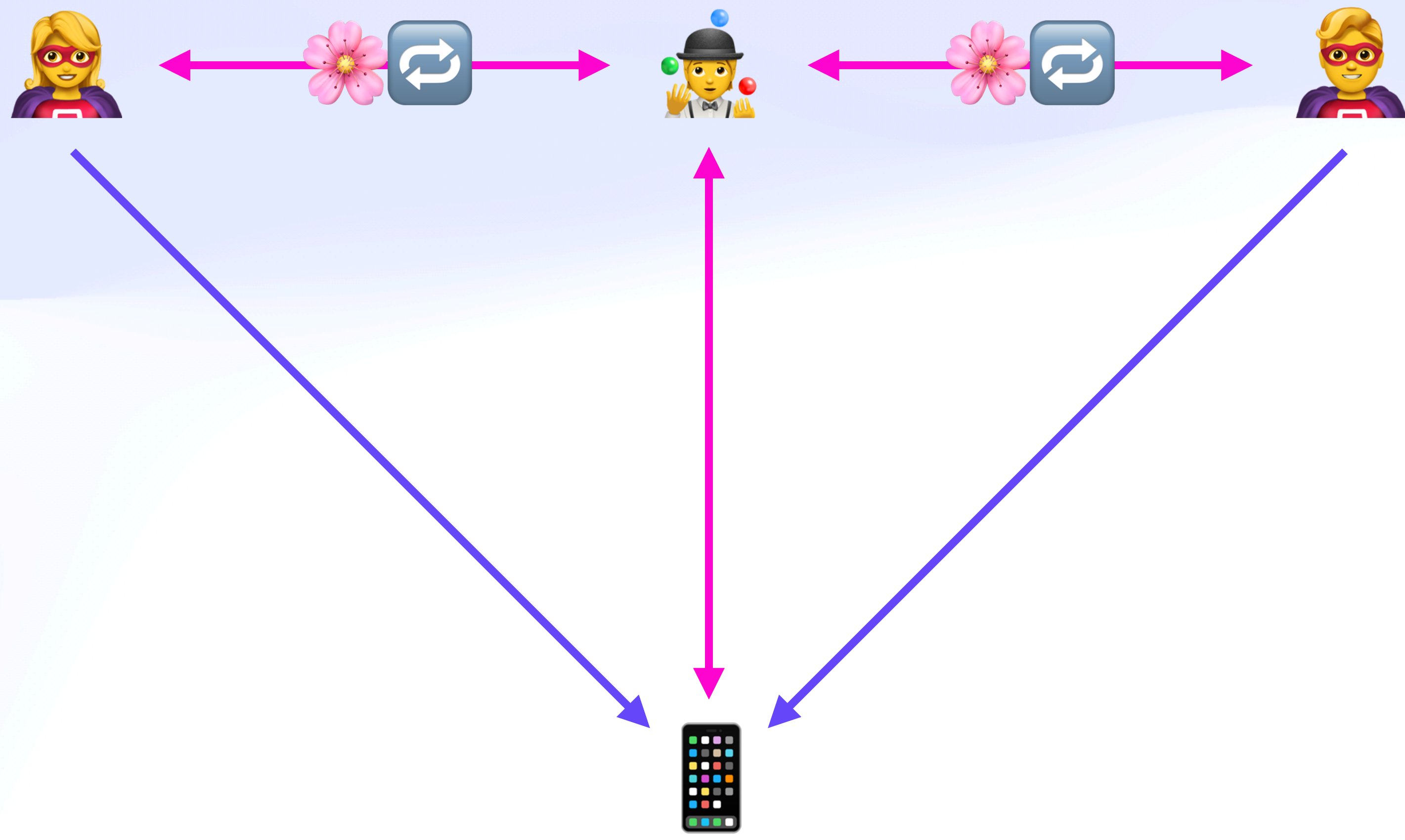
1	0	1	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(Not to scale )



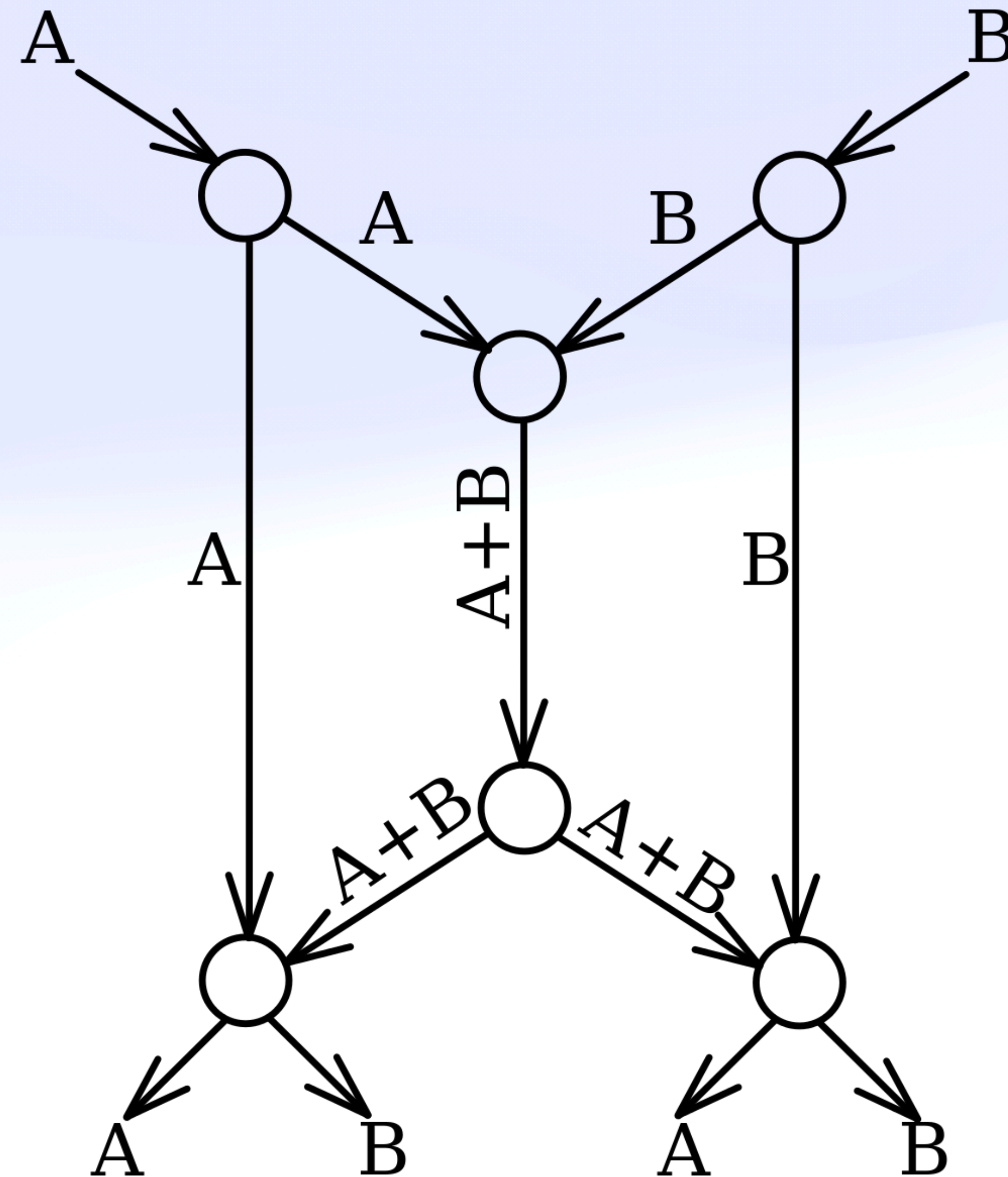
CAR Pool

Provider Coordinator



CAR Pool

Linear Network Coding



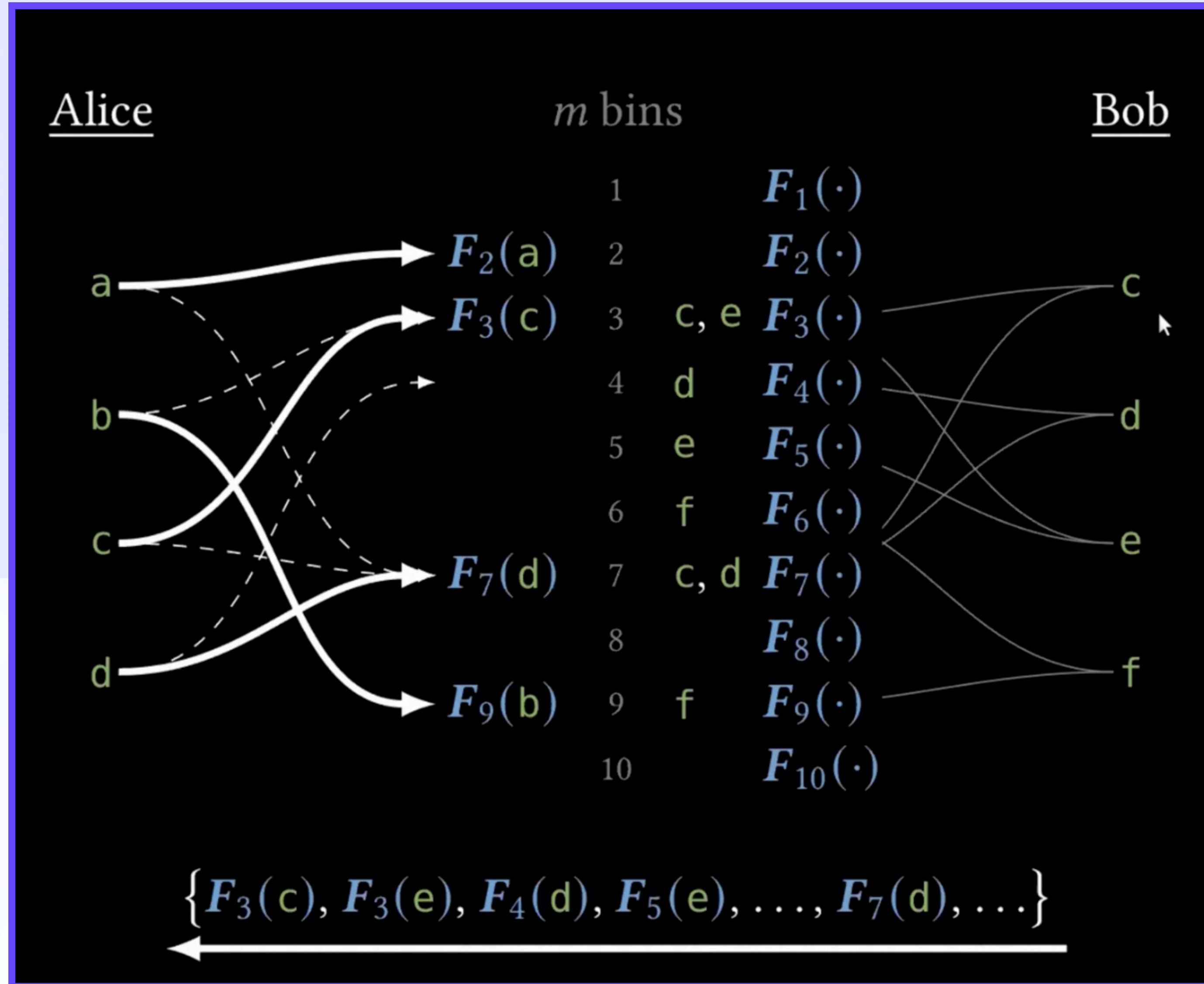
BONUS:

Wild Ideas for Future Research

PSI, PSU, NN Compression, XOR-Folding, etc

CAR Pool

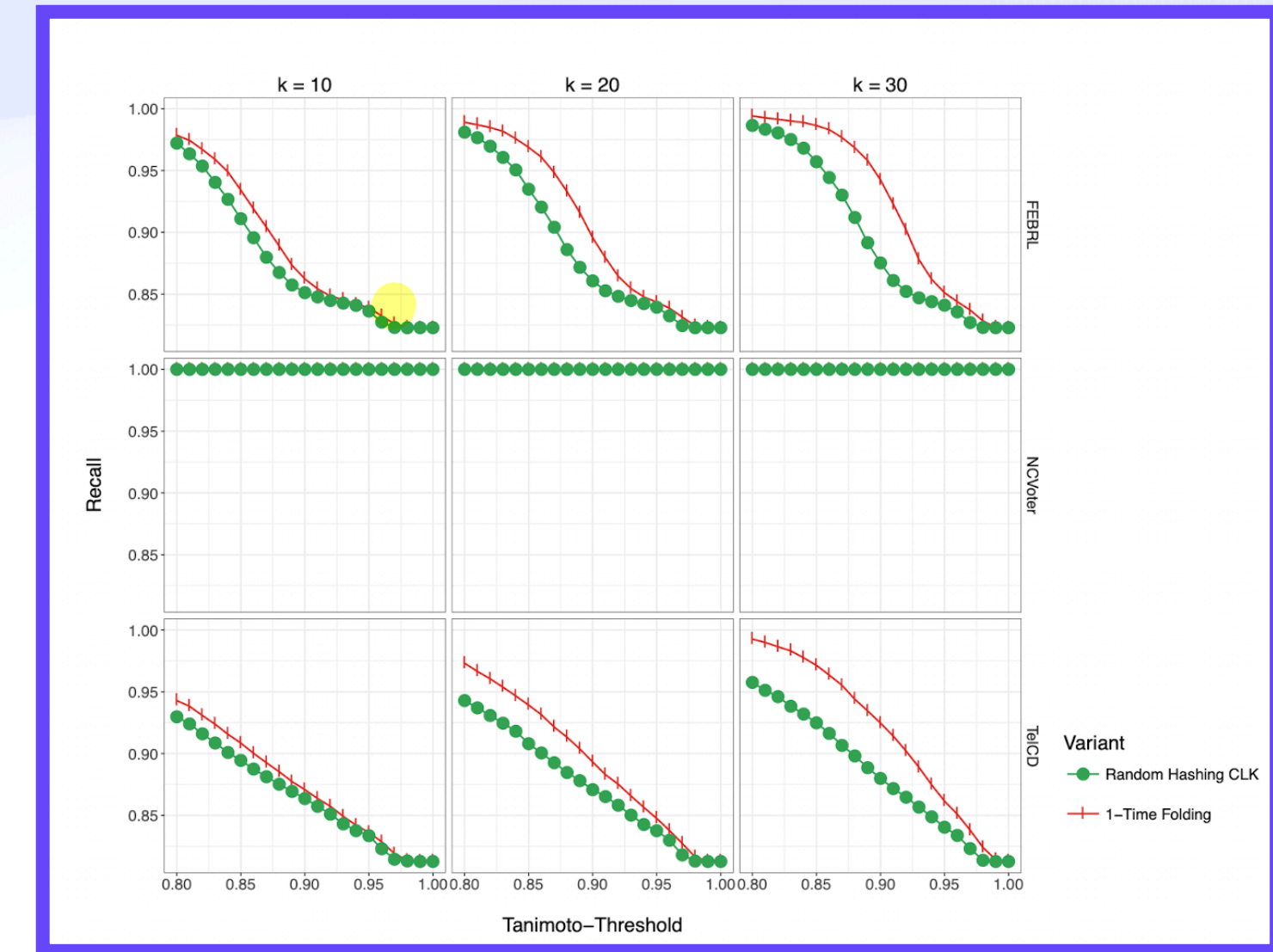
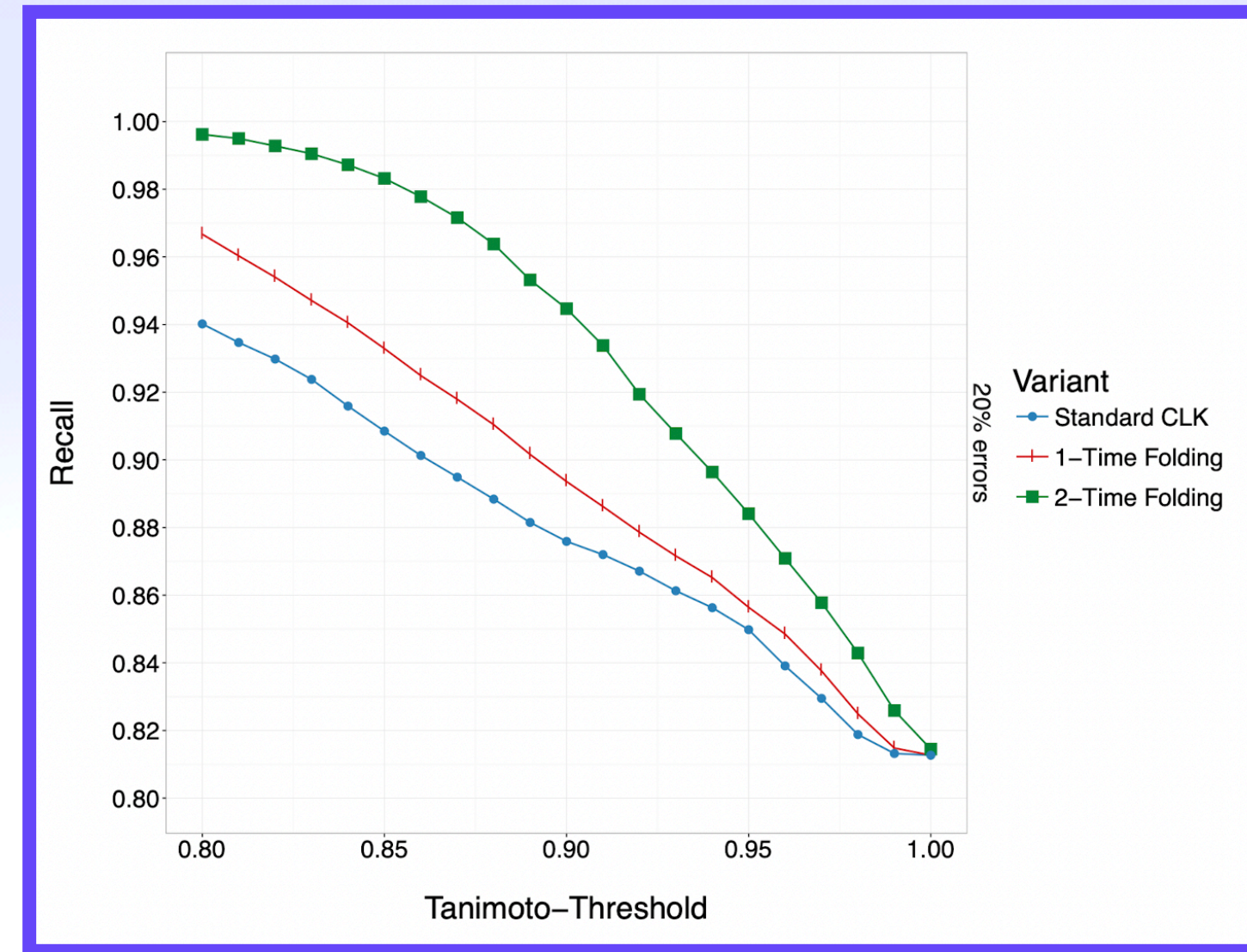
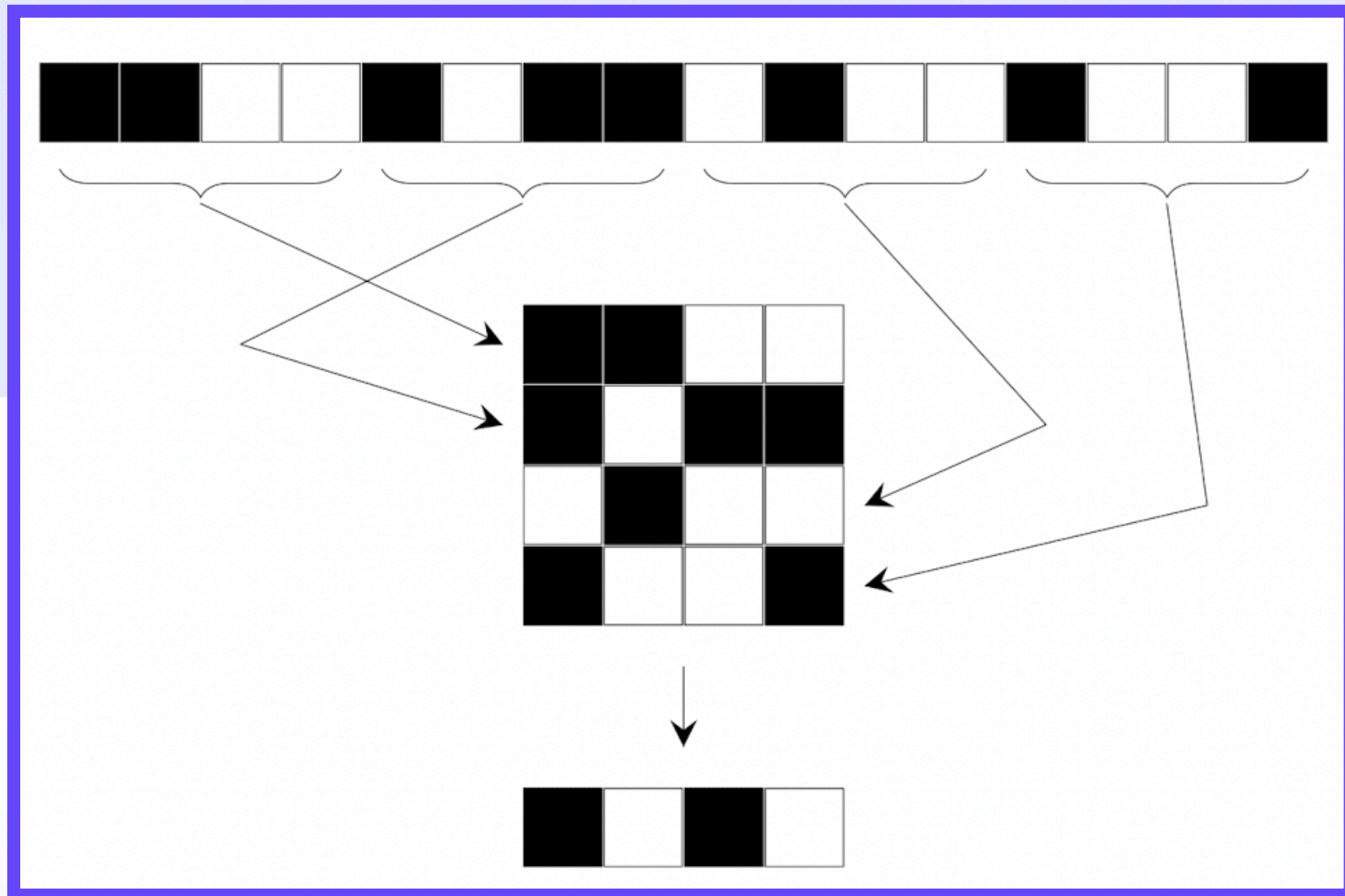
PSI



[youtube.com/watch?v=tsCD4i9SpxA](https://www.youtube.com/watch?v=tsCD4i9SpxA)

CAR Pool

XOR-Folding



https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3527984

An Eager Backbone 🍖

Content Addressed Alliance *Teaser*



 ***Thank You, IPFS ping*** 

<https://fission.codes>

✨ github.com/fission-suite/spec ✨

{brooklyn,justin}@fission.codes

@expede & @justincjohnson