HTTP Prioritization for Product Performance

Introduction

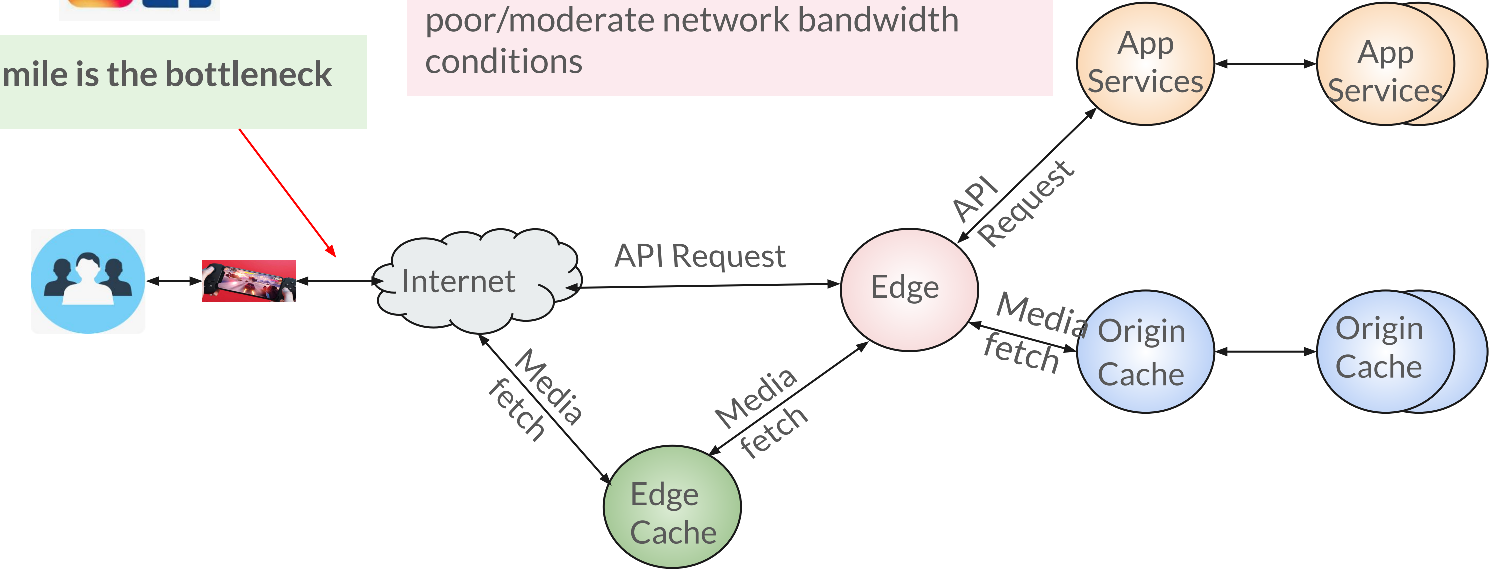
1. Meta HTTP Application E2E Architecture
2. A/B Testing for User Performance Optimization
3. Success Stories on using HTTP/3 Prioritization for Meta User Performance

High Level Overview on Meta Products Deployed on the Internet

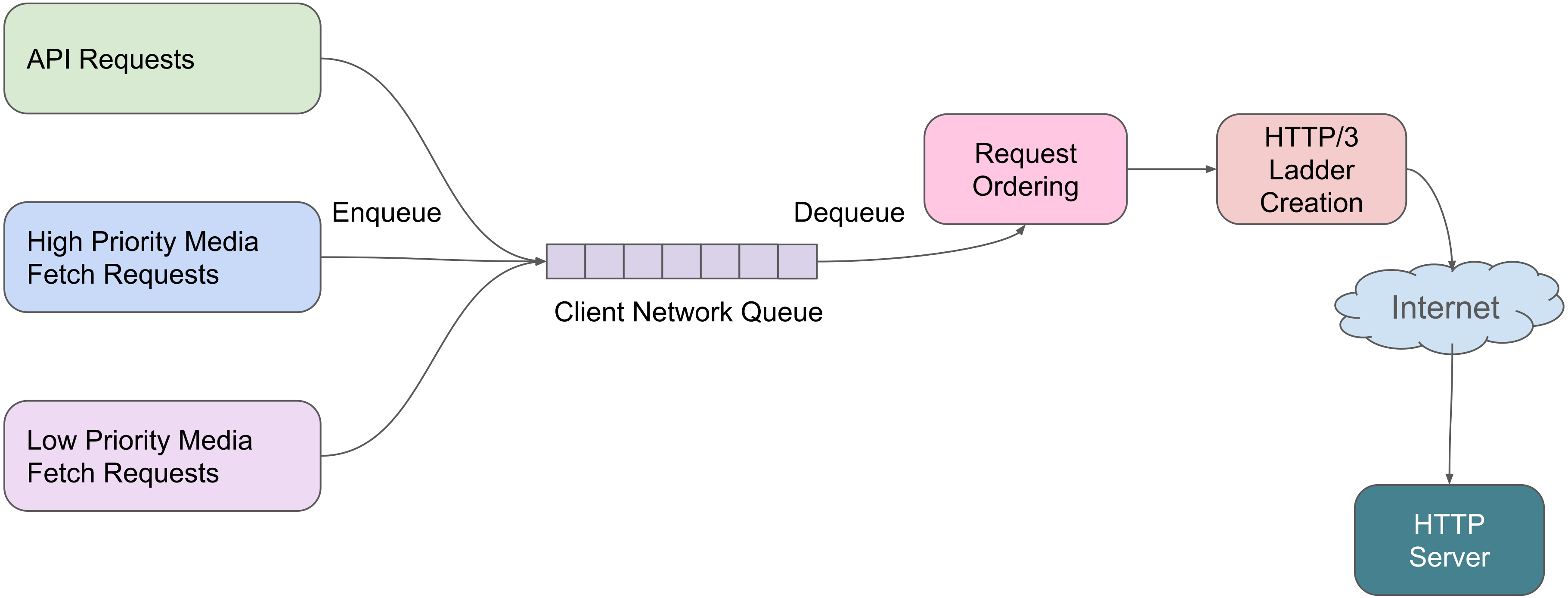
Challenges: Large number of overlapping HTTP requests going through the network

Challenges: Large user populations in poor/moderate network bandwidth conditions

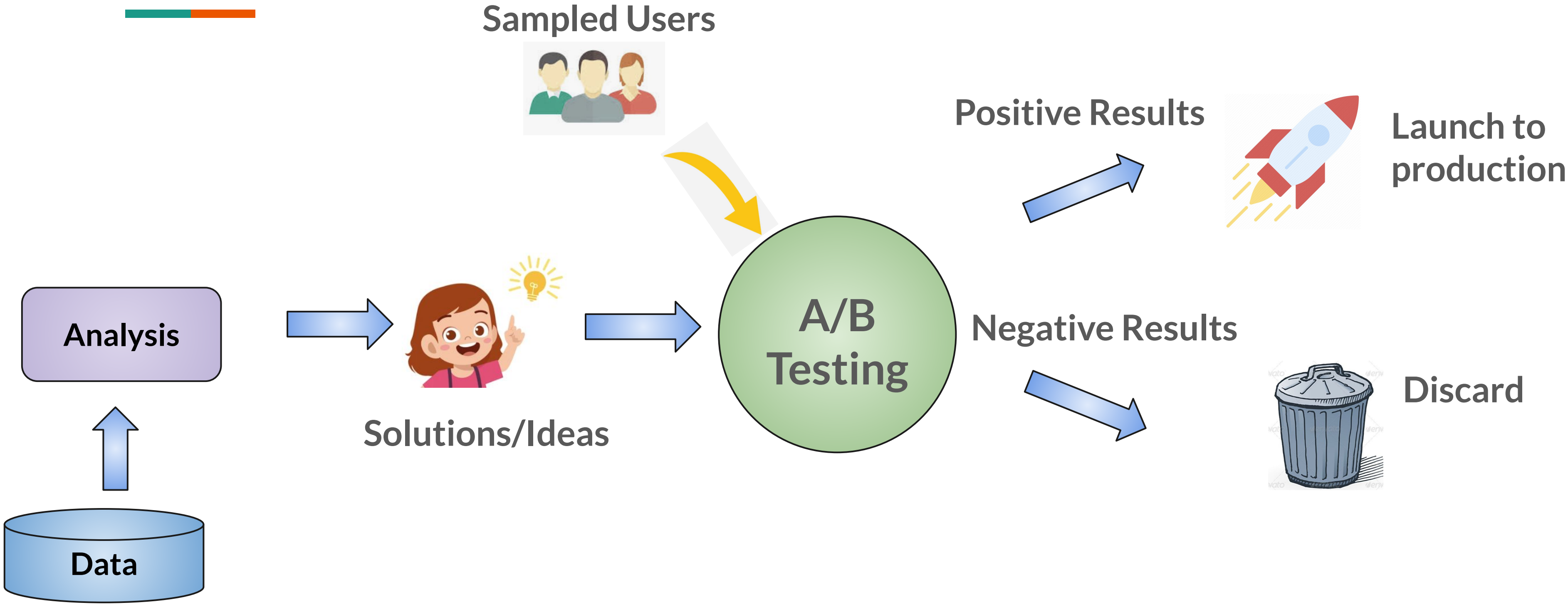
Last mile is the bottleneck



Requests Prioritization Flow



A/B Testing for User Performance Optimization



Sampled Users

Positive Results

Launch to production

Analysis

Solutions/Ideas

A/B Testing

Negative Results

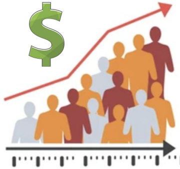
Discard

Data

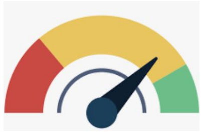
Success Stories - HTTP/3 Prioritization is Effective

Bypassing Client Network Queue

Tuning HTTP/3 prioritization based on Application logic



DAU and Revenue Wins



Video/Image Performance Gains

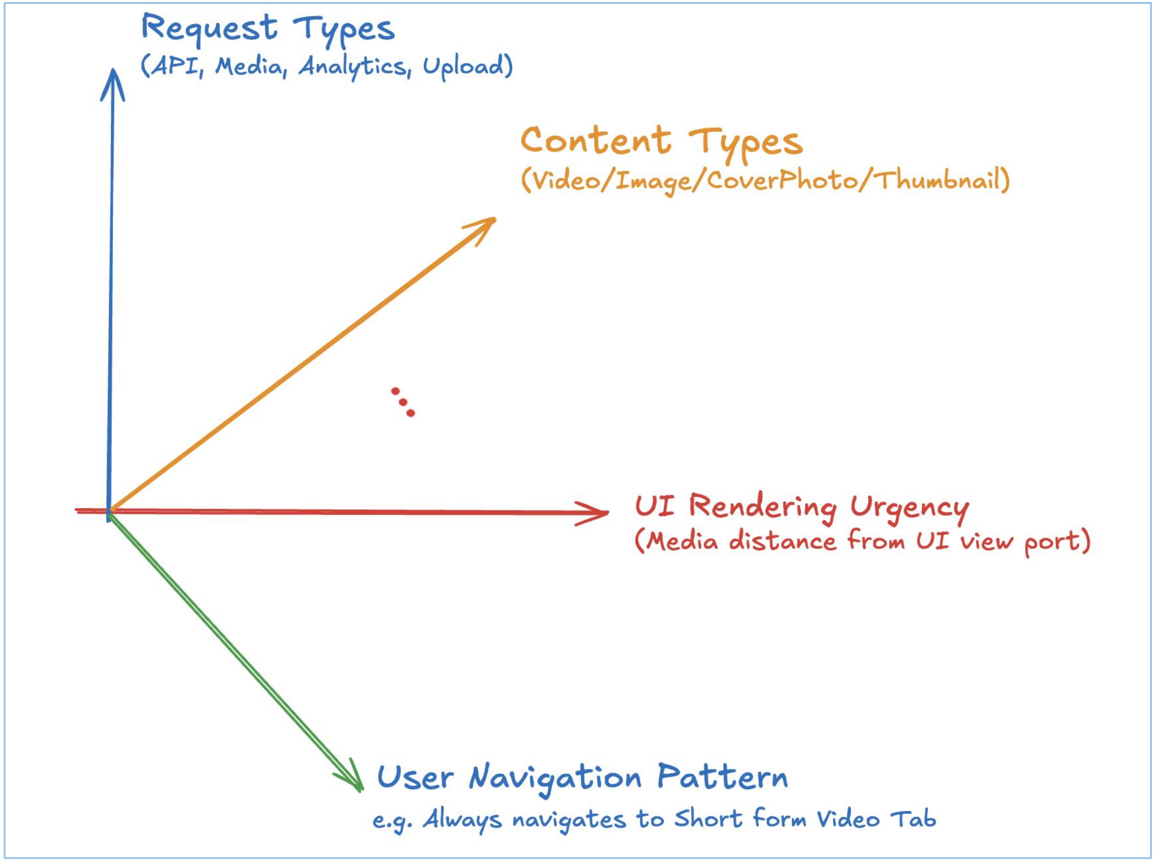
User Engagement Gains

Use Cases & Challenges

1. Complex Product Requirements for HTTP Prioritization

2. HTTP Prioritization Use Cases & Design Challenges

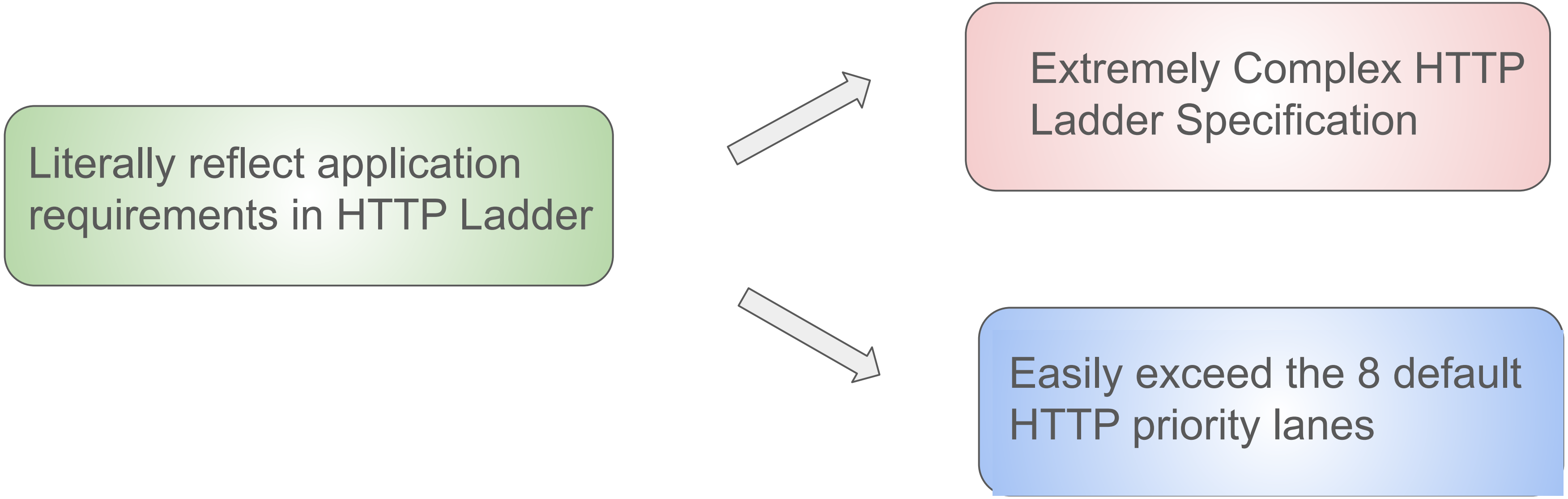
Multi-Dimensional Complex Application Requirements for HTTP Prioritization


Request Types
(API, Media, Analytics, Upload)
Content Types
(Video/Image/CoverPhoto/Thumbnail)
UI Rendering Urgency
(Media distance from UI view port)
User Navigation Pattern
e.g. Always navigates to Short form Video Tab

The complexity:

1. Different types of requests have different priority
2. Different media contents have different priority
3. Different distances to UI rendering viewport have different priority
4. Different network condition generates different prioritization sensitivity

Challenge 1: HTTP ladder - Simple or Complex?

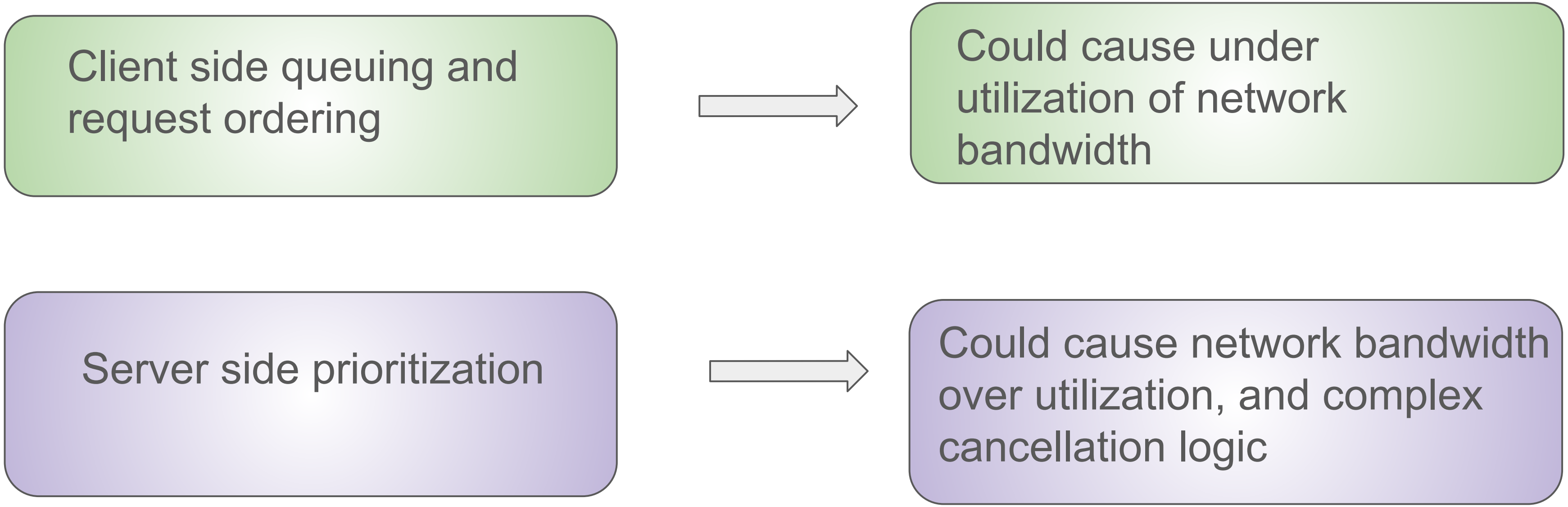


Literally reflect application requirements in HTTP Ladder

Extremely Complex HTTP Ladder Specification

Easily exceed the 8 default HTTP priority lanes

Challenge 2: Client or Server Prioritization?

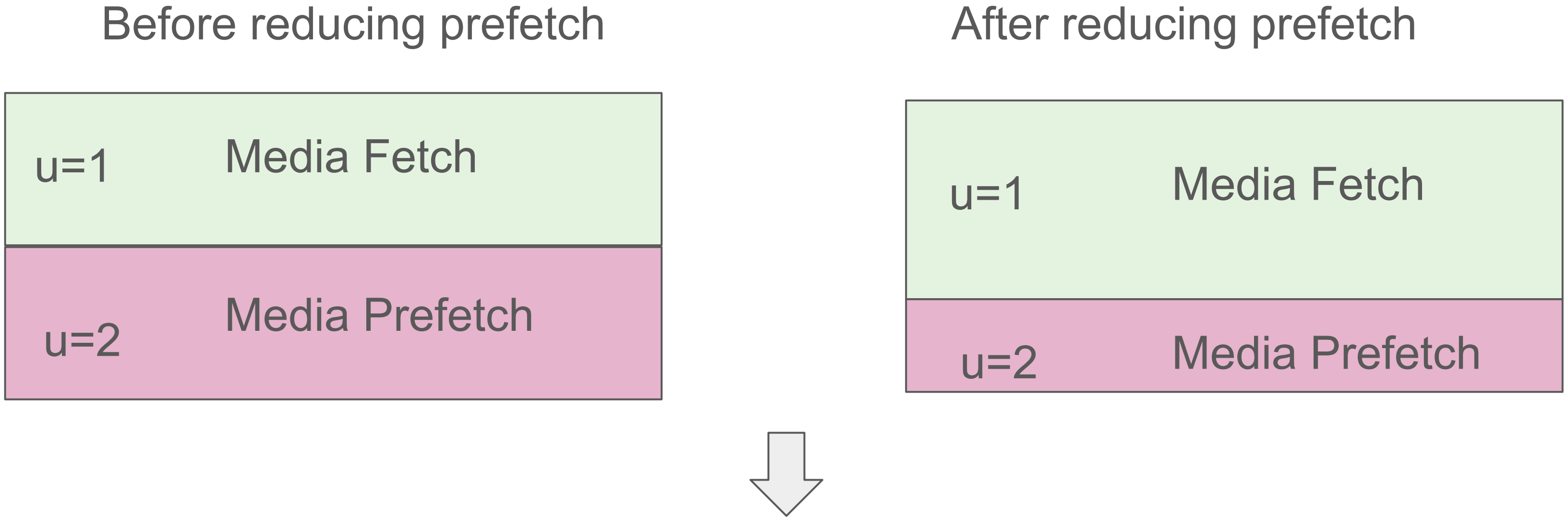


Client side queuing and request ordering → Could cause under utilization of network bandwidth

Server side prioritization → Could cause network bandwidth over utilization, and complex cancellation logic

Challenge 3: Bandwidth Quota for each Lane?

Example Use Case : Egress Efficiency, i.e. reduce egress volume



Before reducing prefetch

u=1 Media Fetch

u=2 Media Prefetch

After reducing prefetch

u=1 Media Fetch

u=2 Media Prefetch

Ineffective egress reduction

Questions for the Workshop

1. Would it be good idea to lift the default limit of 8 urgency lanes?

2. Any feedback, suggestions, and experience/knowledge sharing on whether client side or server side HTTP request prioritization is better?

3. Would assigning a quota to each HTTP lane be a way to avoid one urgency lane surging scenarios?