

A Deadline-aware Extension for MoQ relays

Chuan Ma

Tsinghua University

Yong Cui, Yixin Liao Hang Shi

Tsinghua University Huawei

Basic agreements on MoQ relays

- Should get access to the header but not the payload
- Should Forward packets
- Can schedule or reorder the packets
- May drop packets under some circumstances
- Maybe other features ...

Why do we need an extension?

- Two dimensional priorities in a single space
 - Time: newer > older
 - Importance: key frame > predict frame, audio > video
- Global priority v.s. Local priority
 - Single priority is difficult to define global priority
- Relay don't know when to give up 'over-queued' object
 - How large should the cache or the jitter buffer be?
- **So, We need some indicators of timeliness**

Timeliness parameter

- Indicate the application's requirement and intention
 - “Deadline”, “time-to-live”, “maxacceptable delay”
- Relays should be aware of it to make better decision
 - Especially in **real-time** transmission

Timeliness parameter in moqt

- An optional field in the object header
 - Deadline: Unix-timestamp (currently)
 - Or other ?
- Some new functions of flags
- Other extensions are similar

```
object_header {  
    ...  
    id(i),  
    message_type(i),  
    object_id(i),  
  
    [nb_objects_previous_group(i),]  
    flags[8],  
    object_length(i),  
    [deadline(i)]  
}
```

Deadline-aware MoQ Relay Components

- Data Unit: Block(Object)
 - basic data unit, containing single frame
- Metadata: Deadline
- Actions
 - Scheduling
 - Dropping
 - Redundancy coding

Deadline-aware MoQ Relay Demo

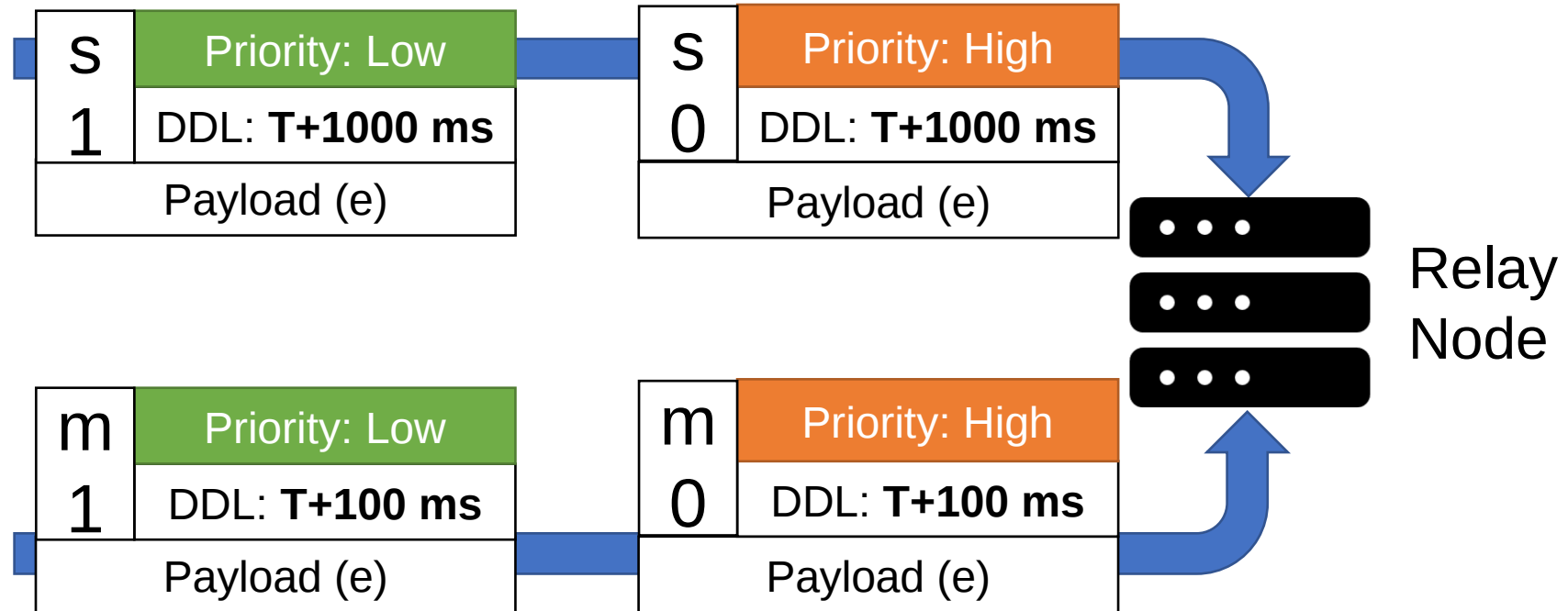
- Publishers emit objects with deadline
 - Showing the expiration time and expect latency
 - Deadline is given in Unix timestamp



Expect latency: <1000 ms

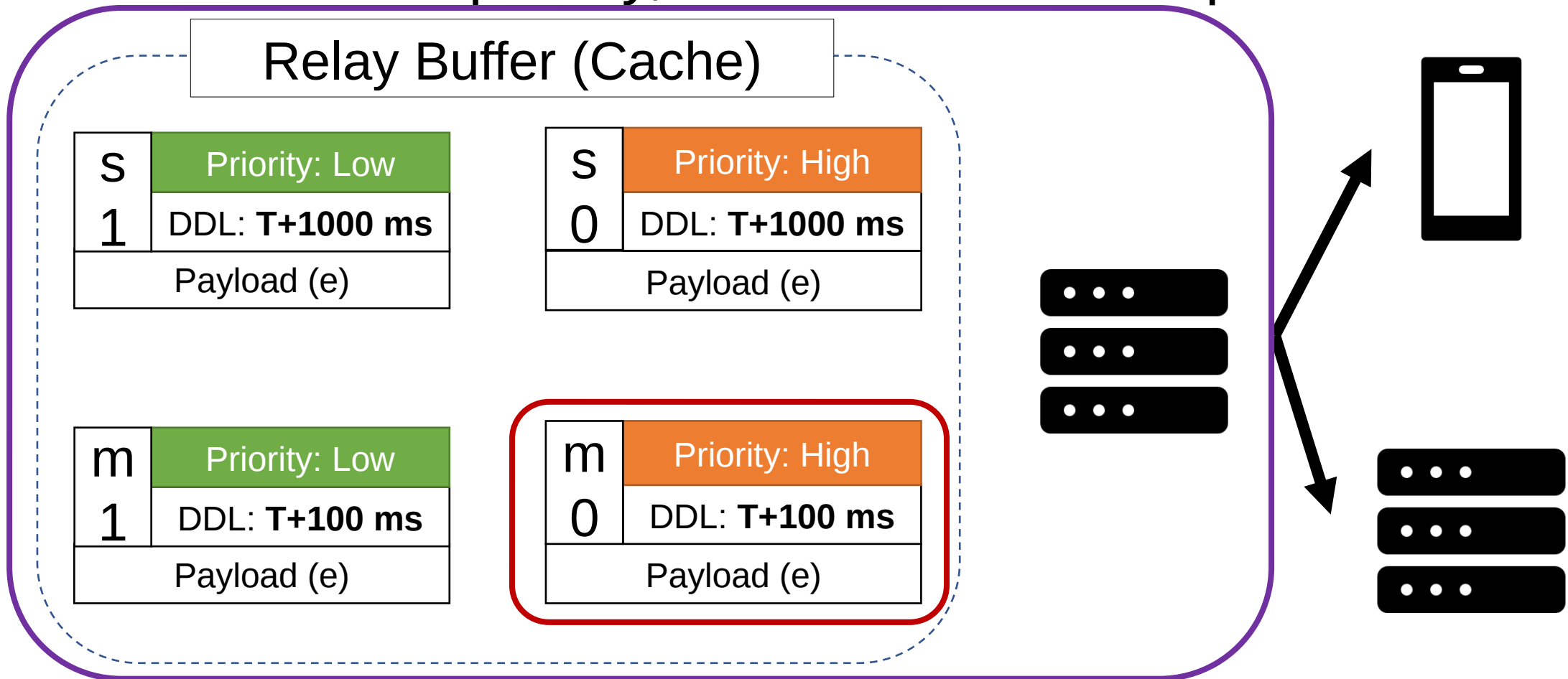


Expect latency: <100 ms



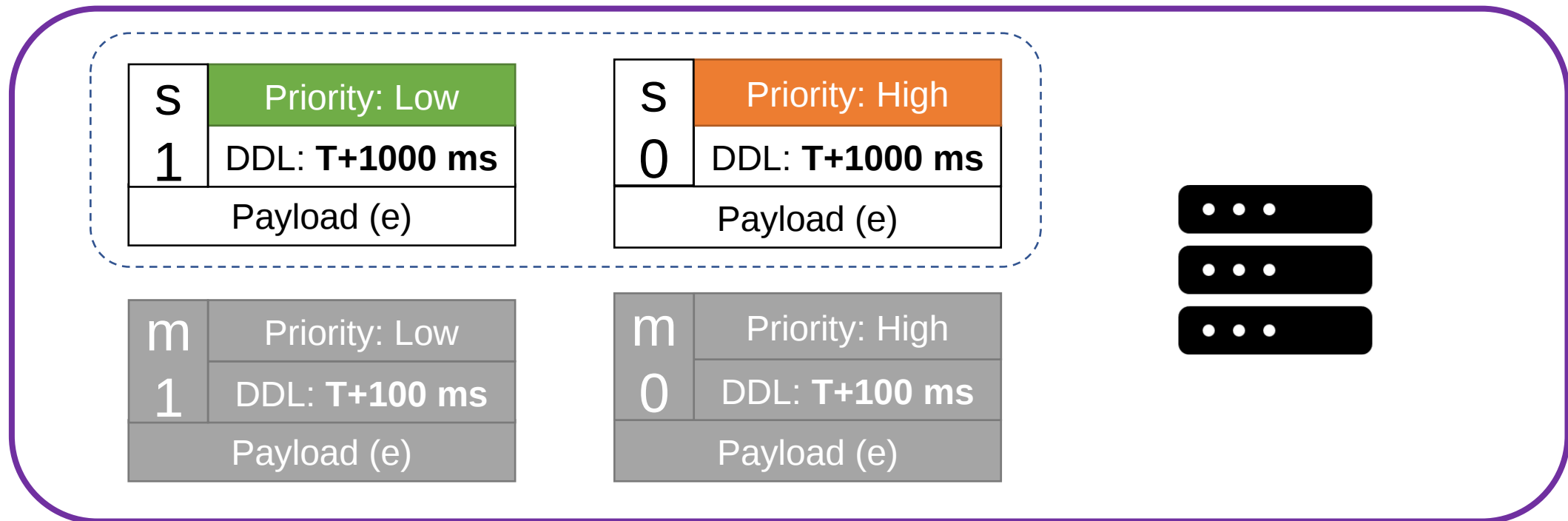
Deadline-aware MoQ Relay Demo

- Schedule with priority, deadline and other parameters



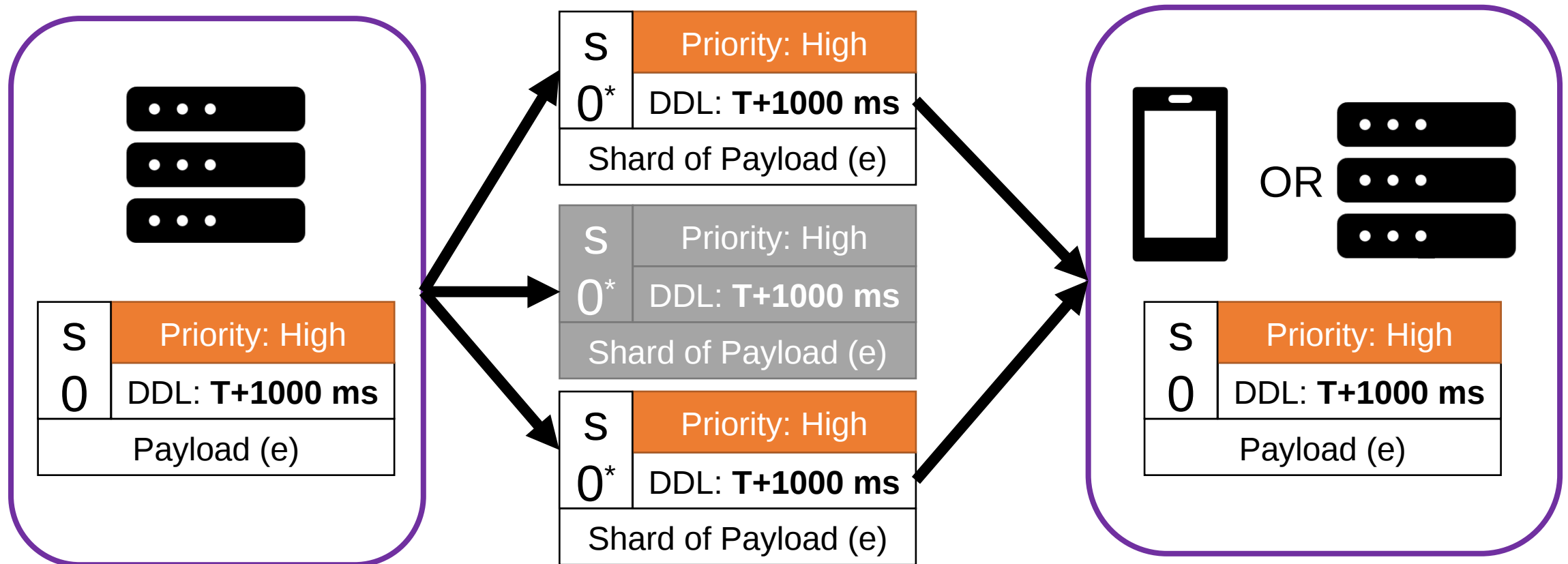
Deadline-aware MoQ Relay Demo

- Relay drops overdue objects (no longer useful)
 - Dropping logic may relate with priority setting
 - Immediate drop or Lazy drop depends on other designs



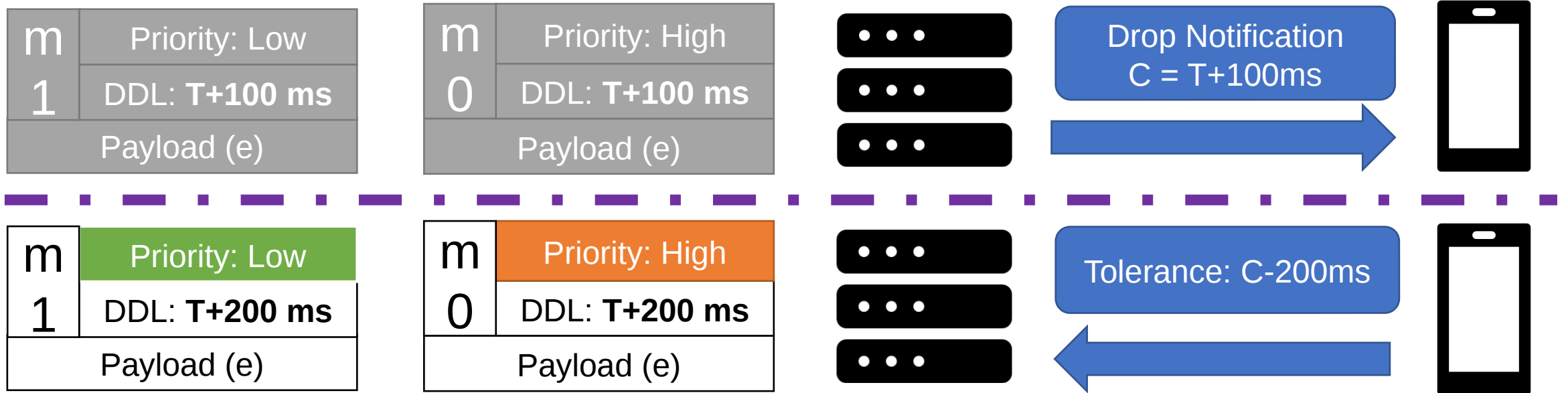
Deadline-aware Optional Functions

- [Optional] Add FEC if needed
 - When the network is bad and both ends of a hop have codec



Deadline-aware Optional Functions

- [Optional] Drop notification and Subscribers request
 - Relay notify the subscriber about the drop (control message)
 - Subscribers request with Tolerance (update DDL)



The potential benefits of Deadline-aware

- Lower latency, Save bandwidth
- Increase real-time performance of MoQ
- Help the relay make smart decisions

p.s. still need measurement and confirmation

Conclusions

- Timeliness Header Parameter
 - Some parameters like deadline which shows the freshness or validity of data should be added to object header
- Deadline-aware MoQ Relay
 - Relays get better knowledge of app's timeliness requirement
- Future Work
 - The detailed implementation of deadline-aware relay still needs further discussion, test and measurement
 - Update the draft

Thanks

Q&A