Constrained RESTful Environments WG (core)

Chairs:

Jaime Jiménez <jaime.jimenez@ericsson.com>

Carsten Bormann <cabo@tzi.org>

Mailing List:

core@ietf.org

Jabber:

core@jabber.ietf.org

- **We assume people have read the drafts**
- **Meetings serve to advance difficult issues by making good use of face-to-face communications**
- **Note Well: Be aware of the IPR principles, according to RFC 8179 and its updates**

üBlue sheets
üScribe(s)

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/ (Privacy Policy)

Agenda Bashing

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:52 Active drafts**
- **11:52–12:06 FASOR**
- **12:06–12:20 Streaming**
- **12:20–12:20 Other new work**

Advertisements

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

Draft-ietf-core-senml
→ RFC 8428

✓

2018-08-31

draft-ietf-core-links-json: Status

- JSON version of 6690-to-be — avoid need for another parser
  - Started Feb 2012, added CBOR variants mid-2015
- Focus was: roundtrippable with RFC 6690
  - Inherit limitations of RFC 6690 (e.g., percent-encoding)
- Submitted to IESG on 2017-04-02: Lots of feedback
- Re-focus:
  - Still cover all of RFC 6690
  - Be more general, don't inherit the limitations
- More recent discussion:
  points to CORAL as the more likely ultimate target

draft-ietf-core-cocoa: Status

- Submitted to IESG 2017-12-16
  - Responsible AD here: Mirja Kühlewind (TSV AD)
  - Great AD feedback
- London IETF uncovered potential for misunderstanding
  Ran out of time resolving this in Montreal IETF
  Still not resolved, try again this week
  Will lead to –04

- CoCoA is not the end-all of congestion control work for CoAP
- Proposed new work: draft-jarvinen-core-fasor
  (Thu)

draft-ietf-core-object-security: Status

- Submitted to IESG 2018-02-15
- Revisions –11, –12 based on IESG comments done in March
- continuous minor updates –13, –14, –15 since
- Still blocked on one remaining DISCUSS

Too Many Requests Response Code for CoAP

draft-ietf-core-too-many-reqs-05 (and TBD -06)

IETF 103

Intro clarifications

- Why using Max-Age and not new option?
  - 5.03 is using Max-Age like this already
  - Proxy caching rules for Max-Age map nicely
- Clarified that this draft is not defining “new” Max-Age use
- Should probably also update IANA registry references for the option

Server behavior clarifications

- If client does not respect back-off from 4.29, server MAY respond with 5.03.
- “Server MAY also limit how often it answers to a client, e.g., to once every estimated RTT”
    - New proposal: “Server should rate-limit 4.29 replies taking into account its usual load shedding policies”
- Note: keeping per-client state may be counterproductive
- Reminding that 4.29 should be sent to client causing overload; 5.03 is appropriate to others

Client behavior clarifications

- How to interpret Max-Age? “Current at time of transmission”
  - Details to be handled in “CoAP clarifications and corrections” draft
  - Clarified that default value expected if missing (defined in 7252)

Security clarifications

- CoAP RFC’s security considerations apply
- Should trust response only to level one trusts underlying security
- Responses without encryption could leak information about server overload and client traffic patterns
- Noting that dropping requests is likely to make clients retry

Proxy clarifications?

- Many clients behind proxy may look like one client to a server. Too-many-requests reply may go to wrong client.
- How to avoid client being starved by other clients?
  - Can we propose some good proxy behavior?
  - Out of scope for this draft?

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

Comments from Klaus

- 1. "This specification allows to indicate that an optional part is not present by substituting a null value for the representation of the part." -- Do we need this?
- 3.1. -- I see that draft-ietf-core-coap-pubsub-05 is still proposing a new response code (2.07) for this scenario. Will -pubsub switch to multipart-ct as described in this section? If not, better remove the example.

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- 13:50–14:00 Intro, Agenda, Status
- 14:00–14:05 Post-WGLC: Multipart-CT (CB)
- 14:05–14:25 Recently adopted, in adoption
- 14:25–14:40 OSCORE, continued (MT)
- 14:40–15:00 Resource-Directory, Link-Format (CA)
- 15:00–15:20 Other CoRE apps
- 15:20–15:30 Protocol Negotiation
- 15:30–15:50 CoRECONF
- 15:50–15:50 Pulling items forward from Thursday

draft-hartke-core-stateless-02

- (adoption call finished, to be resubmitted as draft-ietf)

draft-bormann-core-corr-clar-00

- Modeled after RFC 4815
- Meant to be a running document for a few years
- Might need to adopt some process for assigning state to the entries

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

Group OSCORE - Secure Group Communication for CoAP

draft-ietf-core-oscore-groupcomm-03

Marco Tiloca, RISE
Göran Selander, Ericsson
Francesca Palombini, Ericsson
Jiye Park, Universität Duisburg-Essen

IETF 103, CoRE WG, Bangkok, November 5th, 2018

Updates from -02   (1/3)

- Major revision:
  - Addressed two detailed reviews from Jim and Peter – Thanks!

- Improved readability
  - Editorial changes and clarifications
  - Better alignment with *draft-ietf-core-object-security-15*

- Key management is left to the ACE documents
  - The Group Manager performs key provisioning and rekeying
  - The Group Manager acts as repo of public keys
  - Details on *draft-tiloca-ace-oscoap-joining-05*

Updates from -02   (2/3)

- Separate sections for …
  - COSE Object
  - OSCORE Header Compression

- Countersignature
  - Now appended to the encrypted payload of the OSCORE message
  - Keep a simple parsing of a (short) OSCORE Option
  - Limit the impact of message fragmentation

- Extended security considerations
  - More on group-level security
  - New on management of group keying material
  - New on misalignment of security contexts after rekeying

Updates from -02 (3/3)

- Discussed wrap-around of sequence numbers (PIVs)
- Shorter single list of Group Manager responsibilities
- IANA registration request for bit #2 of the Flag Byte
  - Presence of the countersignature

- Appendix D – "Set-up of new endpoints"
  - Rewritten, much shorter, and high-level only

Next steps

- Converge to an implementation version
  - Finalize what aspects are left to the application
  - More security considerations, e.g. deltas from OSCORE
  - Is there any significant issue remained to address?

- Implementation
  - RISE will do one in Java for Californium
  - OSRAM Innovation will do one in C, to be used in Dotdot
  - Anyone else interested to implement this draft?

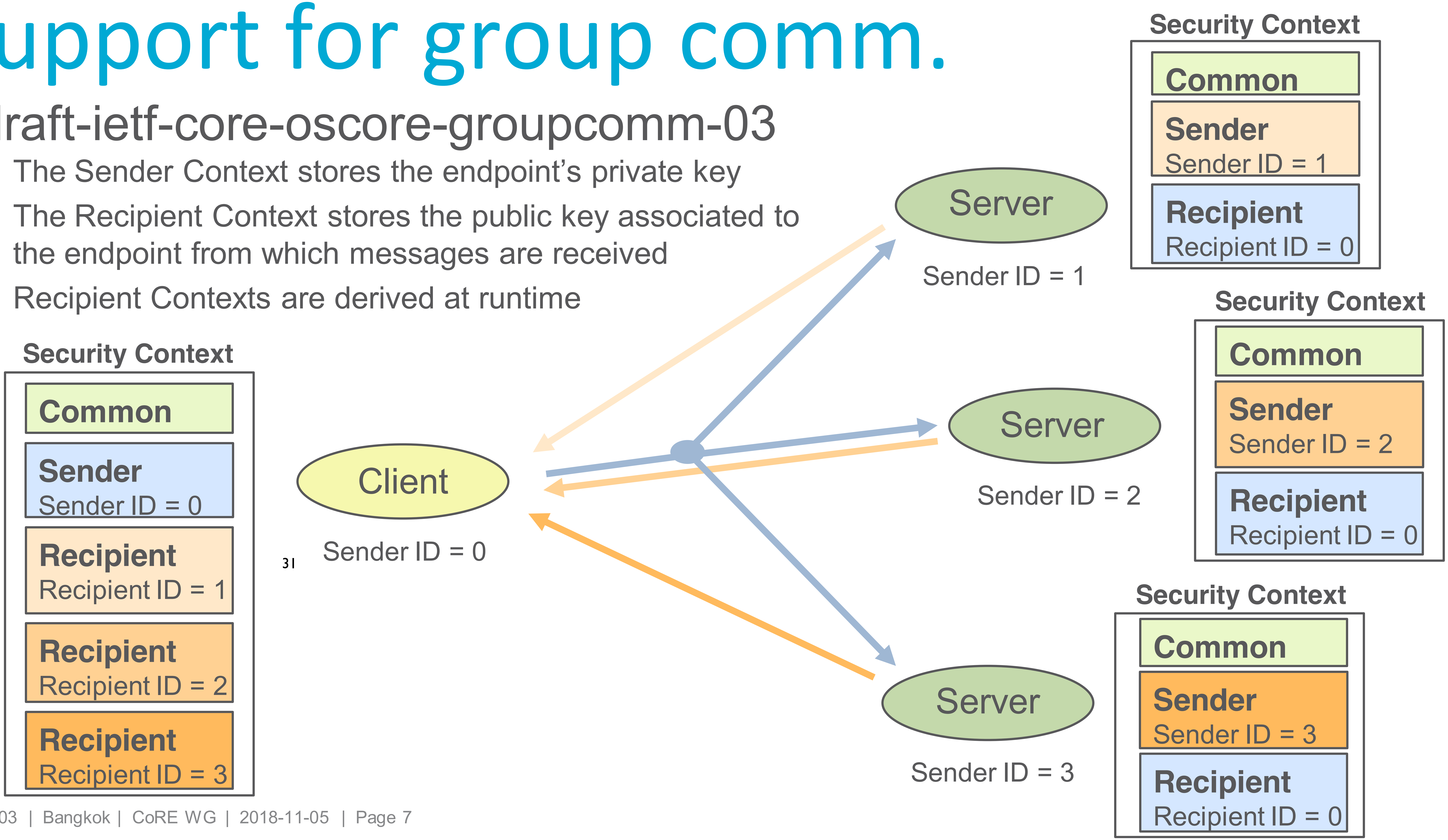
Thank you!

Comments/questions?

https://github.com/core-wg/oscore-groupcomm

Support for group comm.

- draft-ietf-core-oscore-groupcomm-03
  - The Sender Context stores the endpoint’s private key
  - The Recipient Context stores the public key associated to the endpoint from which messages are received
  - Recipient Contexts are derived at runtime



Discovery of OSCORE groups with the CoRE Resource Directory

draft-tiloca-core-oscore-discovery-00

Marco Tiloca, RISE
Christian Amsüss
Peter van der Stok

IETF 103, CoRE WG, Bangkok, November 5th, 2018

Motivation

- From CoRE at IETF 102
  - Does Group OSCORE fit in any way with other works using groups?
  - Use the Resource Directory to facilitate secure group applications

- A newly deployed device
  - Starts with a “Manufacturer Identity”
  - Gets an “Operational Identity” upon deployment

- A device that wants to join an OSCORE group may discover:
  - The Group Identifier of the group (Gid)
  - The multicast IP address(es) used in the group
  - A link to the Group Manager (GM) and its resource to join the group

Motivation

- The group and/or GM are unknown at manufacturing time

- Information on the group changed before device deployment

- The device is deployed
  - Before the GM is deployed
  - Before the OSCORE group is created

Goal

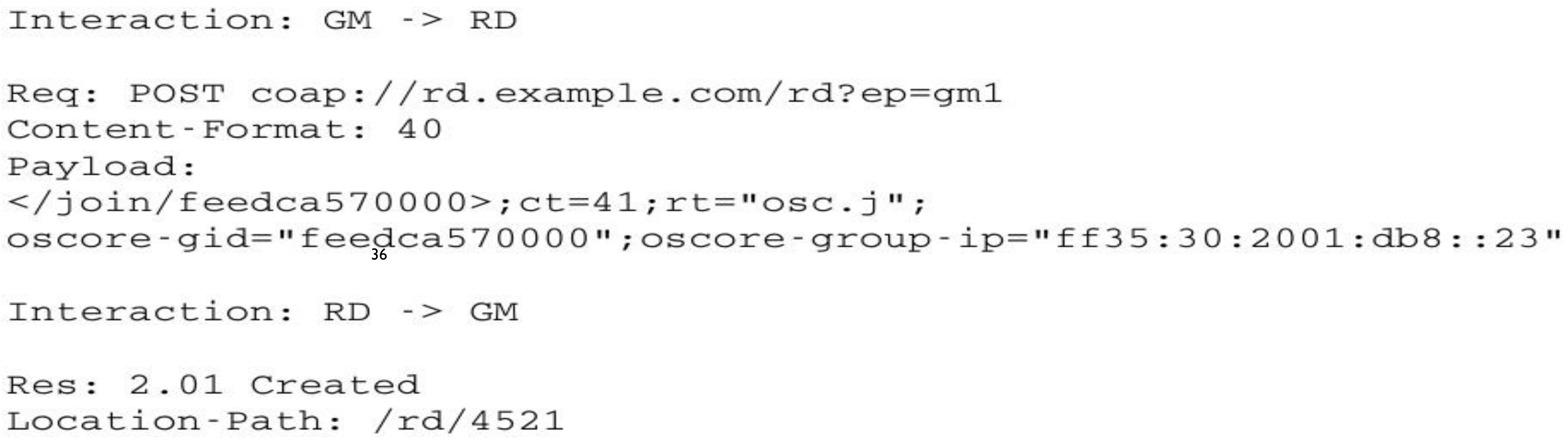
- Use the CoRE Resource Directory (RD) to:
  - Discover an OSCORE group
  - Retrieve information to join the group through its GM

- This uses resource lookup
  - The joining device needs a pointer to the join resource at the GM

- The actual joining process is out of scope
  - Yet, this method is consistent with *draft-tiloca-ace-oscoap-joining-05*

Registration

- The GM registers itself with the RD
  - MUST include all its join resources, with their link attributes
  - New ‘rt’ value “osc.j” in the CoRE Parameters registry

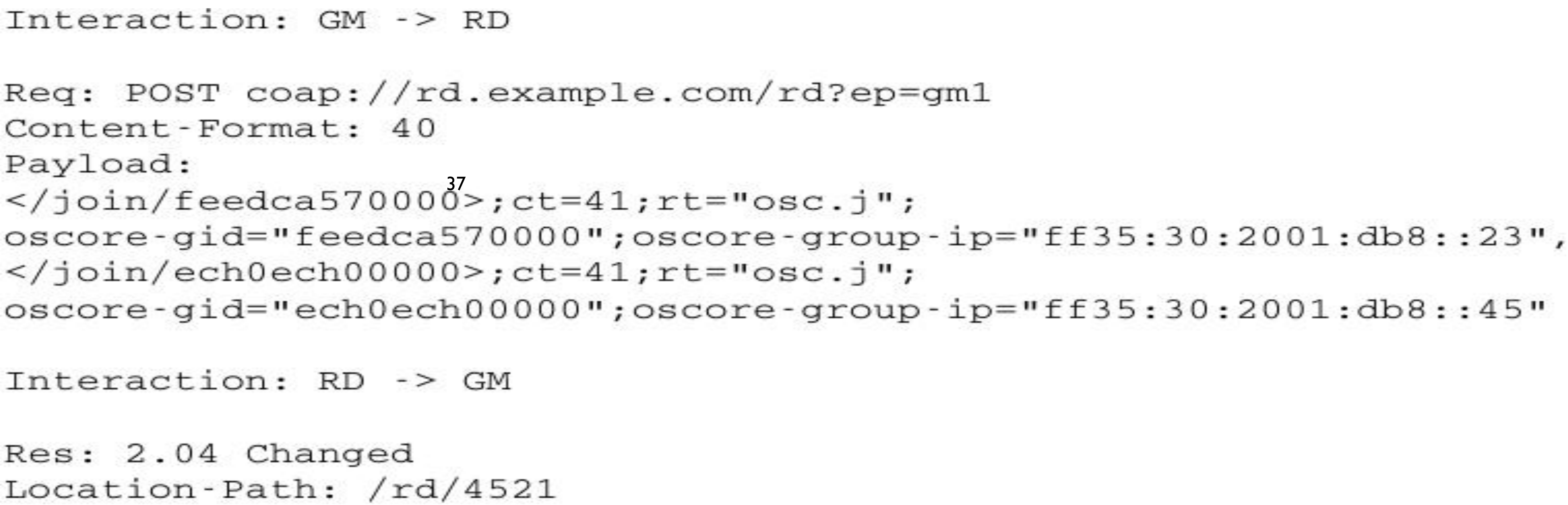


```
Interaction: GM -> RD

Req: POST coap://rd.example.com/rd?ep=gm1
Content-Format: 40
Payload:
</join/feedca570000>;ct=41;rt="osc.j";
oscore-gid="feedca570000";oscore-group-ip="ff35:30:2001:db8::23"

Interaction: RD -> GM

Res: 2.01 Created
Location-Path: /rd/4521
```

Addition/update

- The GM has to
  - Update its own registration within its lifetime

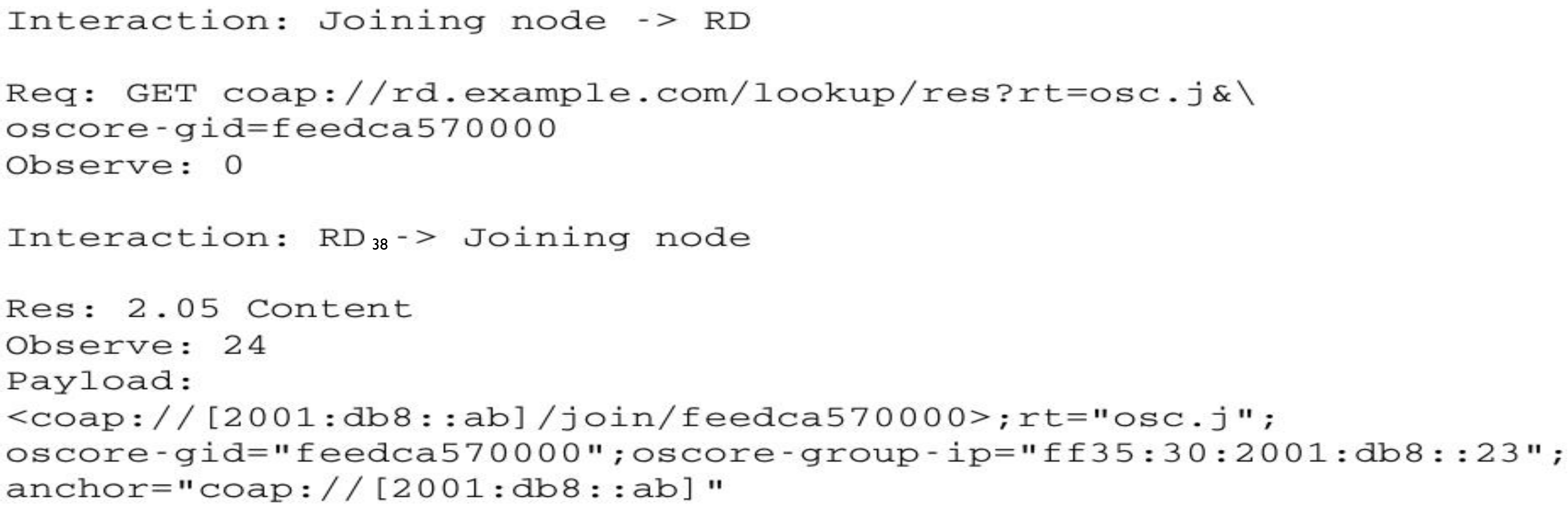
- The GM can add or update OSCORE groups
  - A group with its join resource is created or deleted
  - Information related to the group has changed



```
Interaction: GM -> RD

Req: POST coap://rd.example.com/rd?ep=gm1
Content-Format: 40
Payload:
</join/feedca570000>;ct=41;rt="osc.j";
oscore-gid="feedca570000";oscore-group-ip="ff35:30:2001:db8::23",
</join/ech0ech00000>;ct=41;rt="osc.j";
oscore-gid="ech0ech00000";oscore-group-ip="ff35:30:2001:db8::45"

Interaction: RD -> GM

Res: 2.04 Changed
Location-Path: /rd/4521
```

Discovery

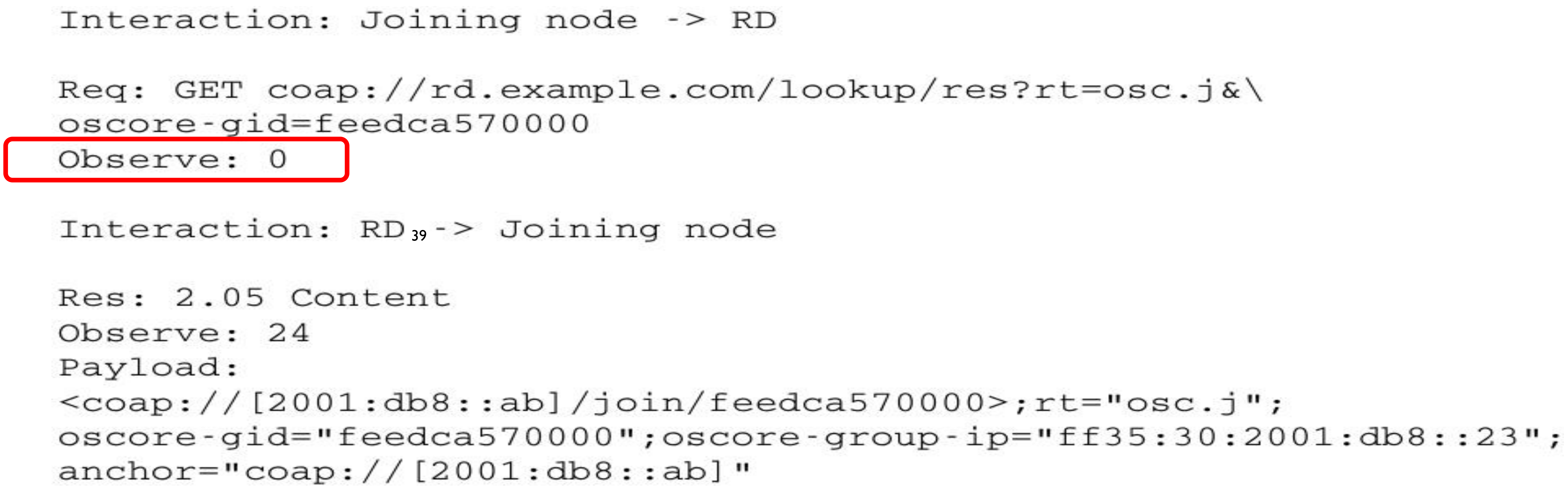
- The device performs a resource lookup at the RD
  - ‘rt’ = “osc.j”   // MUST be present
  - ‘oscore-gid’   // Identifier of the OSCORE group
  - ‘ep’   // Identifier of the GM at the RD



```
Interaction: Joining node -> RD

Req: GET coap://rd.example.com/lookup/res?rt=osc.j&\
oscore-gid=feedca570000
Observe: 0

Interaction: RD -> Joining node

Res: 2.05 Content
Observe: 24
Payload:
<coap://[2001:db8::ab]/join/feedca570000>;rt="osc.j";
oscore-gid="feedca570000";oscore-group-ip="ff35:30:2001:db8::23";
anchor="coap://[2001:db8::ab]"
```

Discovery

› Use of observation
  - Automatic notification if group information changes
  - Useful if this lookup occurs before the group is created
  - Recommended only if ‘oscore-gid’ is used (possible large responses)



```
Interaction: Joining node -> RD

Req: GET coap://rd.example.com/lookup/res?rt=osc.j&\
oscore-gid=feedca570000
Observe: 0

Interaction: RD -> Joining node

Res: 2.05 Content
Observe: 24
Payload:
<coap://[2001:db8::ab]/join/feedca570000>;rt="osc.j";
oscore-gid="feedca570000";oscore-group-ip="ff35:30:2001:db8::23";
anchor="coap://[2001:db8::ab]"
```

Next steps

- Get feedback/comments

- Align the document with possible updates to the RD

Thank you!

Comments/questions?

https://gitlab.com/crimson84/draft-tiloca-core-oscore-discovery

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

Resource Directory

draft-ietf-core-resource-directory

Zach Shelby, Michael Koster, Carsten Bormann, Peter van der Stok,
Christian Amsüss

2018-11-05

Status

From review and dependent document:

- Appendix “Modernized Link Format” is overstepping
- Groups are not used as described

Et cetera

- Security policies updated
- Plug test successful, only details remain, Groups not tested
- Editorial changes

all in -16

~~Modernized Link Format~~

- redefined interpretation of RFC6690 links
- Background: Not implemented as specified (0/10)
- Instead: defined unambiguous subset
- Downside: Some use cases need to wait for CoRAL or similar, or depend on implementation specifics

see core-links-json

Group proposal

~~Groups: separate concept, enumerating membership~~

Groups: almost an endpoint (with endpoint type `et=core.gp`).
No members registered, but resources.

Group proposal

~~Groups: separate concept, enumerating membership~~

Groups: almost an endpoint (with endpoint type `et=core.gp`).
No members registered, but resources.

```
GET /rd-lookup/res?ep=my-group
<coap://[ff05::8431]/light>;rt="light";...
```

Group proposal

~~Groups: separate concept, enumerating membership~~

Groups: almost an endpoint (with endpoint type `et=core.gp`).
No members registered, but resources.

```
GET /rd-lookup/res?ep=my-group
<coap://[ff05::8431]/light>;rt="light";...
```

-17: Draft size -10%, compatible with implementations,
Groups described an usage pattern

Next steps for `resource-directory`

Does anyone use more than the limited subset of RFC6690?

Is anyone using pre -17 groups?

[50](no, no): publish version for WGLC

RD-DNS-SD

Peter van der Stok, Kerry Lynn, Michael Koster, Christian Amsuess

IETF 103 - CORE Working Group

-03 Updates to -02

Motivation for mapping between
resource discovery and service discovery

DNS Domain:
follow sctl-service-registration draft to determine domain

Service Type:
Analogy between resource type and service type functionality

Instance:

- manufacturer generated name
- UUID
- if- attribute
- During deployment by Commissioning Tool

Suggestion

IANA registry to map Service Type to resource type

TODO

- More restrictions on character string?
- Sollicit comments

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- 13:50–14:00 Intro, Agenda, Status
- 14:00–14:05 Post-WGLC: Multipart-CT (CB)
- 14:05–14:25 Recently adopted, in adoption
- 14:25–14:40 OSCORE, continued (MT)
- 14:40–15:00 Resource-Directory, Link-Format (CA)
- 15:00–15:20 Other CoRE apps
- 15:20–15:30 Protocol Negotiation
- 15:30–15:50 CoRECONF
- 15:50–15:50 Pulling items forward from Thursday

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

SID proposal

Peter van der Stok

IETF 103 - CoRE Working Group

SID reminder

The contents of YANG specifications are transported over constrained networks.

CBOR is used to serialize the contents
The YANG names can be very long and are reduced to numeric identifiers called: SID.

For example: ANIMA WG specifies the Voucher in YANG.

SID: YANG Schema Item iDentifier

SID registration

Once SIDs are allocated and described in an RFC,
they MUST NOT change.
SID ranges are allocated to modues from the comi.space facility,
and may be subject to change during I-D development.

A RFC range exists to be fragmented over ranges allocated to RFCs

Once the draft is accepted as RFC, the following actions should be taken:
- The SID range for every module in the RFC is allocated from the RFC range.
- The contents of the SID files (one per module) are included in the RFC.
- IANA registers the module names, RFC number, and the SID range.
- IANA registers the YANG name to SID map for every module in the RFC.

SID: YANG Schema Item iDentifier

IANA involvement

Ask IANA to provide an extension to YANG parameter registry:
https://www.iana.org/assignments/yang-parameters/yang-parameters.xhtml

It contains a YANG module sub-registry.

RFC6020, section 14.1

Suggestion to create a additional SID module sub-registry.

SID: YANG Schema Item iDentifier

Question

Support to insert equivalent text in core-sid draft?

62

How to finish this?

- Proposal: Add an editor to the documents
- Finish the last lap

- Volunteer: Ivaylo Petrov

All times are in time-warped ICT (UTC+07:00)

Monday (120 min)

- **13:50–14:00 Intro, Agenda, Status**
- **14:00–14:05 Post-WGLC: Multipart-CT (CB)**
- **14:05–14:25 Recently adopted, in adoption**
- **14:25–14:40 OSCORE, continued (MT)**
- **14:40–15:00 Resource-Directory, Link-Format (CA)**
- **15:00–15:20 Other CoRE apps**
- **15:20–15:30 Protocol Negotiation**
- **15:30–15:50 CoRECONF**
- **15:50–15:50 Pulling items forward from Thursday**

Draft-ietf-core-dev-urn-03

Arkko, Jennings & Shelby

A Uniform Resource Name (URN) namespace for hardware device identifiers.

Potentially useful in applications such as in sensor data streams and storage, or equipment inventories.

Complements other similar identifiers NIs (RFC 6920), UUIDs (RFC 4122), IMEIs (RFC 7254) etc. Supports, e.g., MAC and EUI-64, identifiers as well as various organisation-specific free formats.

urn:dev:mac:0024befffe804ff1

Version -03

- No major changes
- Some reference updates
- Went back disallowing %-encoding
  - DEV URNs are likely to appear in SenML sensor name fields
  - RFC 8428 prohibits names to include %:

    name MUST consist only of characters out of the set “A" to "Z", "a" to "z", and "0" to "9", as well as "-", ":", ".", “/", and "_"

Moving Forward

- This draft formally defines some parts of LwM2M OMA specifications that specified the os and ops syntaxes
    - I think it makes for the IETF to do that; we should define the generic formats that have a need in the industry, including making changes when necessary
    - Shout now if that’s a problem for any deployment!
- There are some remaining URN issues in LwM2M
    - 1) Need nai, extid, imei-imsi, imei-meid; 2) esn identifiers seem outdated; 3) meid and imei URNs seem to be used incorrectly
    - I think these are beyond the scope of the DEV URN spec and should be dealt with separately and maybe by someone else
- Last call?

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:52 Active drafts**
- **11:52–12:06 FASOR**
- **12:06–12:20 Streaming (Monday)**
- **12:20–12:20 Other new work**

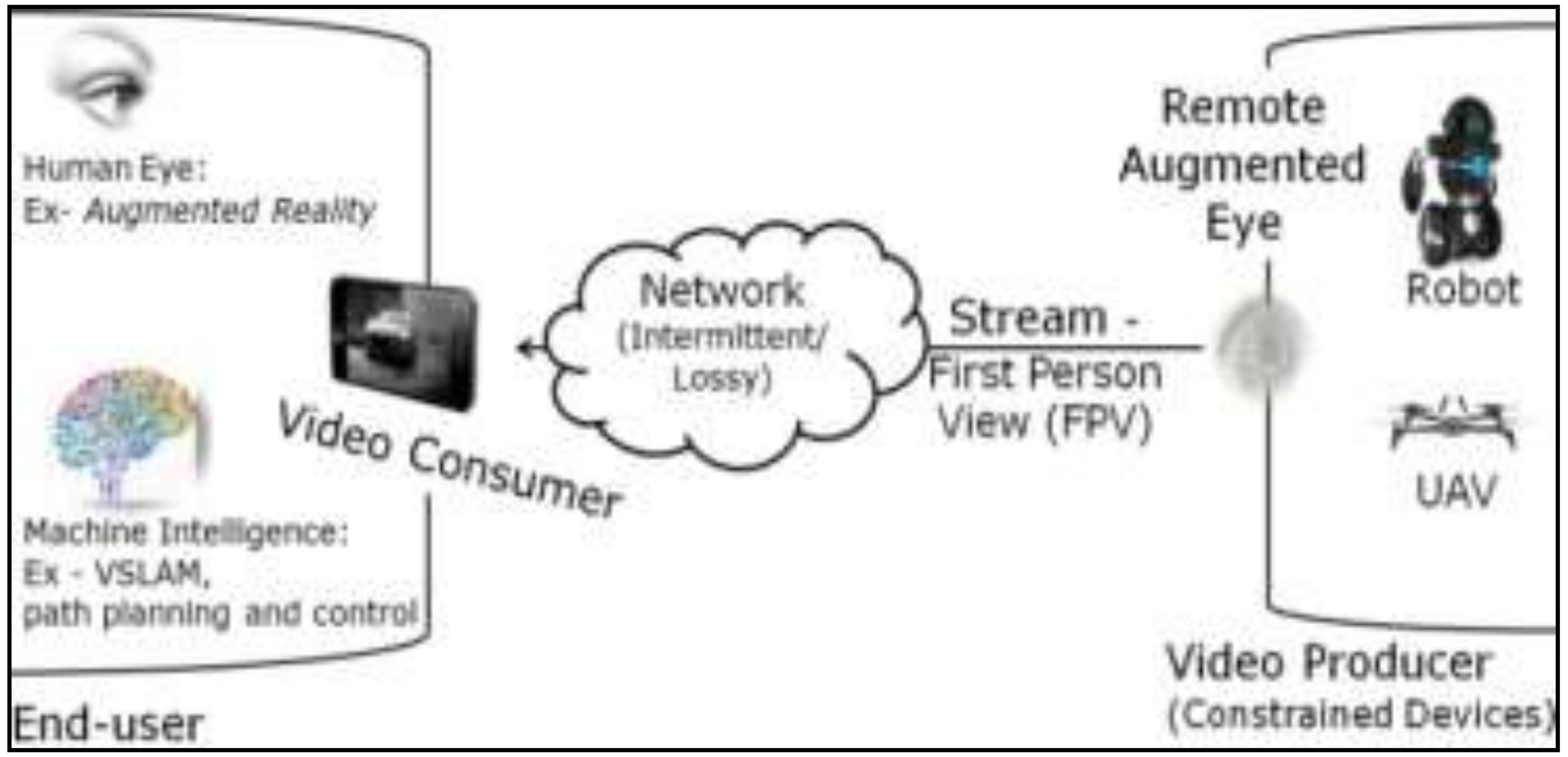
Adaptive RESTful Real-time Live Streaming for Things (A-REaLiST)

(draft-bhattacharyya-core-a-realist-00)

Abhijan Bhattacharyya

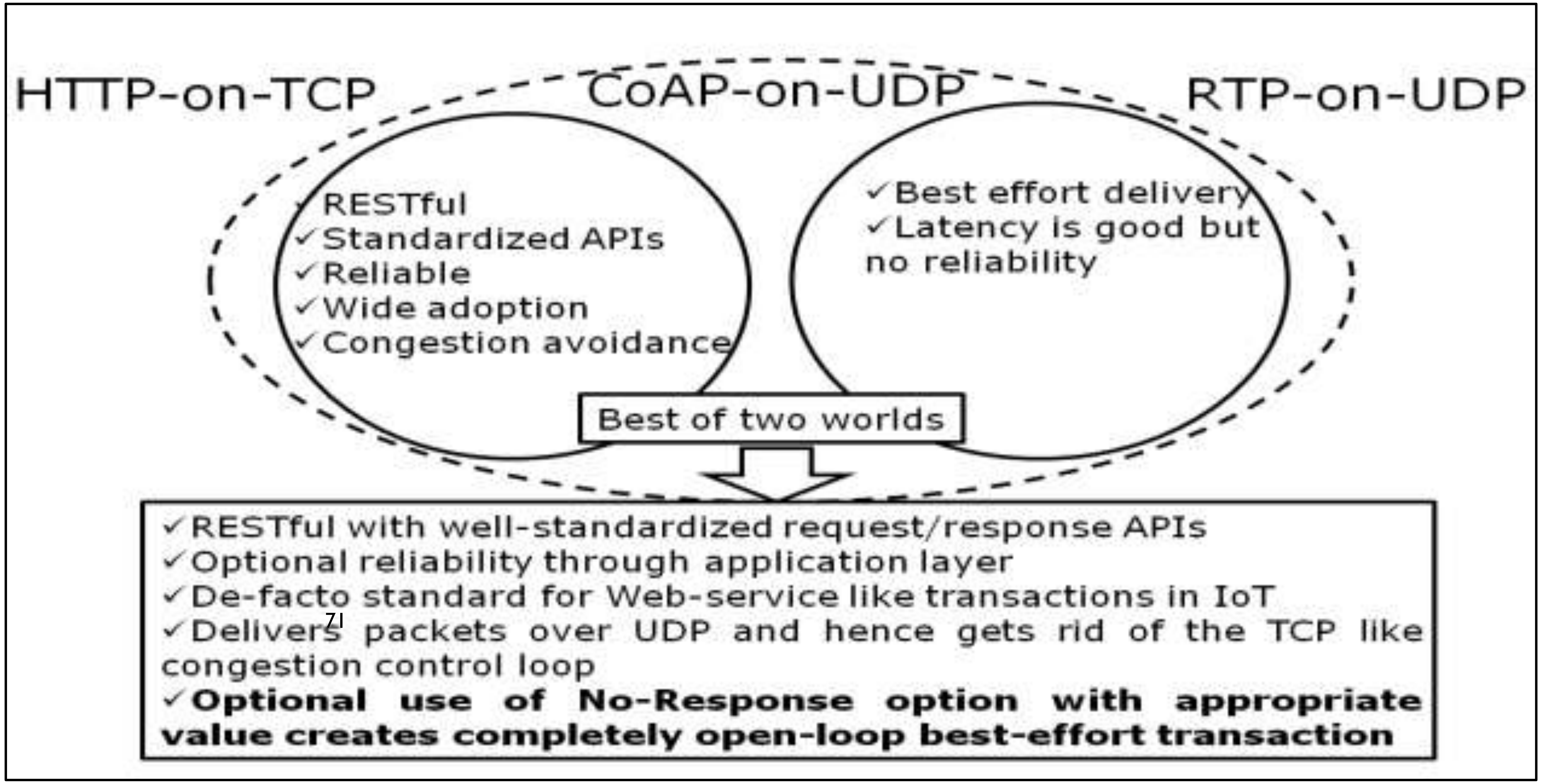
Motivation

- Steaming time-series sensor data gaining importance
  - Visual sensing is unobtrusive
- Immediate trigger: applications requiring real-time actuation decision based on live FPV feedback.
  - AR (Augmented Reality) applications, VSLAM (Visual Simultaneous Localization and Mapping) for maneuvering remote dumb robot terminals
  - Indoor application: Factory or warehouses are typical indoor application
  - Outdoor application: Remote infrastructure monitoring using drones, etc.
- Solution needs to maintain high QoE despite *intermittent* connectivity and fluctuating signal strength
  - Low-latency ● High visual quality ● Low computing ● Energy efficient ● Highly real-time ● No video freezing
- There are problems even in indoor
  - Example: Warehouse/ factory wireless environment has typical problems
    - Sporadic zones without radio coverage
    - Variability in radio environment
      - Change of products, addition / alteration of racks in racks changes the radio attenuation / interference/ shadowing characteristics
      - Addition of access points may create new zones of bad interference
- Experience with existing techniques is not good.



CoAP : rediscovering

Though originally conceived for small sensor updates, but let’s look at CoAP this way:



- ✓RESTful
- ✓Standardized APIs
- ✓Reliable
- ✓Wide adoption
- ✓Congestion avoidance

- ✓Best effort delivery
- ✓Latency is good but no reliability

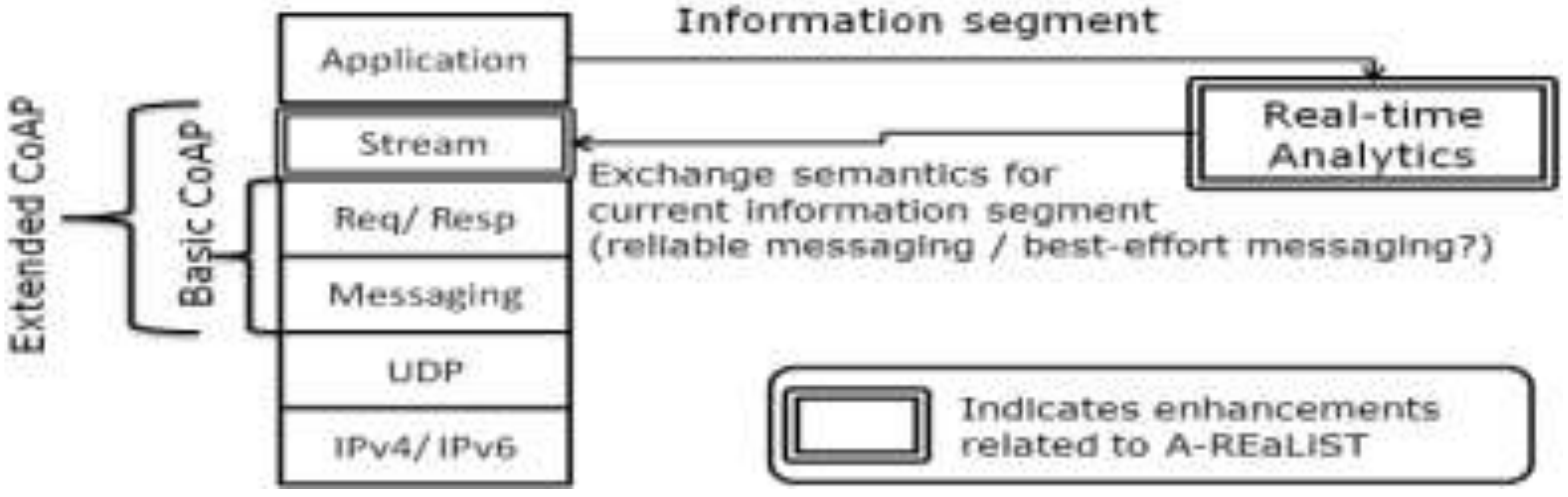
- ✓RESTful with well-standardized request/response APIs
- ✓Optional reliability through application layer
- ✓De-facto standard for Web-service like transactions in IoT
- ✓Delivers packets over UDP and hence gets rid of the TCP like congestion control loop
- ✓**Optional use of No-Response option with appropriate value creates completely open-loop best-effort transaction**

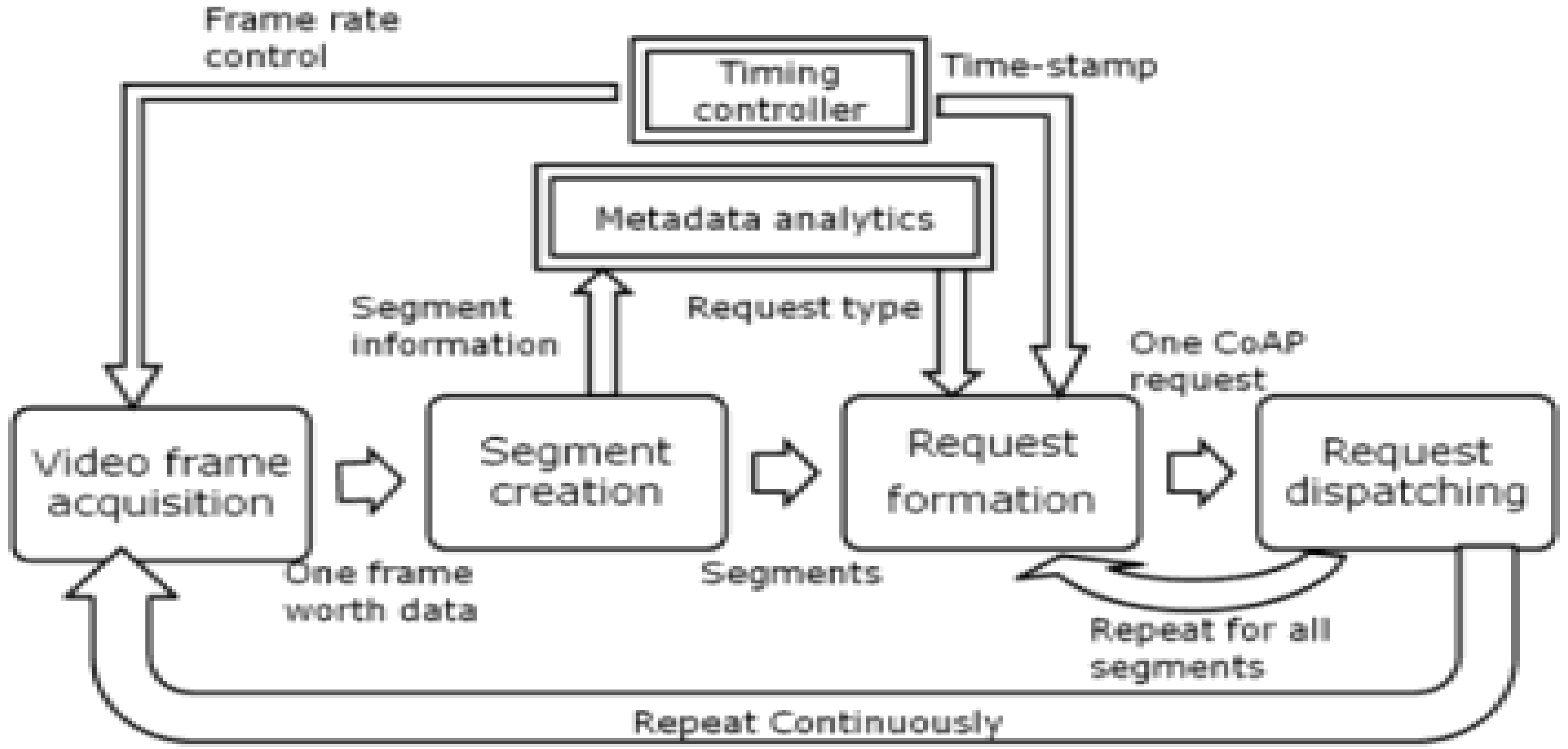
An Expectation

- Can we have a RESTful protocol which is equally equipped to exchange small sensor data as well as stream in real-time with high QoE?
  - Example: Deploy on remote terminals (UAV, etc.) – collect telemetry and other sensor info, as well as get live FPV and send control commands – all through same stack.
- Just like HTTP provides access to normal RESTful web-services as well as streams through a singular infrastructure – can we have a parallel for the IoT world?

A-REaLiST : Core idea

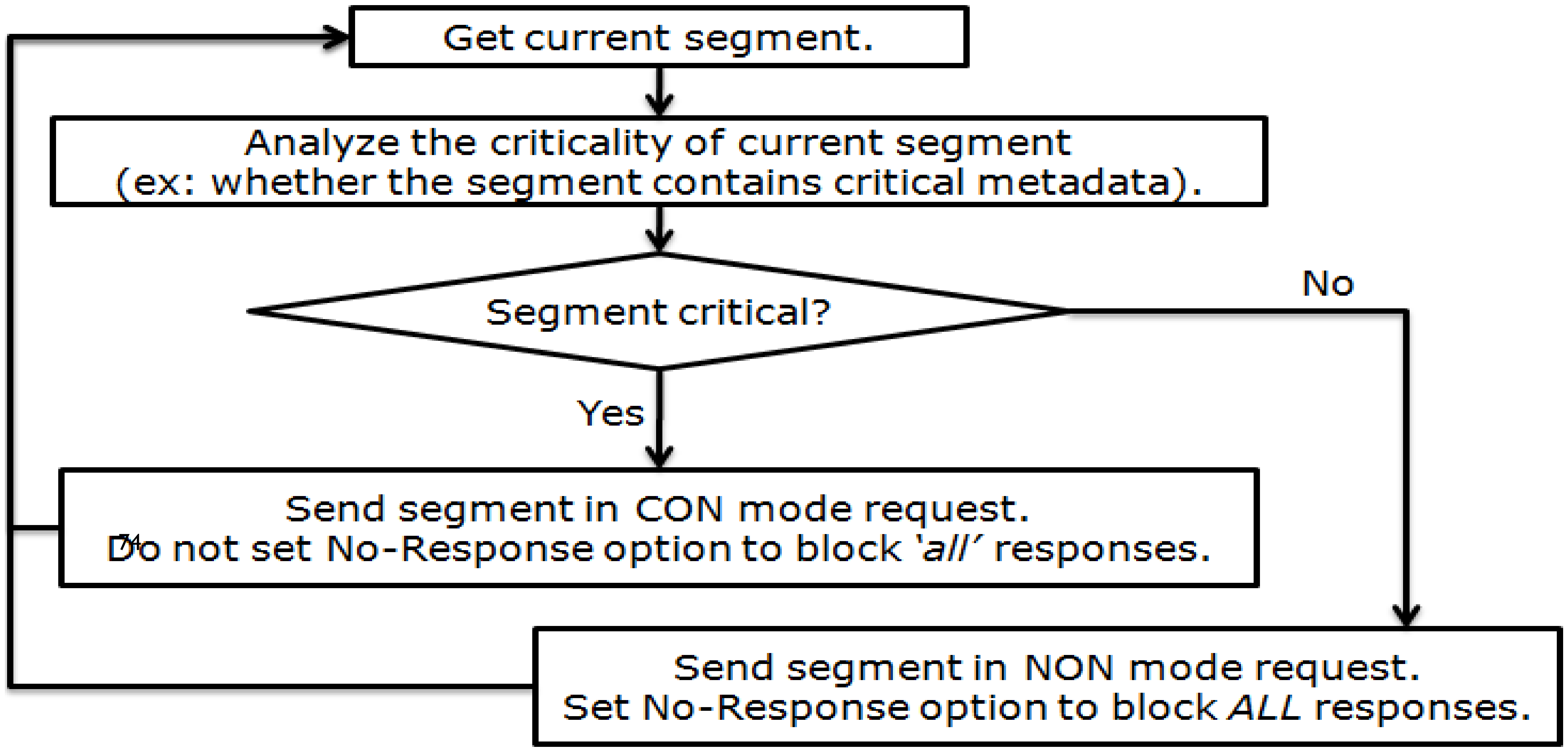
- Content is delivered following the progressive download principles
  - Deliver information segments as CoAP messages
- Strike a balance between reliability and real-time delivery
- Switch the between reliable and best-effort semantics based on the inferred criticality of the information content in a CoAP message
  - Critical information as reliable and non-critical as best-effort
  - Criticality relates to the fact – how important is the information for reconstruction
  - Switching does not have any additional control overhead for CoAP – just a matter of manipulating the header fields intelligently
- An intelligent rendering engine estimate the whole frame despite losing some non-critical information
  - A-REaLiST provides the necessary hooks


Information segment
Application
Stream
Req/ Resp
Messaging
UDP
IPv4/ IPv6
Extended CoAP
Basic CoAP
Real-time Analytics
Exchange semantics for current information segment (reliable messaging / best-effort messaging?)
Indicates enhancements related to A-REaLiST



A-REaLiST : Implicit Congestion Avoidance

- If a critical segment of a frame could not be delivered then drop rest of the segments of that frame
- Rendering engine is anyway going to fail by missing the critical segment - why clog the network?



Get current segment.

Analyze the criticality of current segment
(ex: whether the segment contains critical metadata).

Segment critical?

No

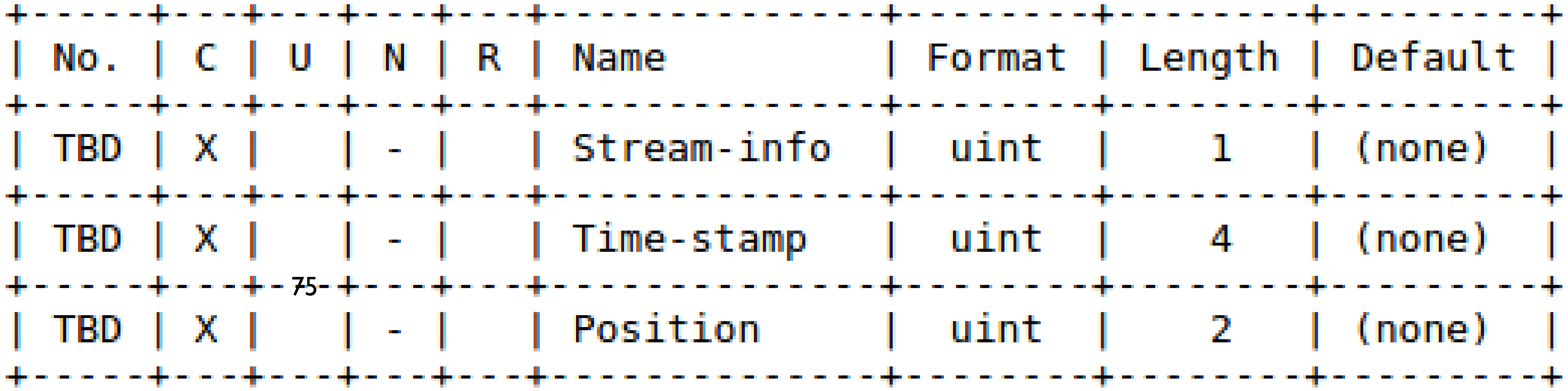
Yes

Send segment in CON mode request.
Do not set No-Response option to block *'all'* responses.

Send segment in NON mode request.
Set No-Response option to block *ALL* responses.

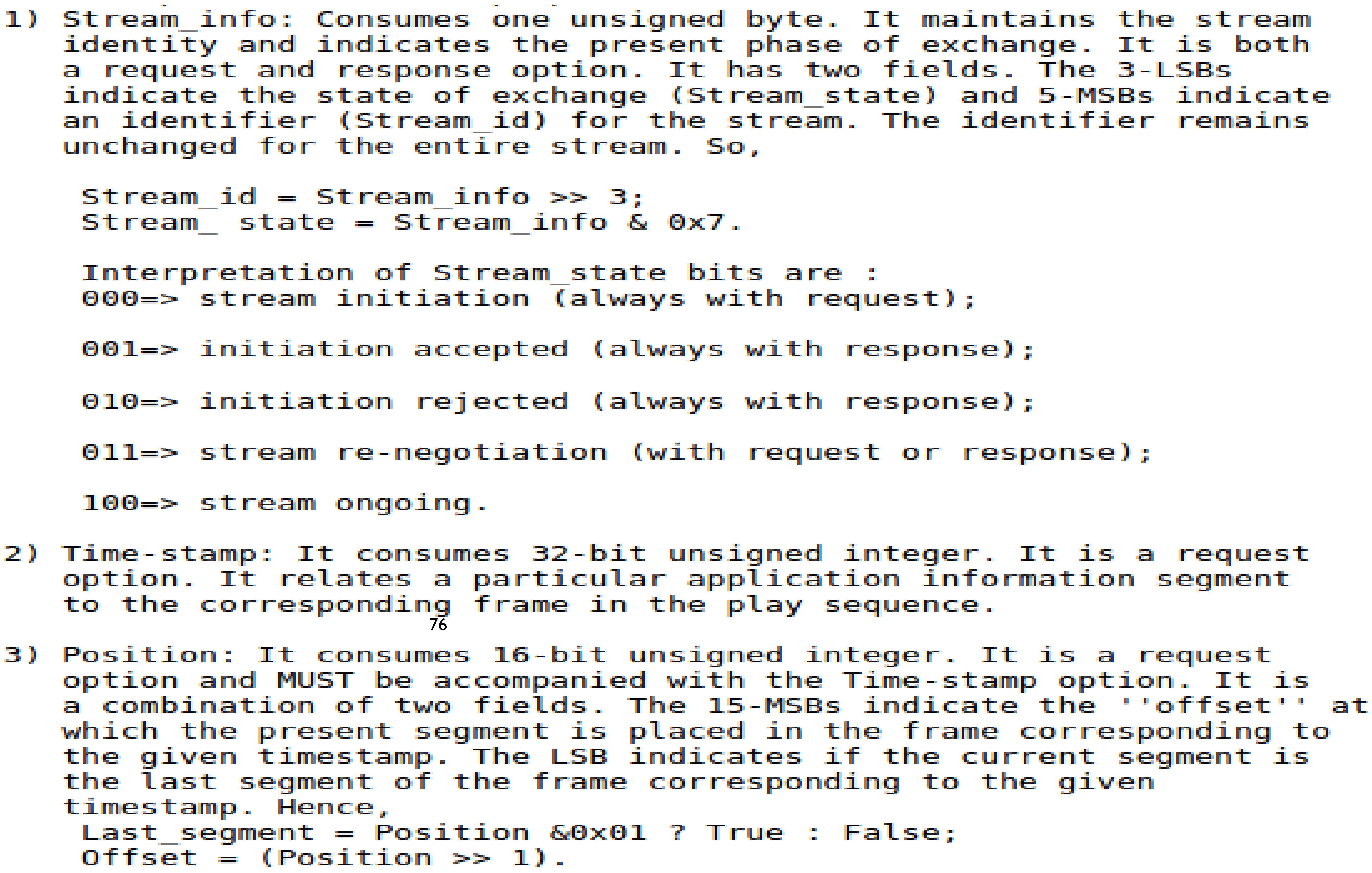
A-REaLiST : Header Extensions

- We need to maintain some mechanism for controlling the negotiation of the stream to allow end-applications to handle the stream-states in a resource efficient manner
- We need to provide some hooks so that end-application can relate the segments
  - 2 levels
    1) Segment maps to which fundamental unit (frame/ GoP)?
    2) Where to position the segment within the unit?



No.	C	U	N	R	Name	Format	Length	Default
TBD	X		-		Stream-info	uint	1	(none)
TBD	X		-		Time-stamp	uint	4	(none)
TBD	X		-		Position	uint	2	(none)

A-REaLiST : Header Extensions (Contd.)



1) Stream_info: Consumes one unsigned byte. It maintains the stream identity and indicates the present phase of exchange. It is both a request and response option. It has two fields. The 3-LSBs indicate the state of exchange (Stream_state) and 5-MSBs indicate an identifier (Stream_id) for the stream. The identifier remains unchanged for the entire stream. So,

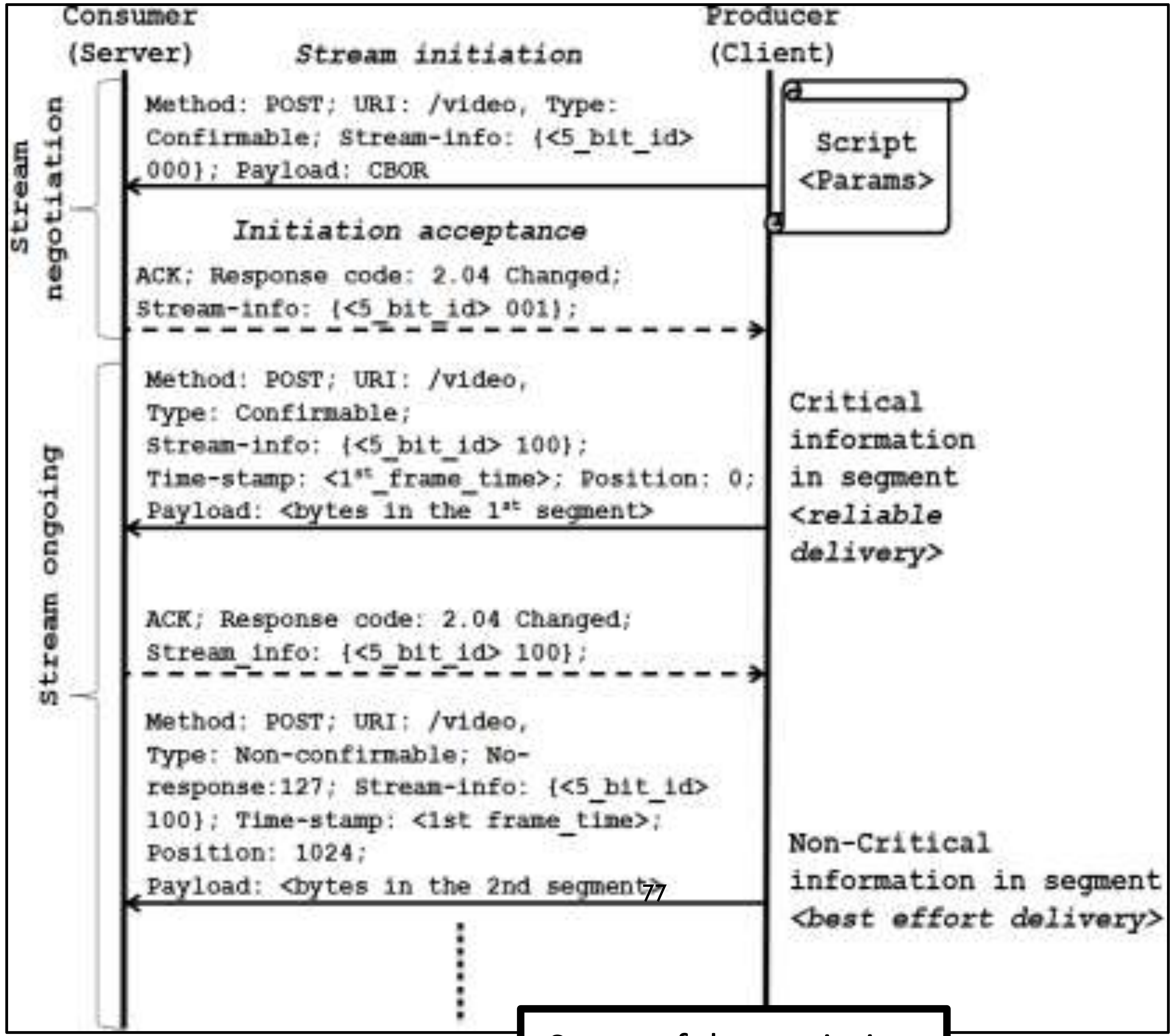
```
 Stream_id = Stream_info >> 3;
 Stream_ state = Stream_info & 0x7.

 Interpretation of Stream_state bits are :
 000=> stream initiation (always with request);

 001=> initiation accepted (always with response);

 010=> initiation rejected (always with response);

 011=> stream re-negotiation (with request or response);

 100=> stream ongoing.
```

2) Time-stamp: It consumes 32-bit unsigned integer. It is a request option. It relates a particular application information segment to the corresponding frame in the play sequence.

3) Position: It consumes 16-bit unsigned integer. It is a request option and MUST be accompanied with the Time-stamp option. It is a combination of two fields. The 15-MSBs indicate the ''offset'' at which the present segment is placed in the frame corresponding to the given timestamp. The LSB indicates if the current segment is the last segment of the frame corresponding to the given timestamp. Hence,

```
 Last_segment = Position &0x01 ? True : False;
 Offset = (Position >> 1).
```

Example Handshakes



Stream initiation

Method: POST; URI: /video, Type: Confirmable; Stream-info: {<5_bit_id> 000}; Payload: CBOR

Initiation acceptance

ACK; Response code: 2.04 Changed; Stream-info: {<5_bit_id> 001};

Method: POST; URI: /video, Type: Confirmable; Stream-info: {<5_bit_id> 100}; Time-stamp: <1st_frame_time>; Position: 0; Payload: <bytes in the 1st segment>

Critical information in segment *<reliable delivery>*

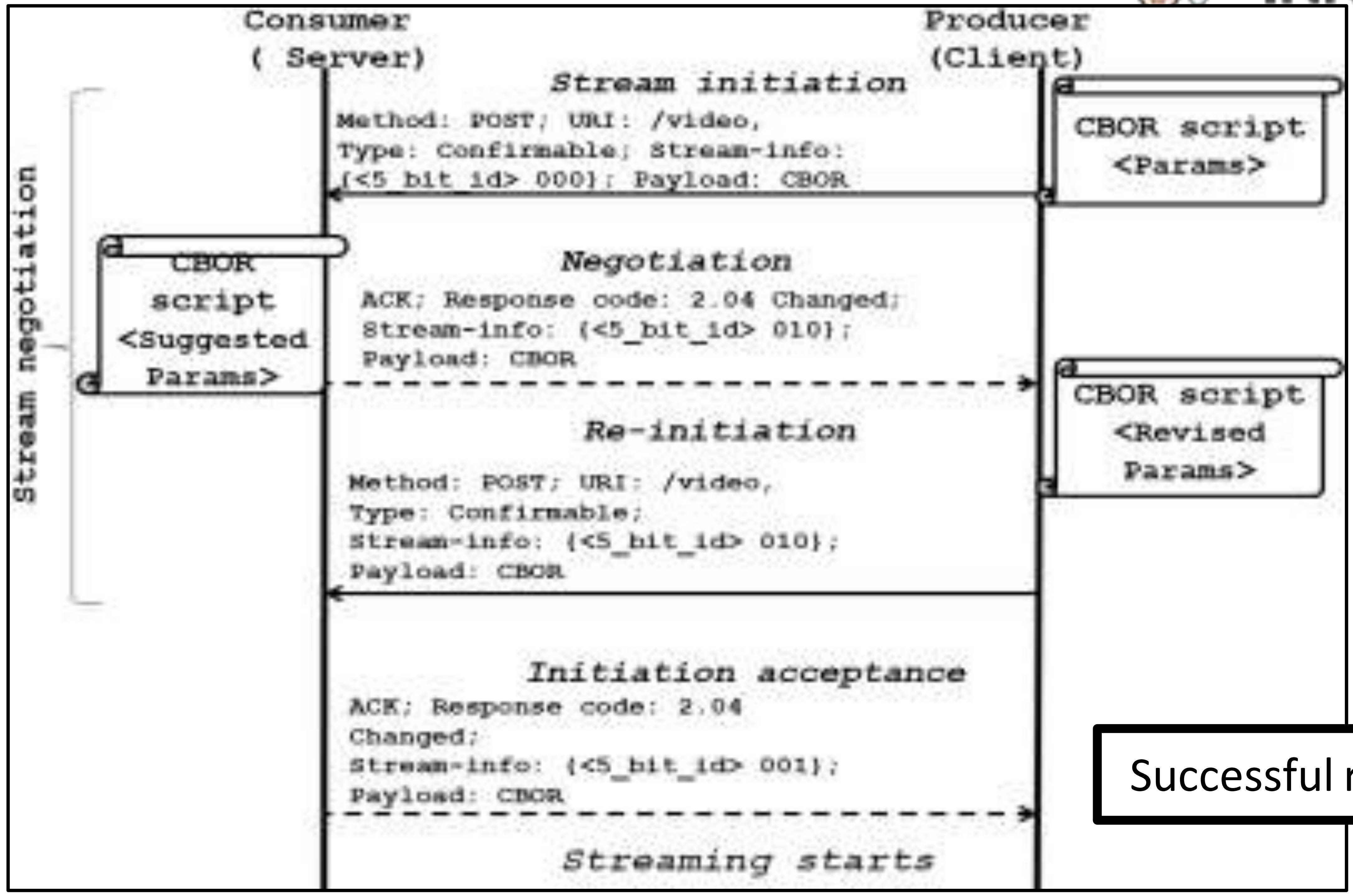
ACK; Response code: 2.04 Changed; Stream_info: {<5_bit_id> 100};

Method: POST; URI: /video, Type: Non-confirmable; No-response:127; Stream-info: {<5_bit_id> 100}; Time-stamp: <1st frame_time>; Position: 1024; Payload: <bytes in the 2nd segment>

Non-Critical information in segment *<best effort delivery>*

Successful negotiation

Note: Initiation is from the producer side.



Stream initiation

Method: POST; URI: /video, Type: Confirmable; Stream-info: {<5_bit_id> 000}; Payload: CBOR

Negotiation

ACK; Response code: 2.04 Changed; Stream-info: {<5_bit_id> 010}; Payload: CBOR

Re-initiation

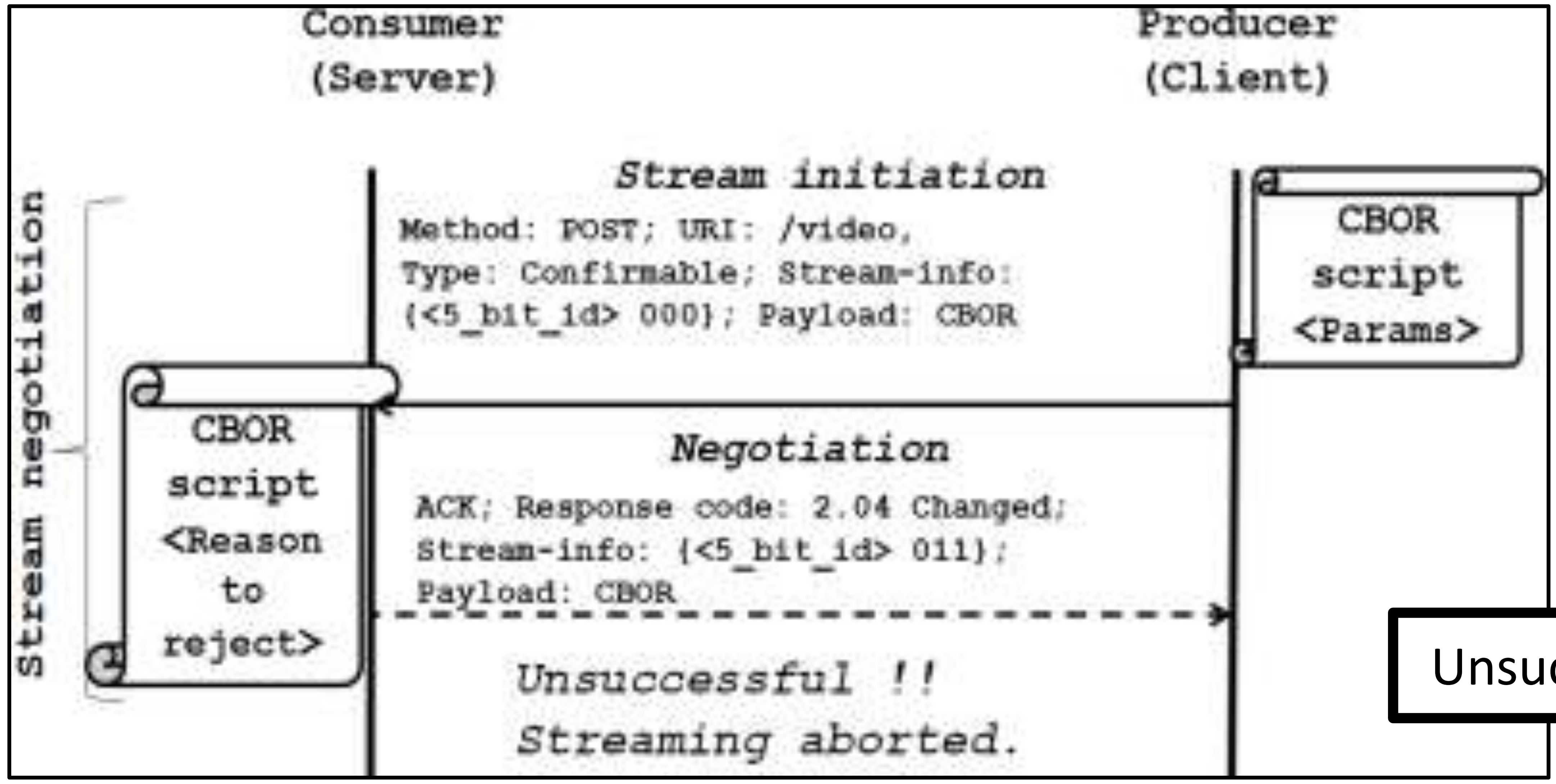
Method: POST; URI: /video, Type: Confirmable; Stream-info: {<5_bit_id> 010}; Payload: CBOR

Initiation acceptance

ACK; Response code: 2.04 Changed; Stream-info: {<5_bit_id> 001}; Payload: CBOR

Streaming starts

Successful re-negotiation



Stream initiation

Method: POST; URI: /video, Type: Confirmable; Stream-info: {<5_bit_id> 000}; Payload: CBOR

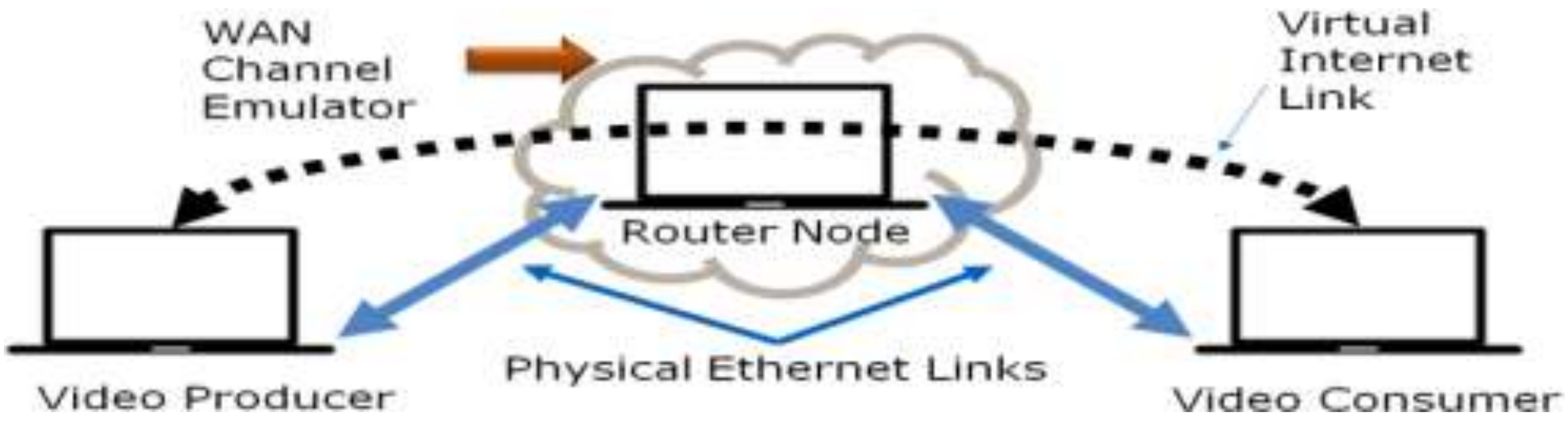
Negotiation

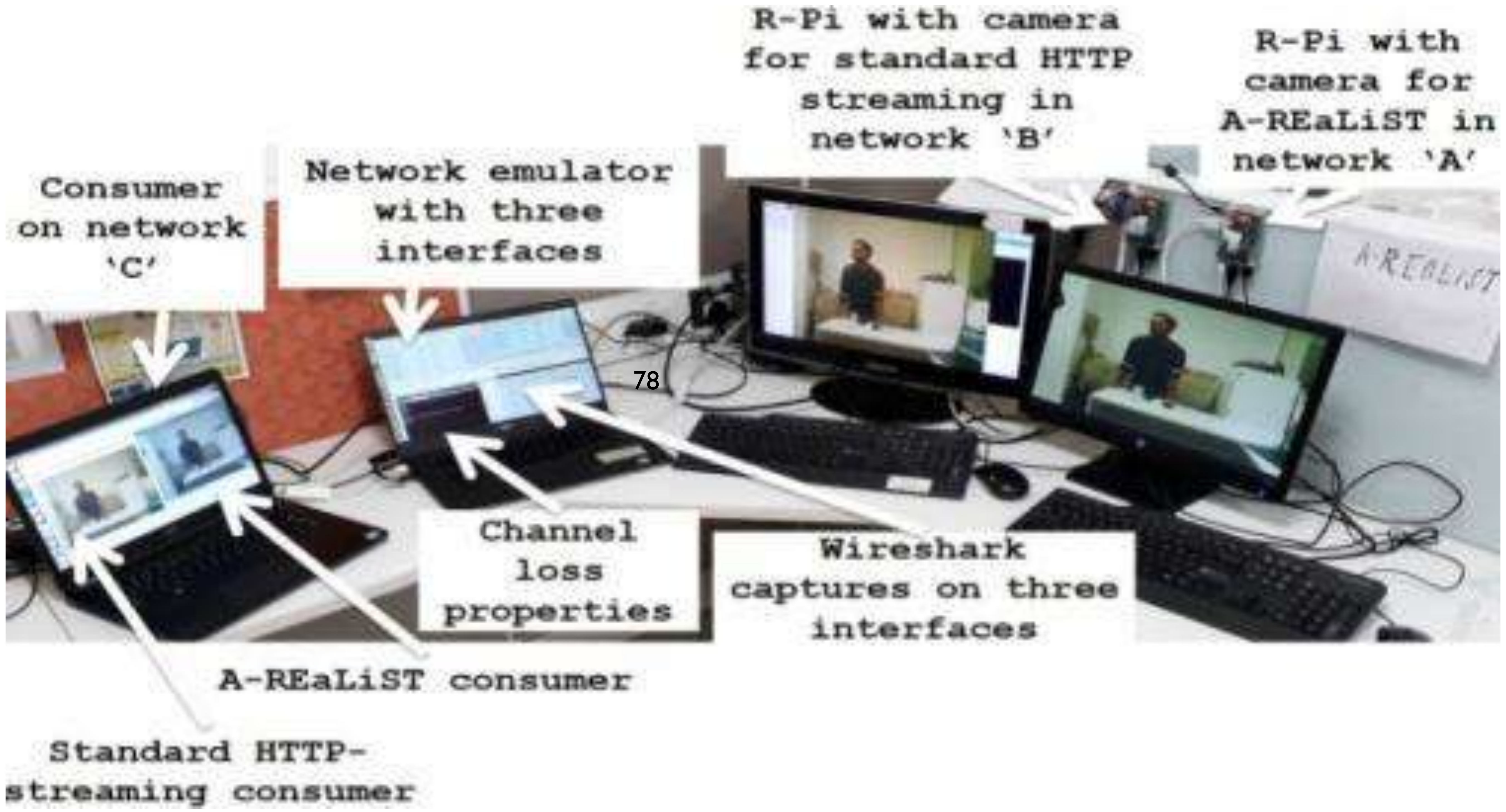
ACK; Response code: 2.04 Changed; Stream-info: {<5_bit_id> 011}; Payload: CBOR

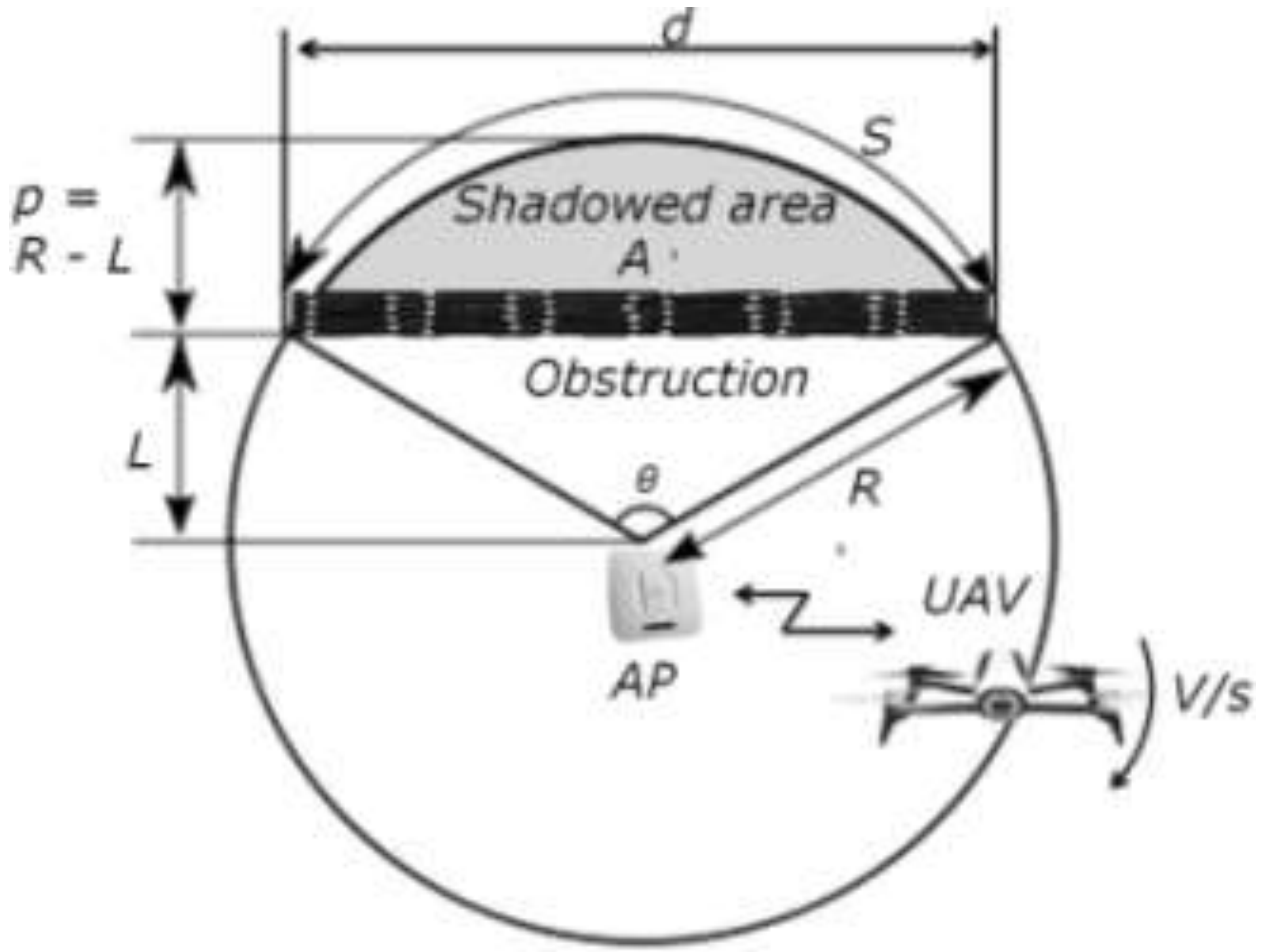
Unsuccessful !! Streaming aborted.

Unsuccessful

Evaluation (emulation)

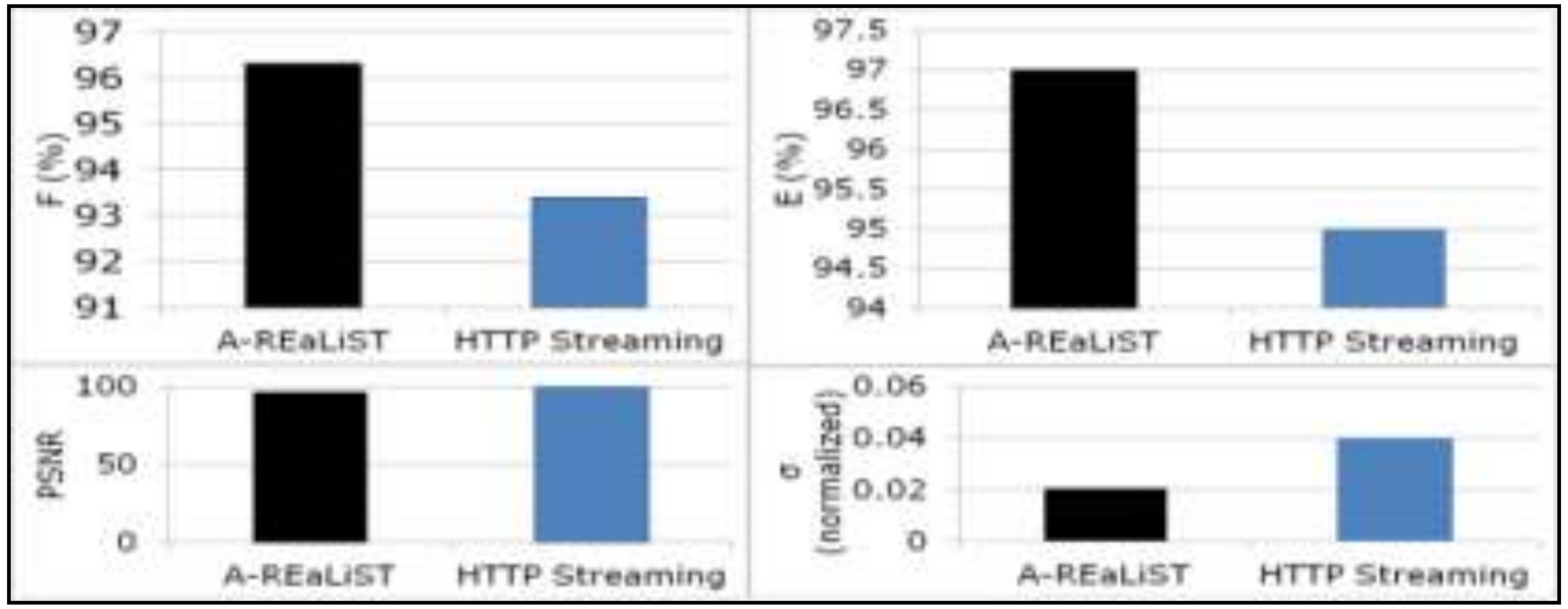


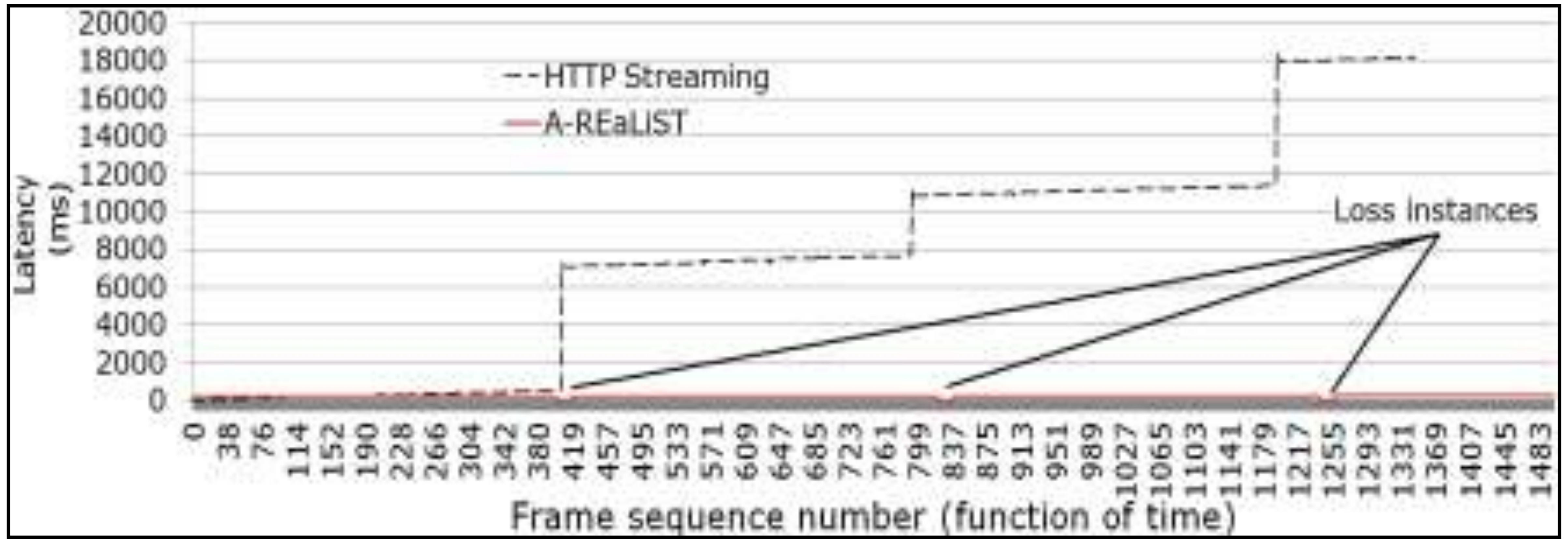

R-Pi with camera for standard HTTP streaming in network 'B'
R-Pi with camera for A-REaLiST in network 'A'
Consumer on network 'C'
Network emulator with three interfaces
A-REALIST
Channel loss properties
Wireshark captures on three interfaces
A-REaLiST consumer
Standard HTTP-streaming consumer



Realistic loss model

Evaluation – comparing with off-the-shelve HTTP-streaming



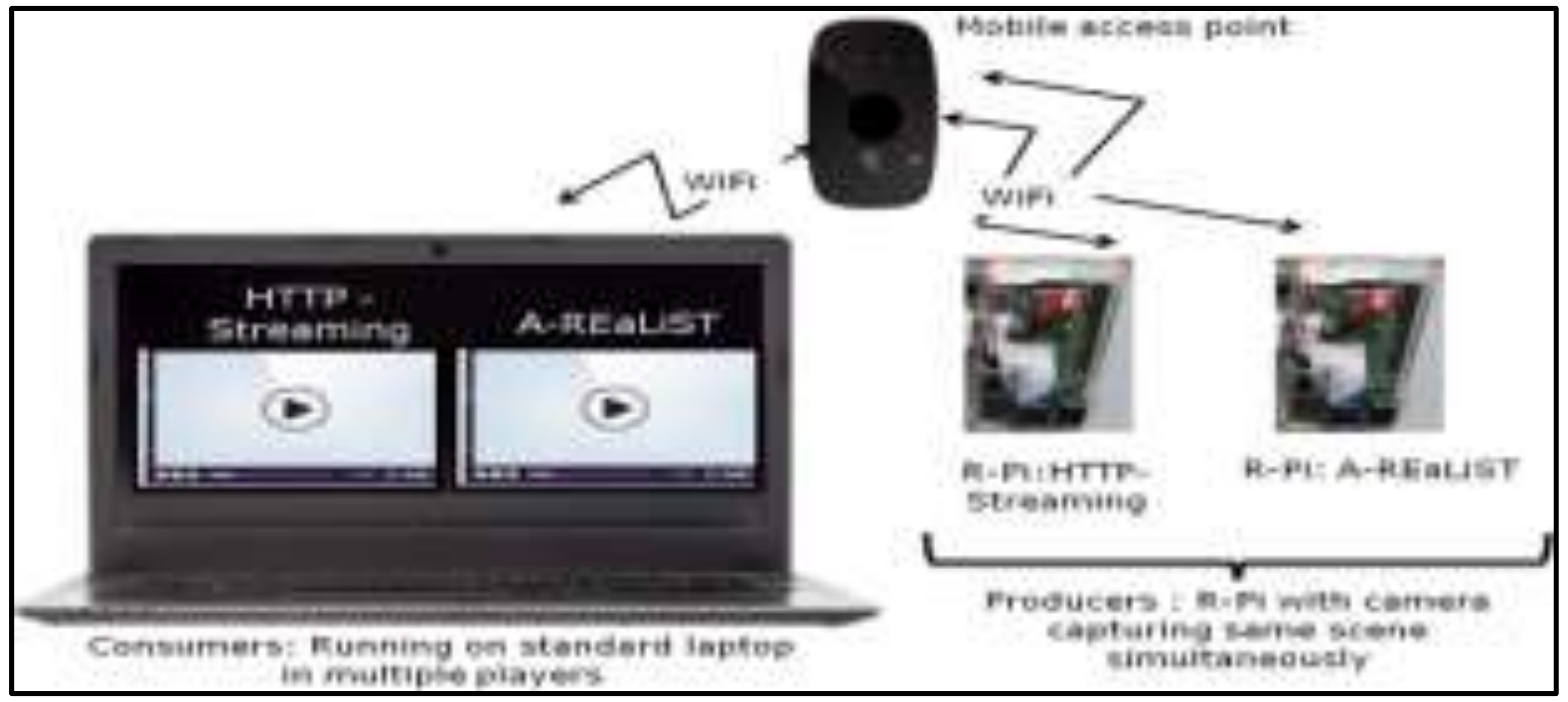


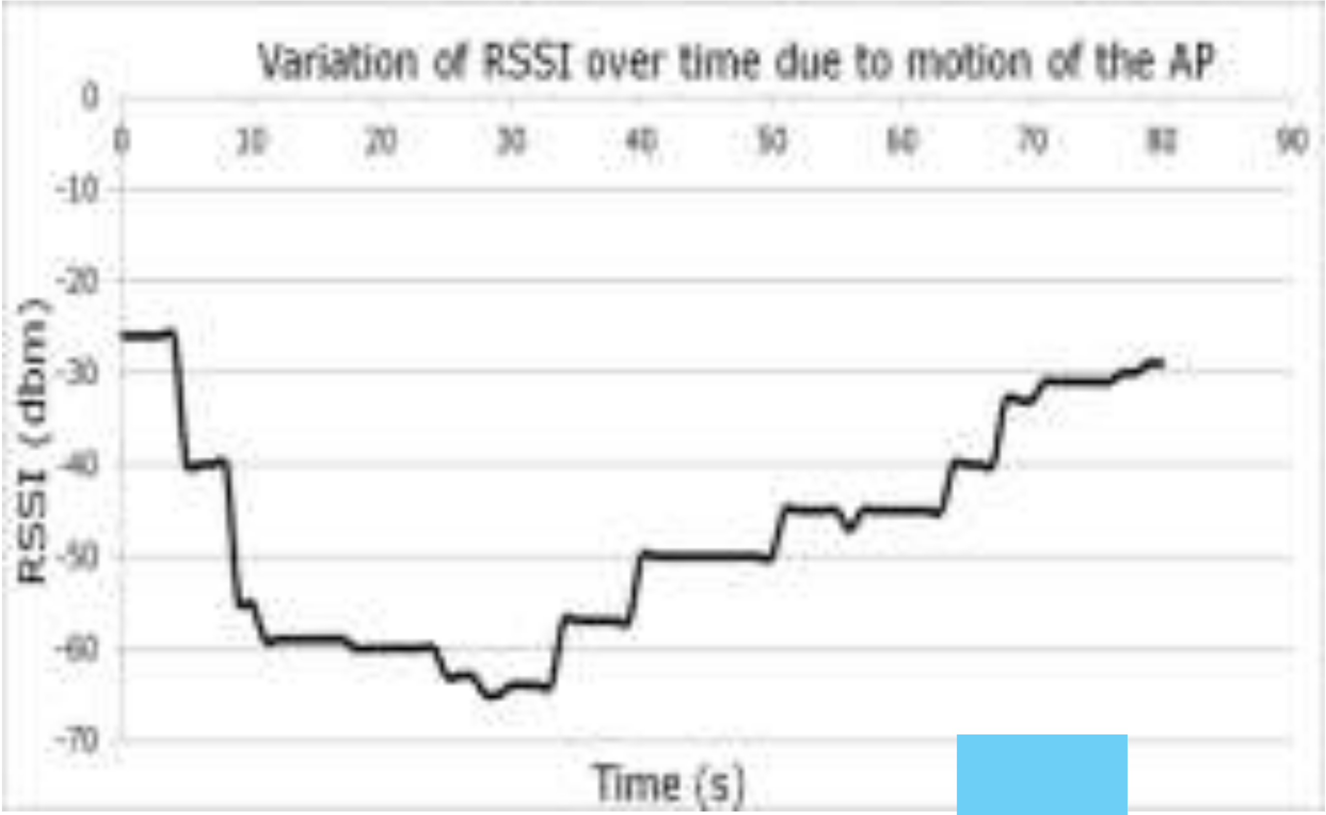
frame reception ratio (F) = = F_C / F_P; *F_C* = number of video frames actually received at *consumer*, *F_P* = number of video frames transmitted at the *producer*; indicates the amount of loss in the network reflected in the video frames.
overall bandwidth efficiency (E) =T_C / (B_{PTx} + B_{CTx}). Here, *T_C* = the total frame size received at the *consumer*. **B_{PTx}**=Total bytes transmitted by *producer*; **B_{CTx}**=Total bytes transmitted by *consumer*.
σ = Standard Deviation in Inter-frame Gap

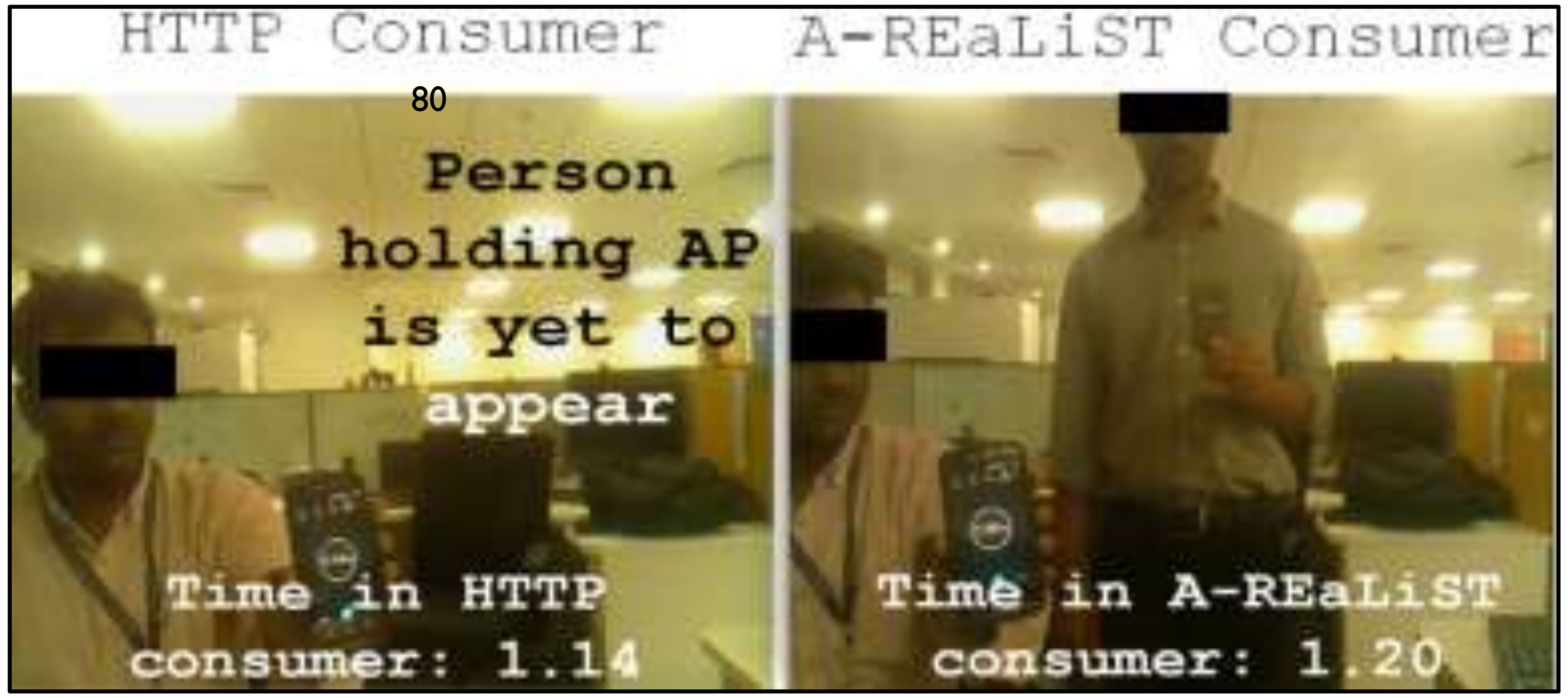
Compared against RTP also. Better PSNR.

Note: We have not used ABR in the experiments

Evaluation – With real APs


Mobile access point
WiFi
WiFi
HTTP - Streaming
A-REaLiST
R-Pi:HTTP-Streaming
R-Pi: A-REaLiST
Producers : R-Pi with camera capturing same scene simultaneously
Consumers: Running on standard laptop in multiple players





Person holding AP is yet to appear

Time in HTTP consumer: 1.14

Time in A-REaLiST consumer: 1.20

Thank you

Constrained RESTful Environments WG (core)

Chairs:

Jaime Jiménez <jaime.jimenez@ericsson.com>

Carsten Bormann <cabo@tzi.org>

Mailing List:

core@ietf.org

Jabber:

core@jabber.ietf.org

http://6lowapp.net

- **We assume people have read the drafts**
- **Meetings serve to advance difficult issues by making good use of face-to-face communications**
- **Note Well: Be aware of the IPR principles, according to RFC 8179 and its updates**

üBlue sheets
üScribe(s)

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/ (Privacy Policy)

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min) (old)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:52 Active drafts**
- **11:52–12:06 FASOR**
- **12:06–12:20 Streaming (Monday)**
- **12:20–12:20 Other new work**

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:36 OSCORE base, ERT, actuator**
- **11:36–11:56 Active drafts**
- **11:56–12:16 FASOR**
- **12:16–12:20 Other new work**

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:36 OSCORE base, ERT, actuator**
- **11:36–11:56 Active drafts**
- **11:56–12:16 FASOR**
- **12:16–12:20 Other new work**

OSCORE

draft-ietf-core-object-security-15

Status

- Version -15, submitted late August
- One DISCUSS left
- A few comments from Ekr via Alexey was brought to our attention this Sunday:

- **Comments about D.4 Unprotected Message Fields**:
  - ”Outer Code can be changed . . .” *Typically very concerning if you can change HTTP method.*
  - ”The server can verify what scheme was used in the last hop but not what was requested by the client . . ” *Why is that OK?*
  - ”Changing a NON to a CON, cause the receiving endpoint to respond. . .” *This seems obviously unsafe.*
- **Proposal:**
- Minor clarifications + New subsection in Appendix D: ”Threat Model”, following RFC 3552

- **Comment about storing security context parameters in non-volatile memory** → next slide

Write to Non-Volatile Memory

- Section 7.5 gives examples of how to handle loss of mutable security context
- Writing sequence number to NV memory
  - Simple write scheme: write if SEQ = 0 (mod K), then operation. Read after reboot, then add K.
- Issue: Unpredictable completion of write to NV memory

- **Proposal:**
- Expand on the alternatives to handle loss of security context
  - Including random number based
  - Add details to example in Appendix B.2
- Emphasize the issues
  - Update write scheme: add also term for upper bound of completing write
- Allow application to decide
  - Some devices may handle write to NV better than random numbers

Next Steps

- Push proposed resolutions to CoRE WG Github – Done
- Wait for further comments
- Submit version -16

Echo and Request-Tag

draft-ietf-core-echo-request-tag-03

Status

- Detailed review by Jim Schaad – thanks!
- Main changes since -02:
- Echo:
  - May be used by server in in multiple responses and by client in multiple requests
  - Detailing the OSCORE properties; independent Inner and Outer option
  - Methods in Appendix A updated
  - Clarifications
- Request-Tag
  - Stateless-proxy application
  - Clarifications
- Extended security and privacy considerations
- IANA considerations

- All known comments are adressed.

Controlling Actuators with CoAP

draft-mattsson-core-coap-actuators-06

Status

- Informational draft
- Merge of problem statements leading up to Echo and Request-Tag
- Does the WG want us to complete that?

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:36 OSCORE base, ERT, actuator**
- **11:36–11:56 Active drafts**
- **11:56–12:16 FASOR**
- **12:16–12:20 Other new work**

Hop-Limit

- draft-ietf-core-hop-limit-00 was submitted 2018-09-17
- Discussed at virtual interims; remaining concerns relayed to authors
- Now clarifying whether these are done or need a new revision before WGLC

FETCH & PATCH with SenML

draft-ietf-core-senml-etch-00

IETF 103

Updates since individual -03

- Clarified that SenML PATCH does not reach out (conceptually) to different resources, even if SenML names may map to such
  - Access control needs to be evaluated accordingly

New media types or not?

- -00 proposed new media types for FETCH/PATCH use
- -0x proposed to re-use basic SenML media types
  - Just define different semantics for these methods
  - Mapped nicely…
  - …except for deleting with PATCH and and missing values for FETCH
- Proposal: back to new media type(s)
  - Same media type for FETCH and PATCH?
  - Also CBOR?

How to delete with PATCH?

- “v”: null

  + JSON merge-patch style

  + Kinda clean

  - Variable types for SenML frowned upon

  - JSON type for “v” currently fixed to number in SenML (but not a big issue with the new media types)

- “vdel”: true

  - New tag required

  - Bit more verbose (in JSON)

  + Not having the problems of above

- “op”: “remove”
  - JSON Patch style
- Other options to consider?

draft-ietf-core-interfaces

IETF 103

Next

- Incorporate feedback received
- Rework all of the examples to reflect the most recent versions of senml and link-format
- Interim meeting review

draft-ietf-core-dynlink

IETF 103

Recent

- Reference implementation for the conditional observe attributes – C/C++
- Some learning about the interactions between attributes
- Logic based expression using interval time bounds
- Learning from developing the OCF version

Definitions for notification conditions

```
//notifiable.c
bool notifiable( Resource * r ) {

#define BAND r->band
#define SCALAR_TYPE ( num_type == r->type )
#define STRING_TYPE ( str_type == r->type )
#define BOOLEAN_TYPE ( bool_type == r->type )
#define PMIN_EX ( r->last_sample_time - r->last_rep_time >= r->pmin )
#define PMAX_EX ( r->last_sample_time - r->last_rep_time > r->pmax )
#define LT_EX ( r->v < r->lt ^ r->last_rep_v < r->lt )
#define GT_EX ( r->v > r->gt ^ r->last_rep_v > r->gt )
#define ST_EX ( abs( r->v - r->last_rep_v ) >= st )
#define IN_BAND ( ( r->gt <= r->v && r->v <= r->lt ) ||
                  ( r->v >= r->gt && r->gt >= r->lt ) ||
                  ( r->v <= r->lt && r->lt <= r->gt ) )
#define VB_CHANGE ( r->vb != r->last_rep_vb )
#define VS_CHANGE ( r->vs != r->last_rep_vs )
```

Logic expression

```
  return (
    PMIN_EX &&
    ( SCALAR_TYPE ?
      ( ( !BAND && ( GT_EX || LT_EX || ST_EX || PMAX_EX ) ) ||
        ( BAND && IN_BAND && ( ST_EX || PMAX_EX) ) )
    : STRING_TYPE ?
      ( VS_CHANGE || PMAX_EX )
    : BOOLEAN_TYPE ?
      ( VB_CHANGE || PMAX_EX )
    : false )
  );
}
```

Next

- Add a state diagram for the interactions between attributes
- Incorporate feedback received
- Provide observe attributes as query parameters to the observe request
- Restructure the draft; introduce observe attributes first, then dynamic links, then binding table implementation
- Add implementation notes about link state tracking
- Implementation may reuse observers and updates

draft-ietf-core-coap-pubsub

IETF 103

Next

- Incorporate feedback received
- Track the 4.29 response code draft
- Implementation experience?
- Interim mini-plugfest
    - f-interop could support VPN mode

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- **11:20–11:24 Intro, Agenda**
- **11:24–11:36 OSCORE base, ERT, actuator**
- **11:36–11:56 Active drafts**
- **11:56–12:16 FASOR**
- **12:16–12:20 Other new work**

draft-ietf-core-cocoa: Status

- Submitted to IESG 2017-12-16
  - Responsible AD here: Mirja Kühlewind (TSV AD)
  - Great AD feedback
- London IETF uncovered potential for misunderstanding
  Ran out of time resolving this in Montreal IETF
  Still not resolved, try again this week
  Will lead to –04
- Oops: it turns out there are different understandings between the CoCoA authors, too…
  - Puts validity of simulations and experiments in question
- ➔ Retract draft from IESG processing;
  ➔ new WGLC when this is fixed

FASOR Retransmission Timeout and Congestion Control Mechanism

draft-jarvinen-core-fasor

Ilpo Järvinen[*], Iivo Raitahila[*], Zhen Cao[†] and Markku Kojo[*]

[*]University of Helsinki [†]Huawei

core @ IETF-103
November 8, 2018

Introduction and Objectives

- FASOR (Fast-Slow RTO) balances between the contradictory goals in handling random loss and congestion
  - Triggers RTO fast in case of random losses
  - Triggers RTO slow enough to handle congestion
- In IoT deployments, congestion expected to occur mainly due to large number of parallel devices
  - Test such extreme congestion scenarios now rather than later
- Unlike default CoAP and CoCoA[*], FASOR is not vulnerable to Congestion collapse
  - But still outperforms them in cases with random losses

[*] Applies to CoCoA v03 and earlier. CoCoA's congestion collapse problem will be fixed by an upcoming update.

Problem with Current CoAP RTO Management

- Karn's algorithm: exponential backoff and keep the backed off RTO until unambiguous RTT sample acquired
- CoAP CC algorithms: exponential backoff but DO NOT retain the backed off RTO
- Default CoAP and CoCoA-v03 prone to Congestion collapse[⋆]
  - Unnecessary retransmissions occur persistently if RTT > RTO with the default congestion control algorithm
  - CoCoA not safe either but more complicated
    - Weak estimator hacks around the lack of retaining the backed off RTO (but RTO only updated if <3 rexmits were made)
    - Inflated RTT that triggers 3+ rexmits still causes the collapse
- Lack of retaining backed off RTO good for random losses though

[⋆] I. Järvinen, I. Raitahila, L. Pesola, Z. Cao, and M. Kojo, "Experimental Results with Default CoAP, CoCoA and CoAP over TCP RTO Management & Congestion Control," in *Proceedings of IETF101 / core WG*, Mar. 2018

I. Järvinen, I. Raitahila, Z. Cao, and M. Kojo, "Is CoAP Congestion Safe?," in *Proceedings of the Applied Networking Research Workshop 2018 (ANRW'18)*, July 2018

FASOR (Fast-Slow RTO) in Nutshell

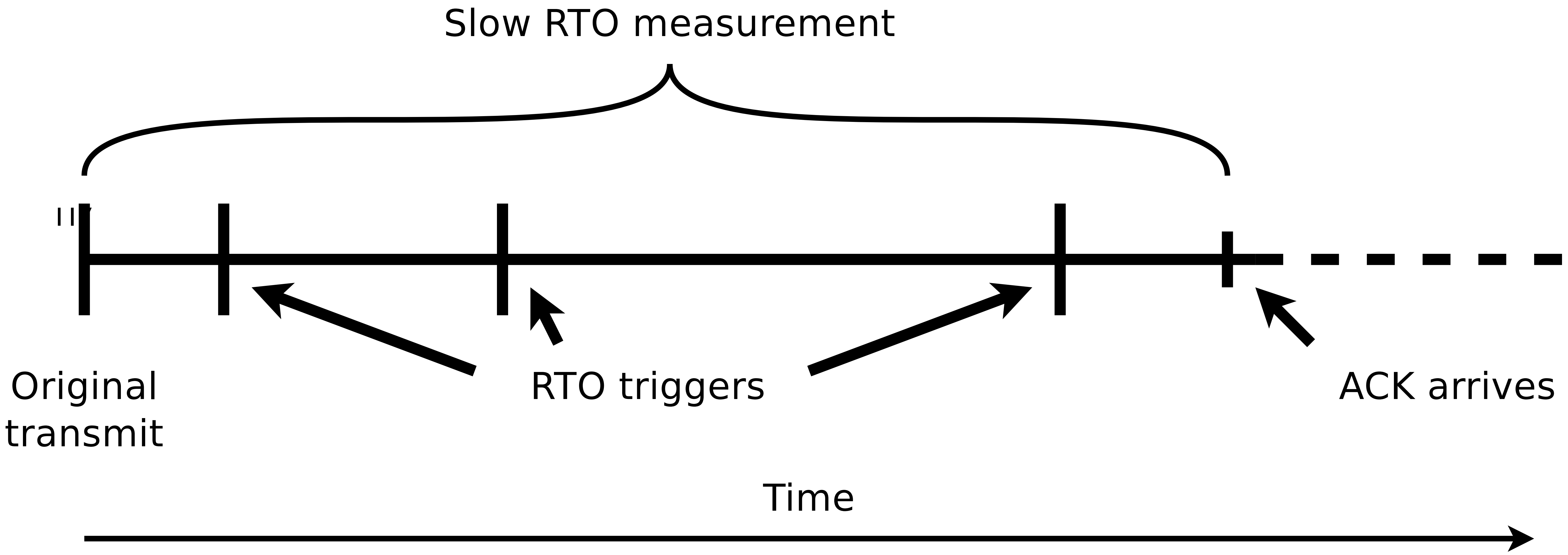
- FASOR (Fast-Slow RTO)[*] tries to find a good middle ground
  - Try to improve random loss
  - . . . but still handles congestion safely, including unnecessary rexmits
- Two ways to calculate RTO
  - FastRTO (normal RTO)
  - New SlowRTO
- New back off logic

[*] I. Järvinen, M. Kojo, I. Raitahila, and Z. Cao, "Fast-Slow Retransmission and Congestion Control Algorithm for CoAP," Internet Draft, Oct. 2018. Work in progress

I. Järvinen, I. Raitahila, Z. Cao, and M. Kojo, "FASOR Retransmission Timeout and Congestion Control Mechanism for CoAP," in *Proceedings of IEEE Globecom 2018*, Dec. 2018. To appear

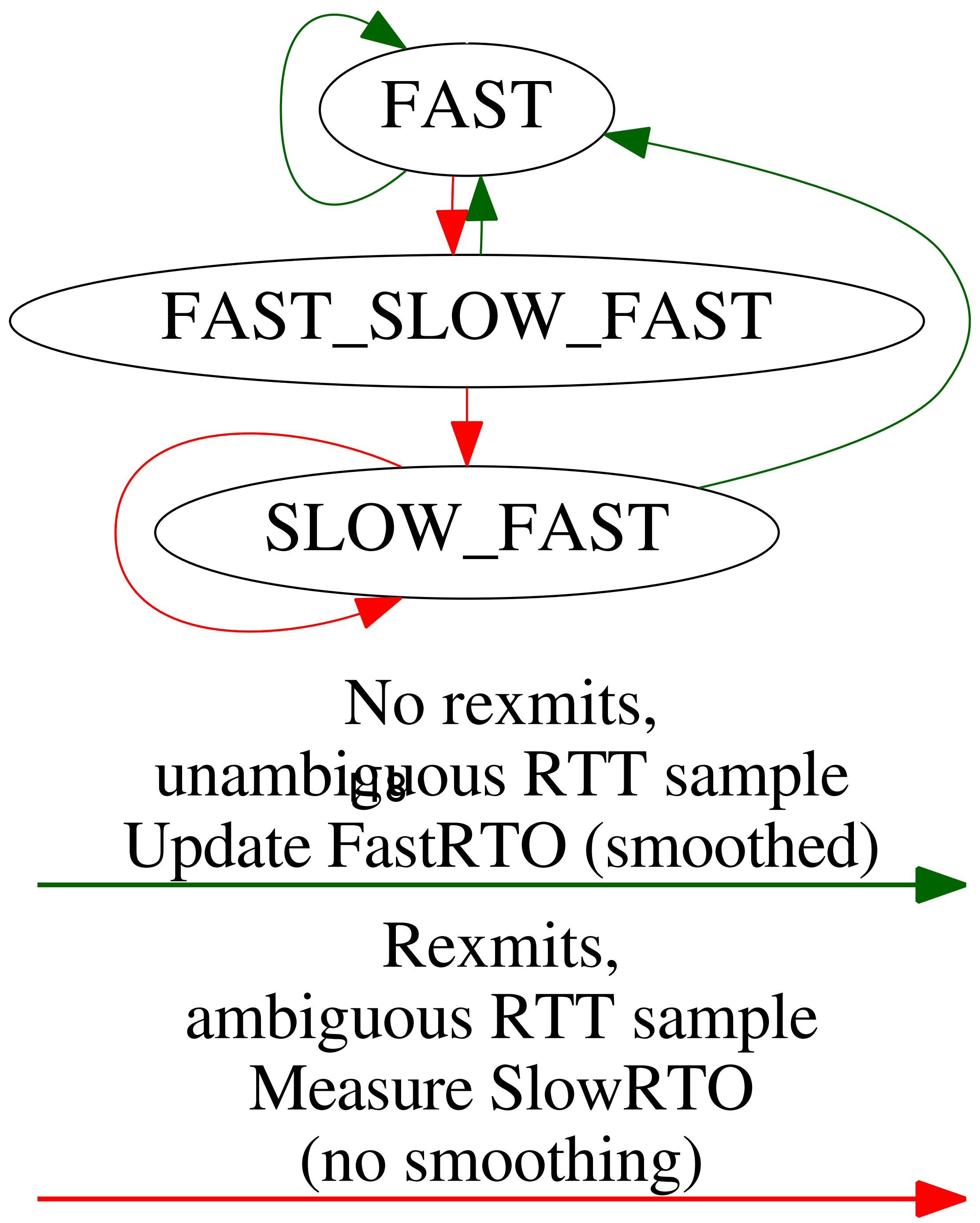
FastRTO and SlowRTO

- FastRTO ≈ RFC 6298 RTT/RTO computation
  - Initialization of RTTVAR changed to R/2K
    - Lowers RTO for short exchanges
- SlowRTO analogous to Karn's algorithm keeping RTO until unambiguous RTT sample
  - Measured when retransmissions were made as the time elapsed from the original copy
  - Multiplied by a factor to allow load growth (1.5 by default)
  - More conservative than Karn's algorithm



FASOR Back Off Logic

- Modify 2-state RTO logic of Karn's algorithm by adding a new state and modify back off series:



State	Back Off Series
FAST	FastRTO, FastRTO*2^1, FastRTO*2^2, ...
FAST_SLOW_FAST	FastRTO, max(SlowRTO, FastRTO*2), FastRTO*2^1, FastRTO*2^2, ...
SLOW_FAST	SlowRTO, FastRTO, FastRTO*2^1, FastRTO*2^2, ...

FASOR States

- FAST
  - “Normal” RTO series with exponential back off
  - When network state is not dubious
- FAST_SLOW_FAST
  - Probe first with FastRTO
    - Helps random loss cases to retransmit quickly
  - If no response and RTO expires, use SlowRTO as conservative back off
    - Allow draining unnecessary retransmissions from network
    - Due to lack of response so far, the sender cannot know if unnecessary retransmissions occurred or not
    - Safe and conservative option taken
  - If still more RTOs trigger, continue with the Fast RTO based exponential back off
- SLOW_FAST
  - Start with SlowRTO to acquire an unambiguous RTT sample with high probability

Optional Features

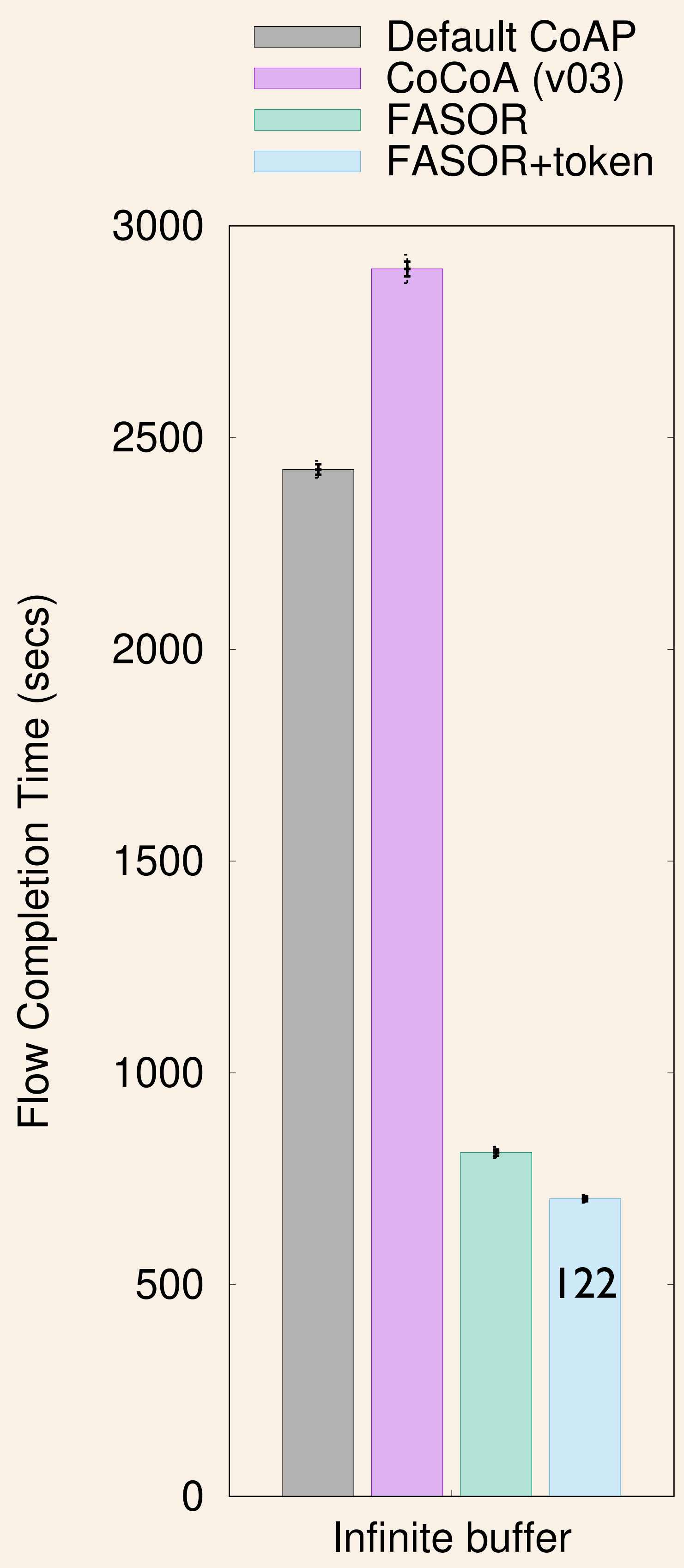
- Token/option variant
  - Encodes ordinal number of the transmissions for the request message to either token or option
  - Receiver echos the ordinal number back unchanged
  - Removes retransmission ambiguity problem
  - Allows accurate RTT estimation also with retransmitted messages

Test Setup

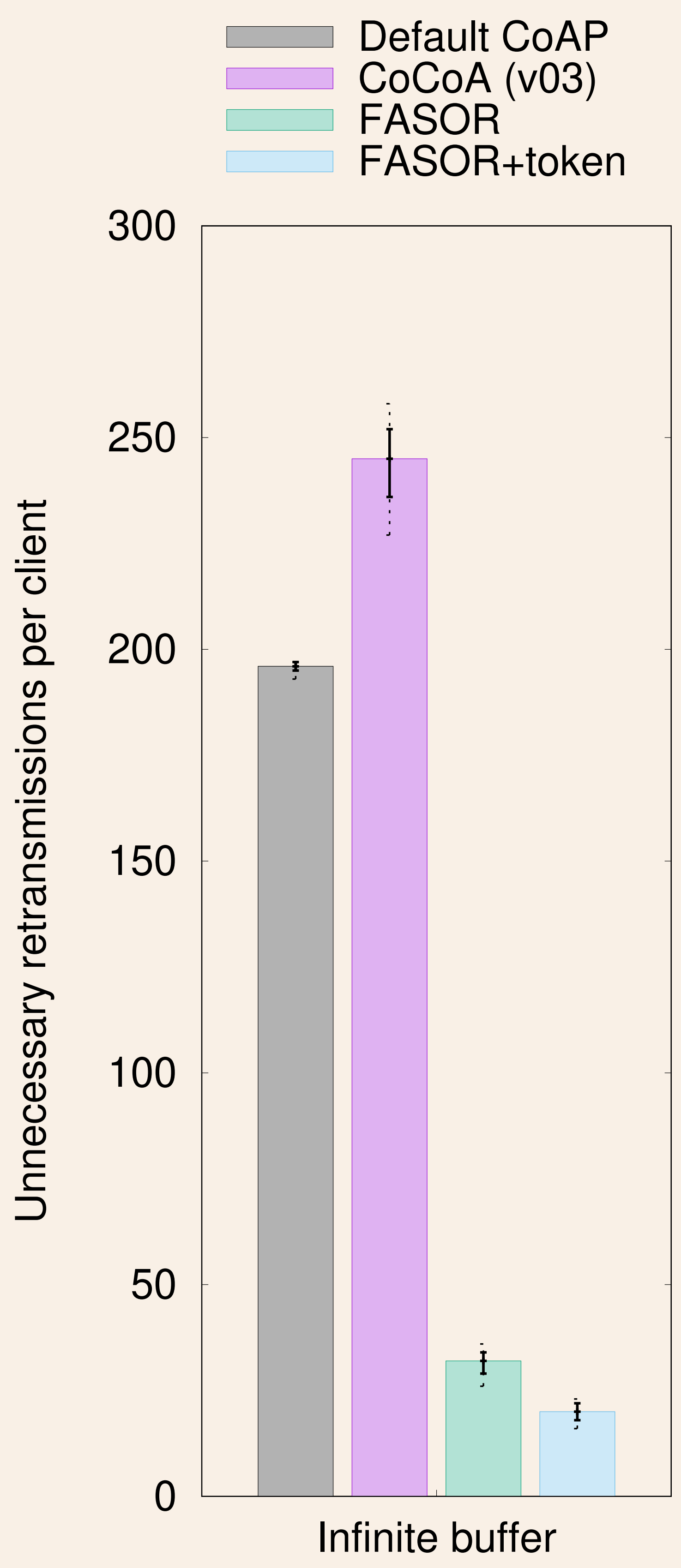
- Bottleneck BW: 30 kbps, base RTT ≈ 660 msecs
- Workload
  - A flow: a series of short-lived clients perform 50 request-responses exchanges in total
  - CC state reset after 1 to 10 message exchanges (new short-lived client starts)
  - Response payload: 60 bytes
  - CoCoA aging is disabled (aging is misapplied also for busy flows)
- Test scenarios
  - Heavy congestion and bufferbloat
    - Up to 400 parallel flows
    - Varying buffer size, including infinite buffer (1410000 bytes)
    - RTT ≈ 10 secs (for 400 clients + infinite buffer)
    - Error-free link
  - Random losses
    - 10 parallel flows
    - No congestion
    - 2-state error model: 0%/50% (medium) or 2%/80% (high)

Results with Heavy Congestion and Bufferbloat

FCT



Unnec. Rexmits

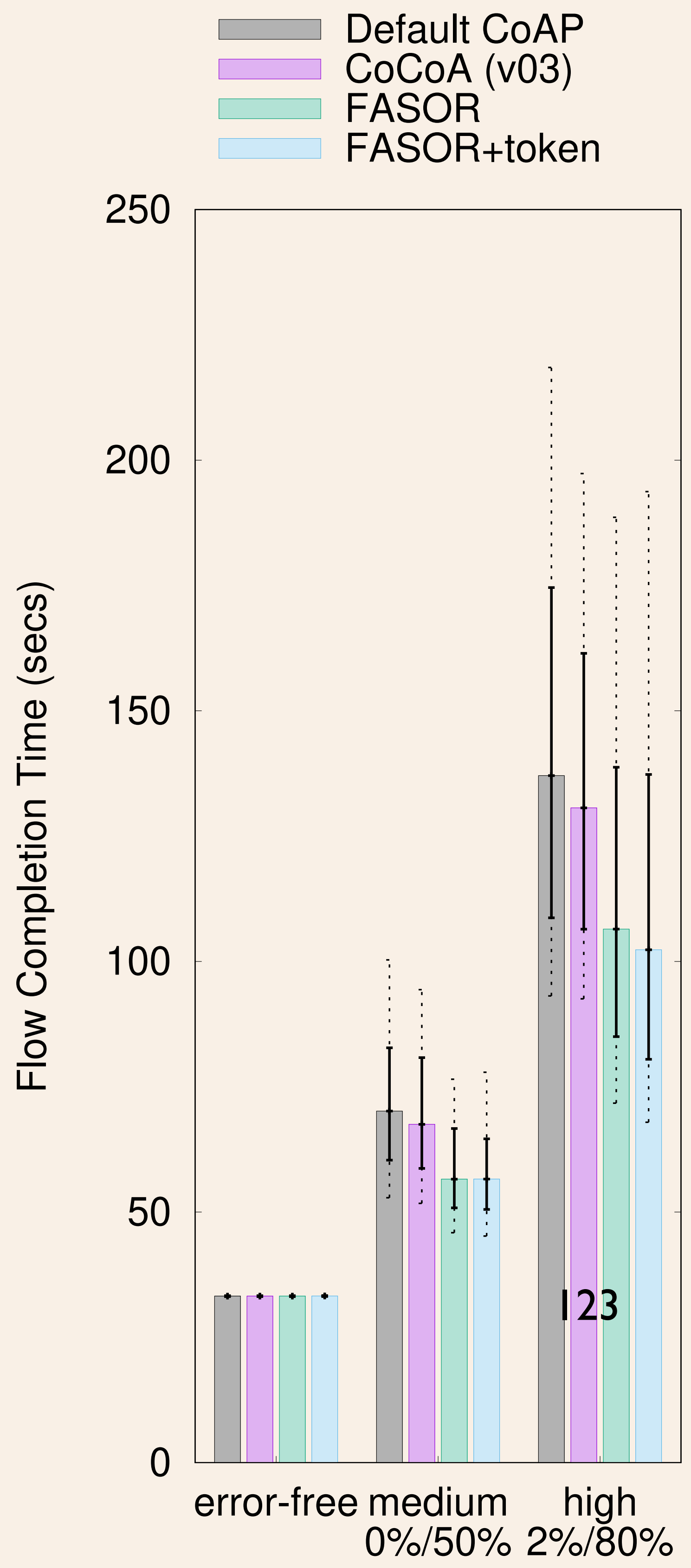


Observations

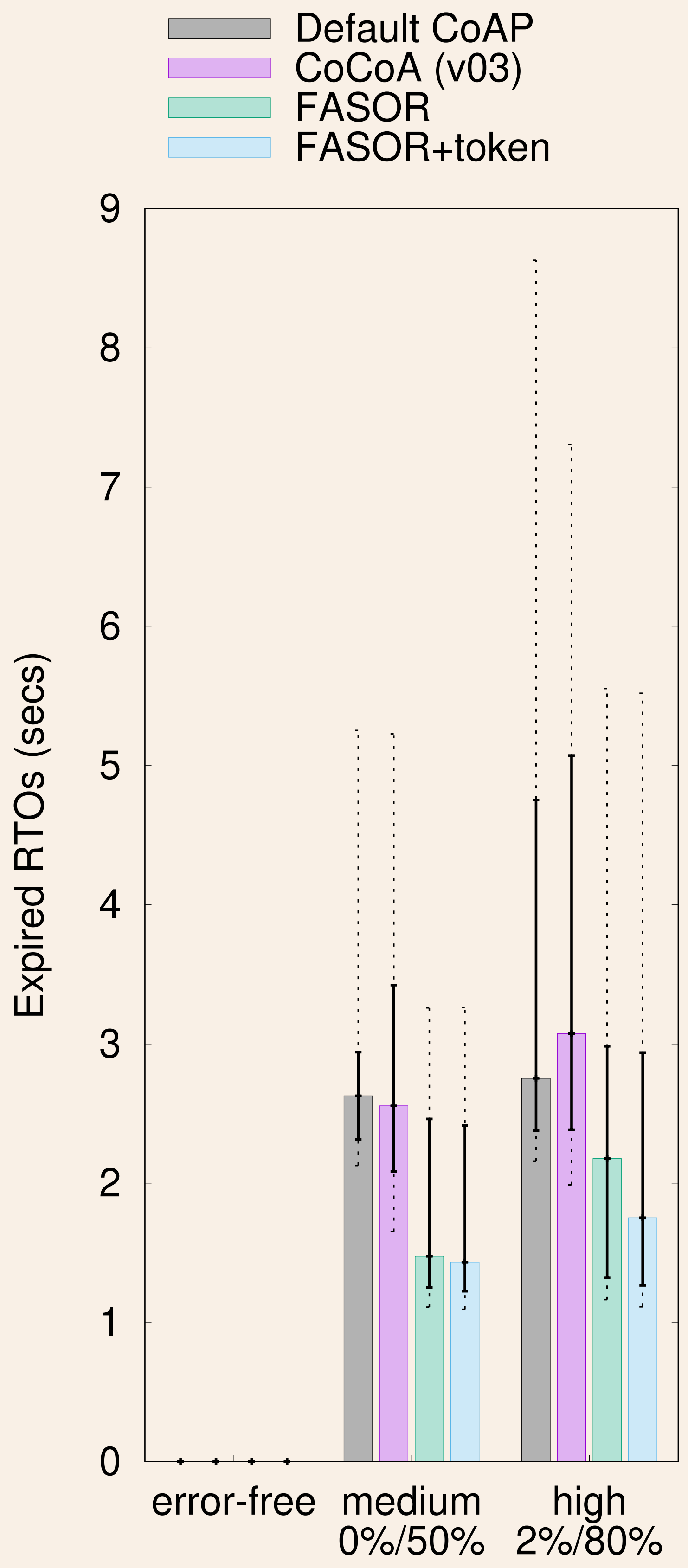
- FCT for Default CoAP and CoCoA-v03 long due to unnecessary rexmits
- Reduction in median with FASOR
  - FCT: 67%-76%
  - Unnecessary rexmits: 83%-91%
- Some unnecessary rexmits unavoidable when new client starts
- Similar pattern visible also in RTT

Results with Random Loss

FCT



Expired RTOs



Observations

- Median of the FCT shorter with FASOR:
  - medium: 16%-19%
  - high: 19%-25%
- FASOR is able to lower RTO value despite the challenging short-lived clients
- CoCoA's weak estimator measures random loss noise on ambiguous RTT samples
  - Its RTO values increase instead of converging towards the real RTT (≈ 660 msecs)

Work Items under Consideration

- FAST_SLOW_FAST back off series may currently be more aggressive than that of FAST state
  - A more conservative version has small but measurable performance impact
- Test with a dithering algorithm that is more similar to the standard dithering algorithm
  - Currently the specification matches with our current implementation
  - Dithering mostly orthogonal to the other parts of FASOR algorithm

Concluding Remarks

- FASOR achieves good balance between handling random losses efficiently and responding to congestion adequately in contrast to the other CC proposals
- Despite handling congestion safely, FASOR outperforms both default CoAP and CoCoA in cases with random losses
  - Making default CoAP and CoCoA congestion safe will likely have negative impact on their performance
  - Therefore, the performance gap is likely to become even larger
- Complexity of FASOR algorithm is comparable to that of CoCoA

- We believe FASOR would be beneficial for the ecosystem
  - Is there interest in this WG to work on this?

Backup Slides

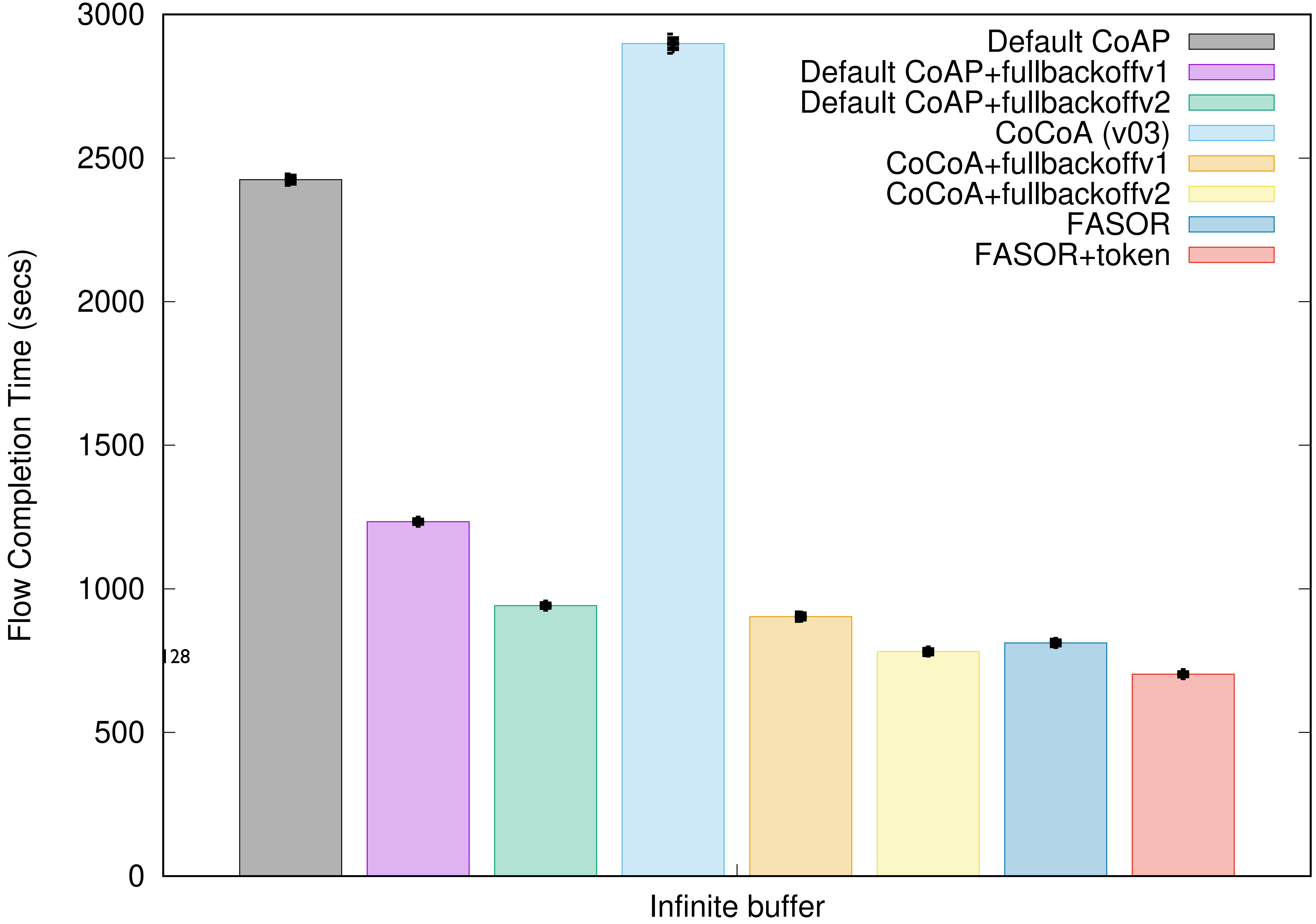
Backup Slides

- "Continuous" workload: 50 request-replies; does not reset CC state after 1 to 10 exchanges
- "Random" workload: 50 request-replies; CC state reset after 1 to 10 exchanges
- "Fullbackoff" variants[*] are congestion safe versions of default CoAP and CoCoA adding retaining RTO similar to Karn's algorithm

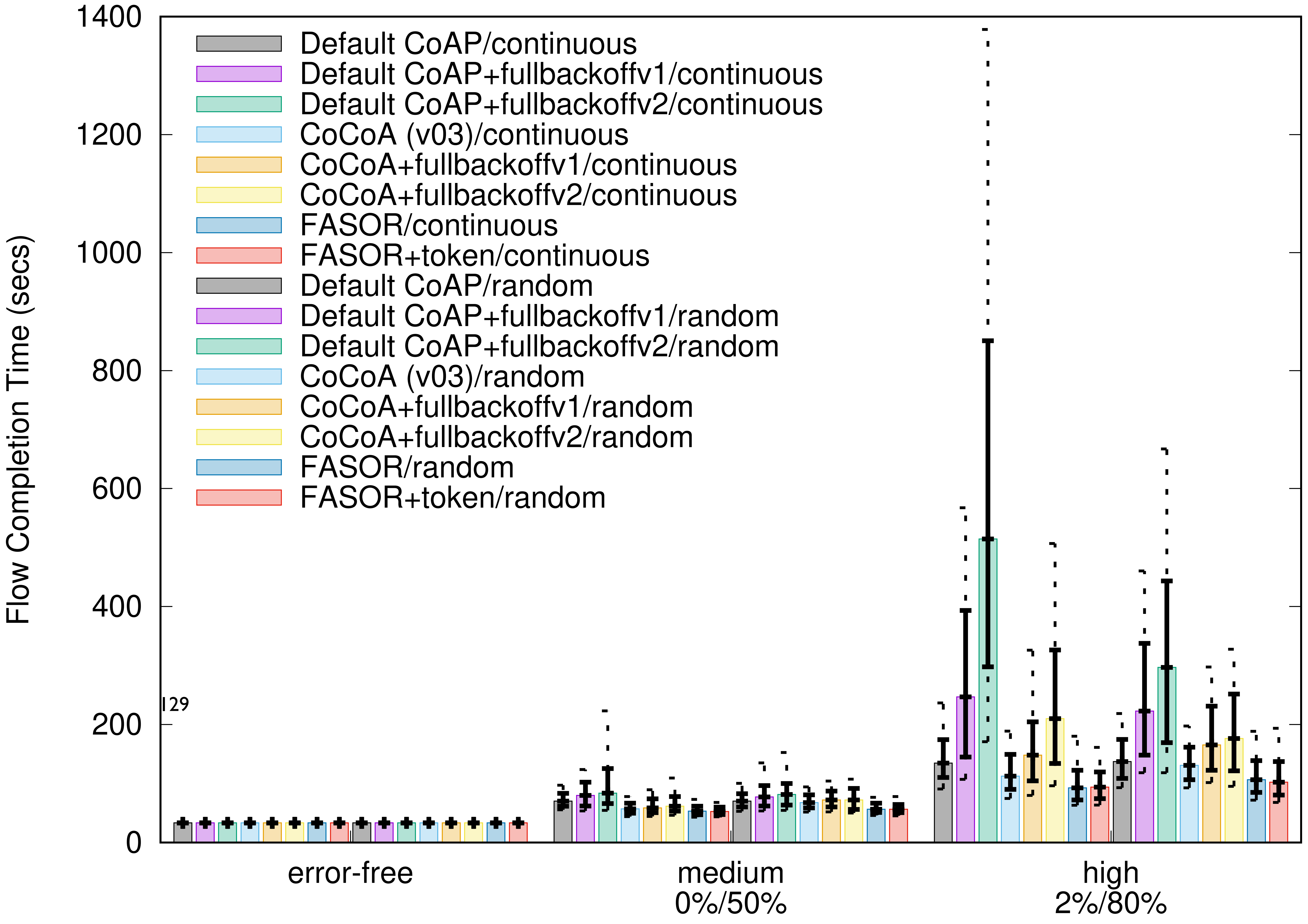
127

[*] I. Järvinen, I. Raitahila, Z. Cao, and M. Kojo, "Is CoAP Congestion Safe?," in *Proceedings of the Applied Networking Research Workshop 2018 (ANRW'18)*, July 2018

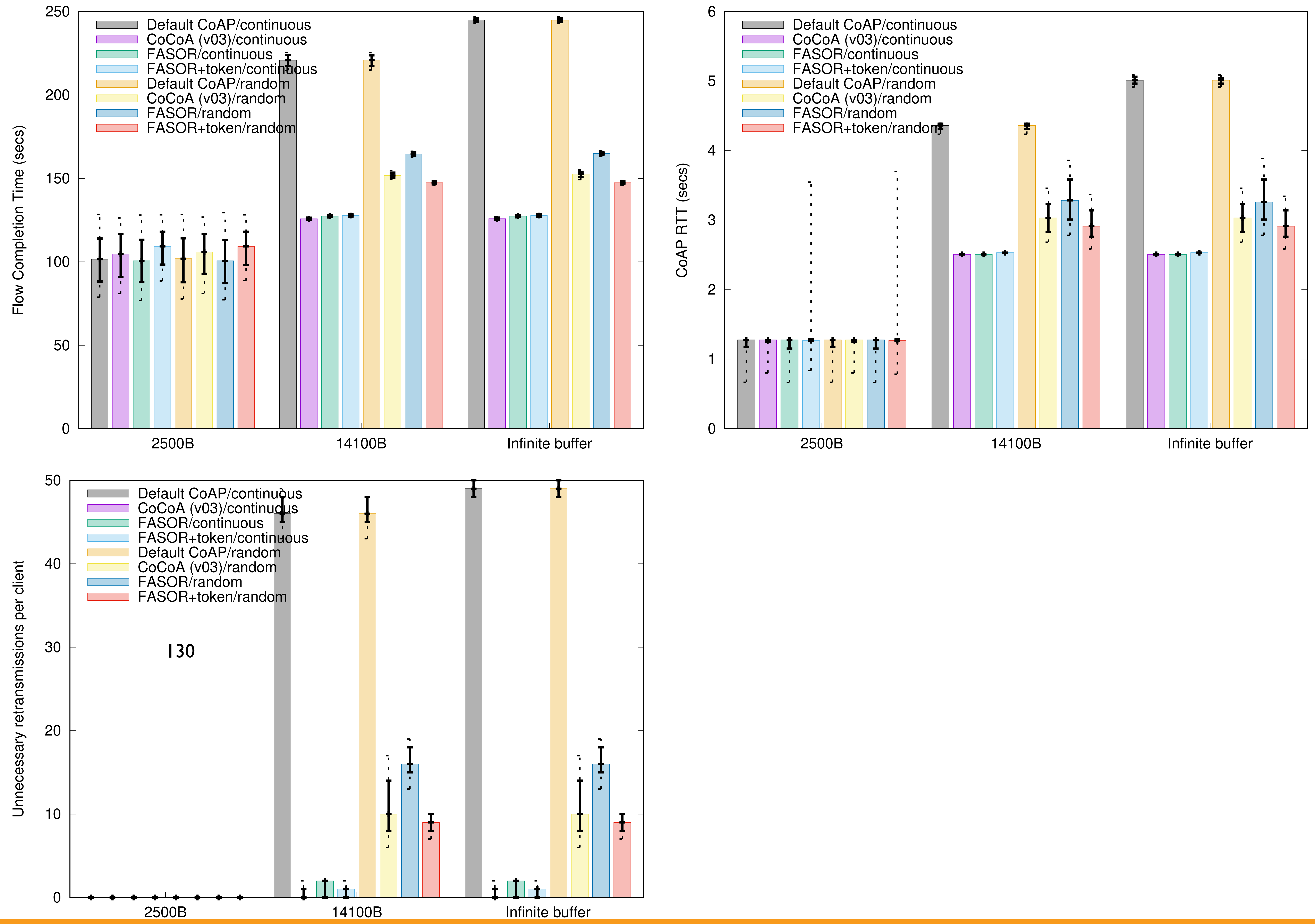
Backup Slides: Fullbackoff with Heavy Congestion



Backup Slides: Fullbackoff Variants with Random Loss

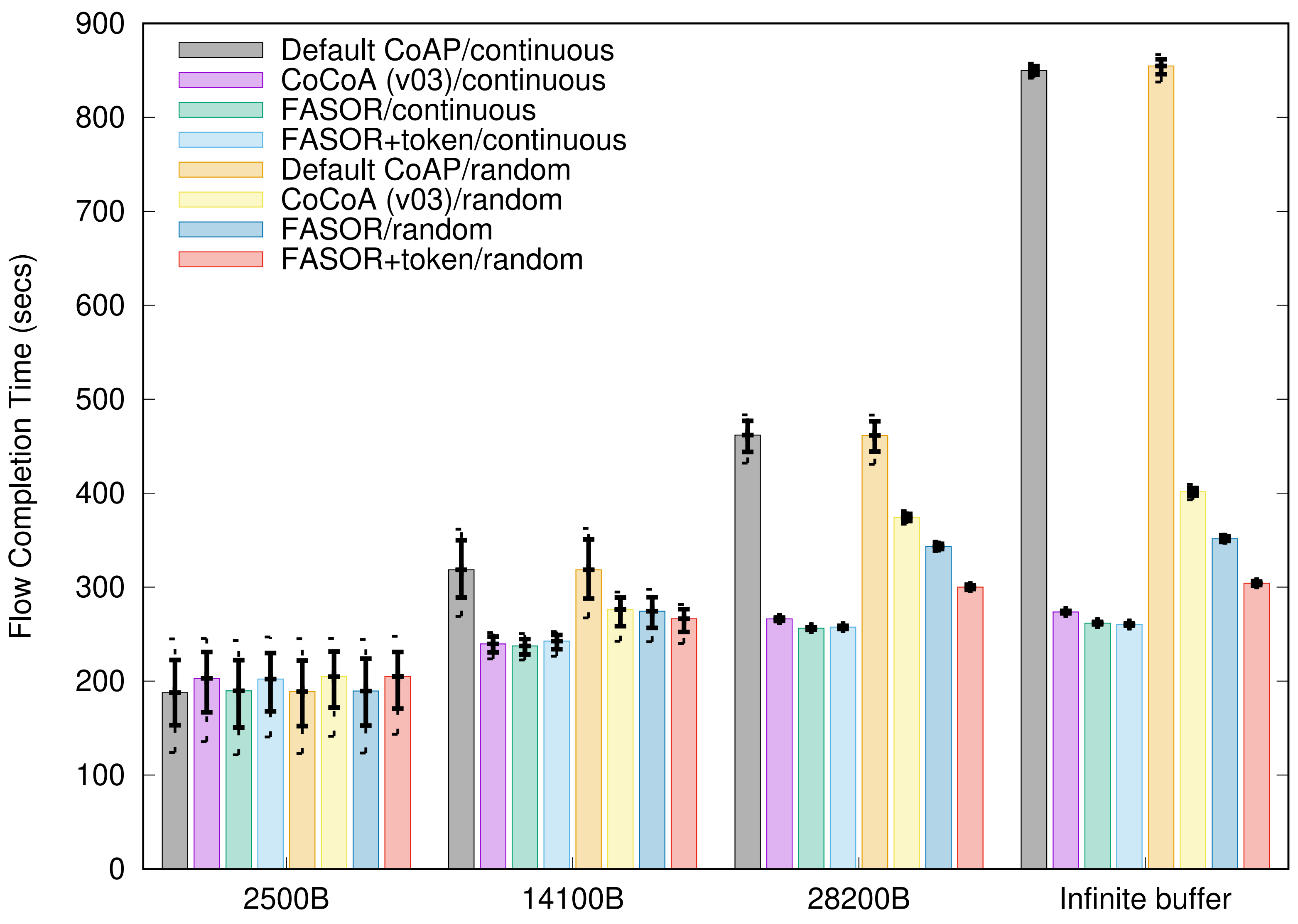


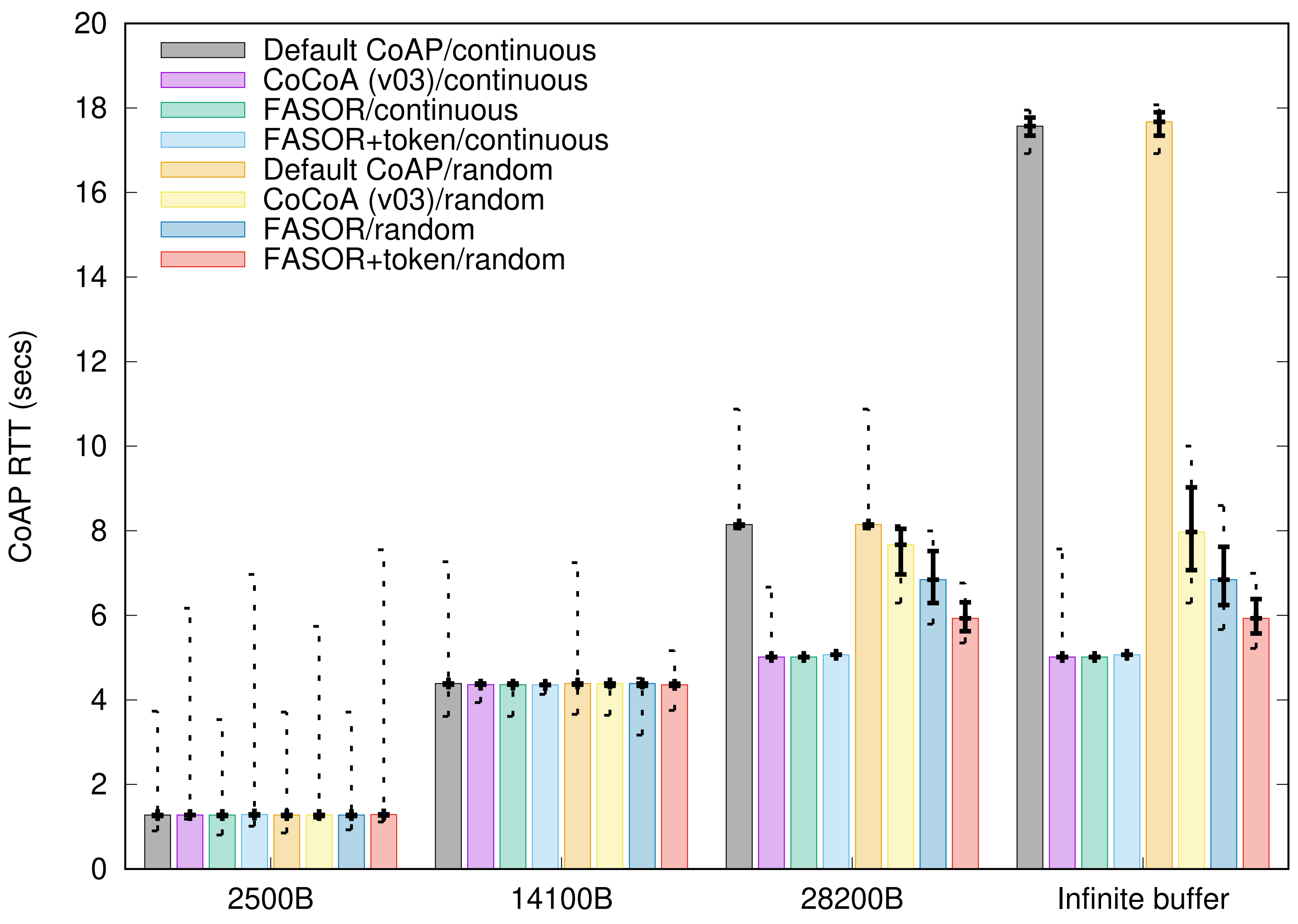
Backup Slides: 100 Parallel Flows

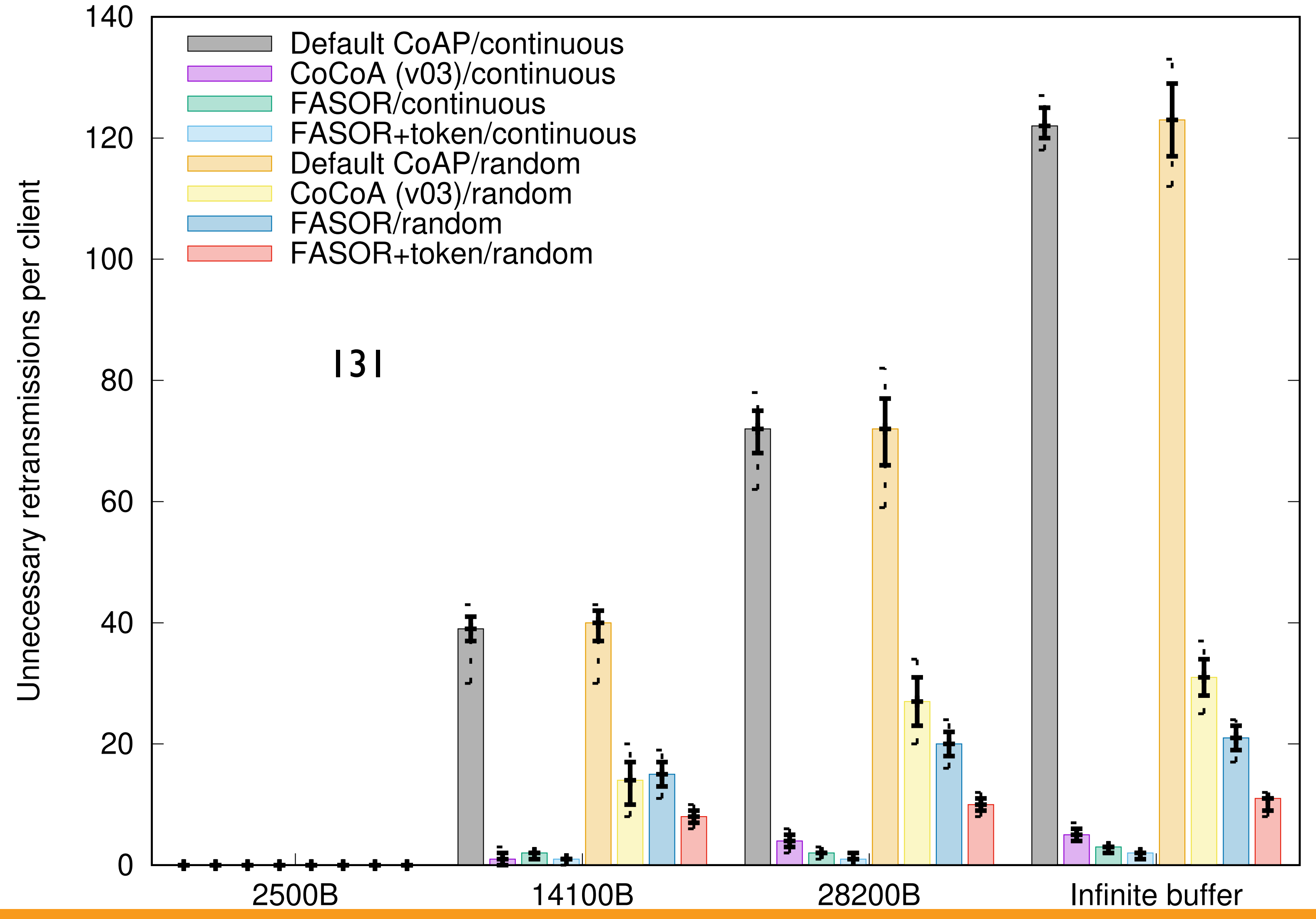


UNIVERSITY OF HELSINKI

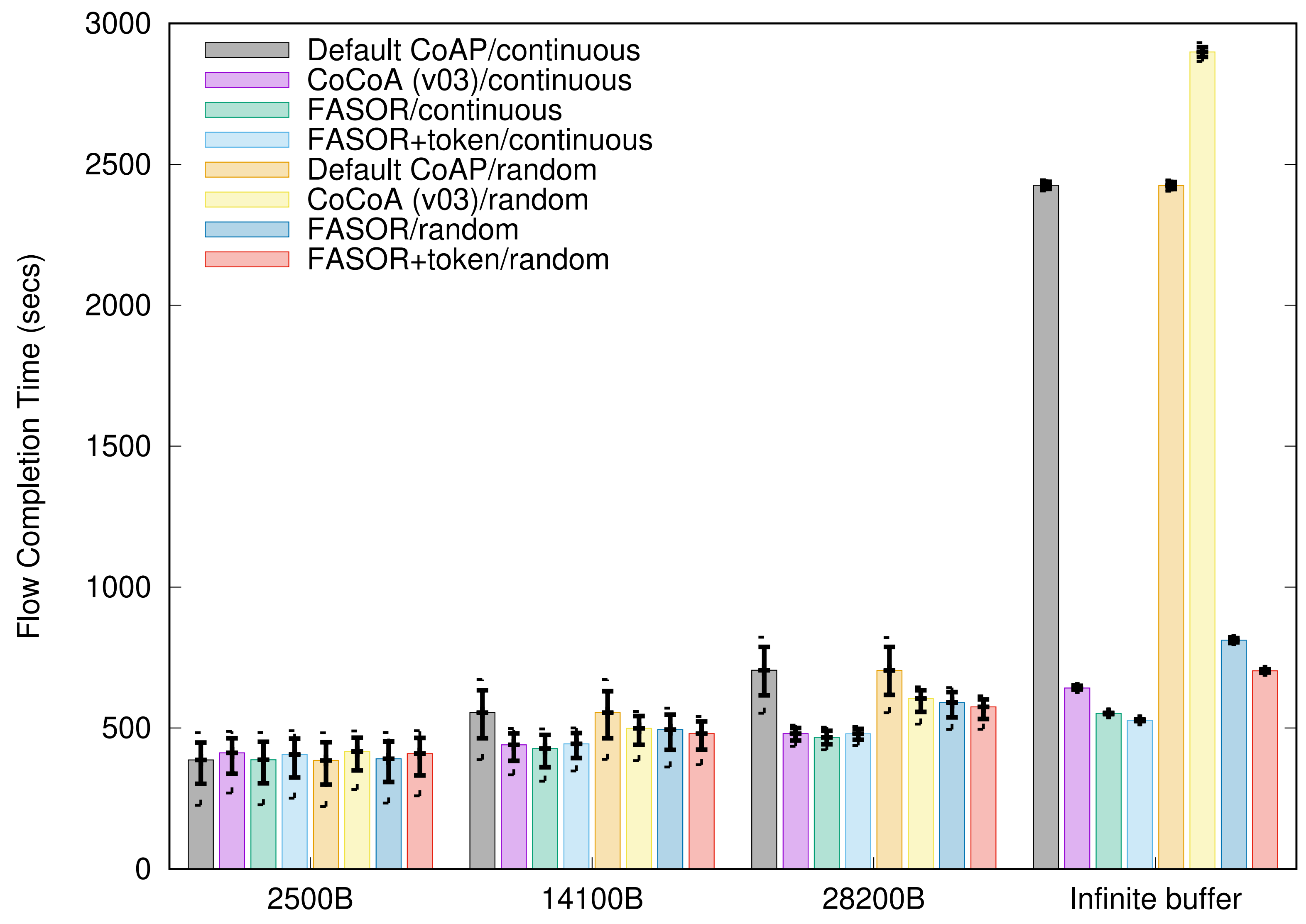
Backup Slides: 200 Parallel Flows

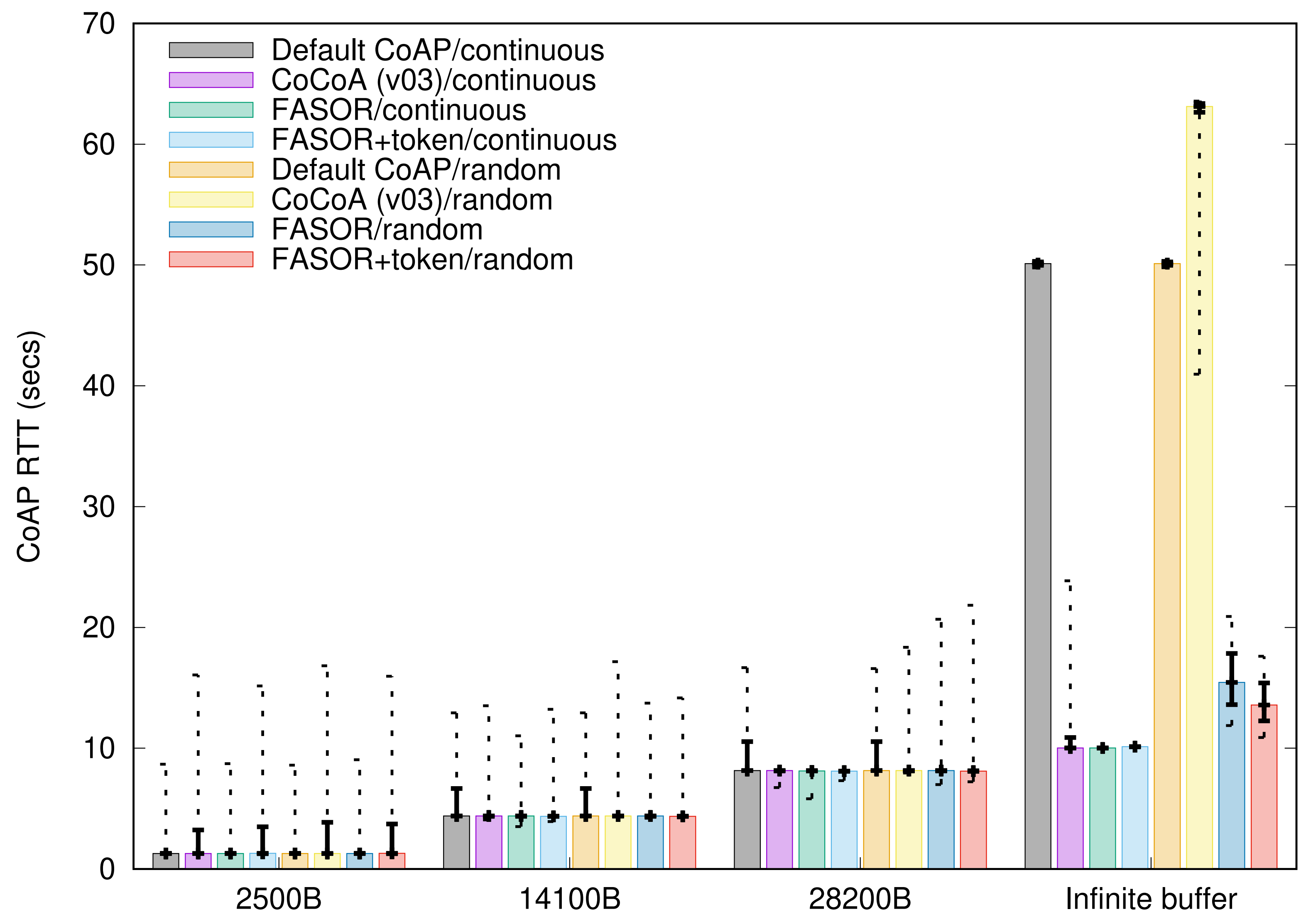


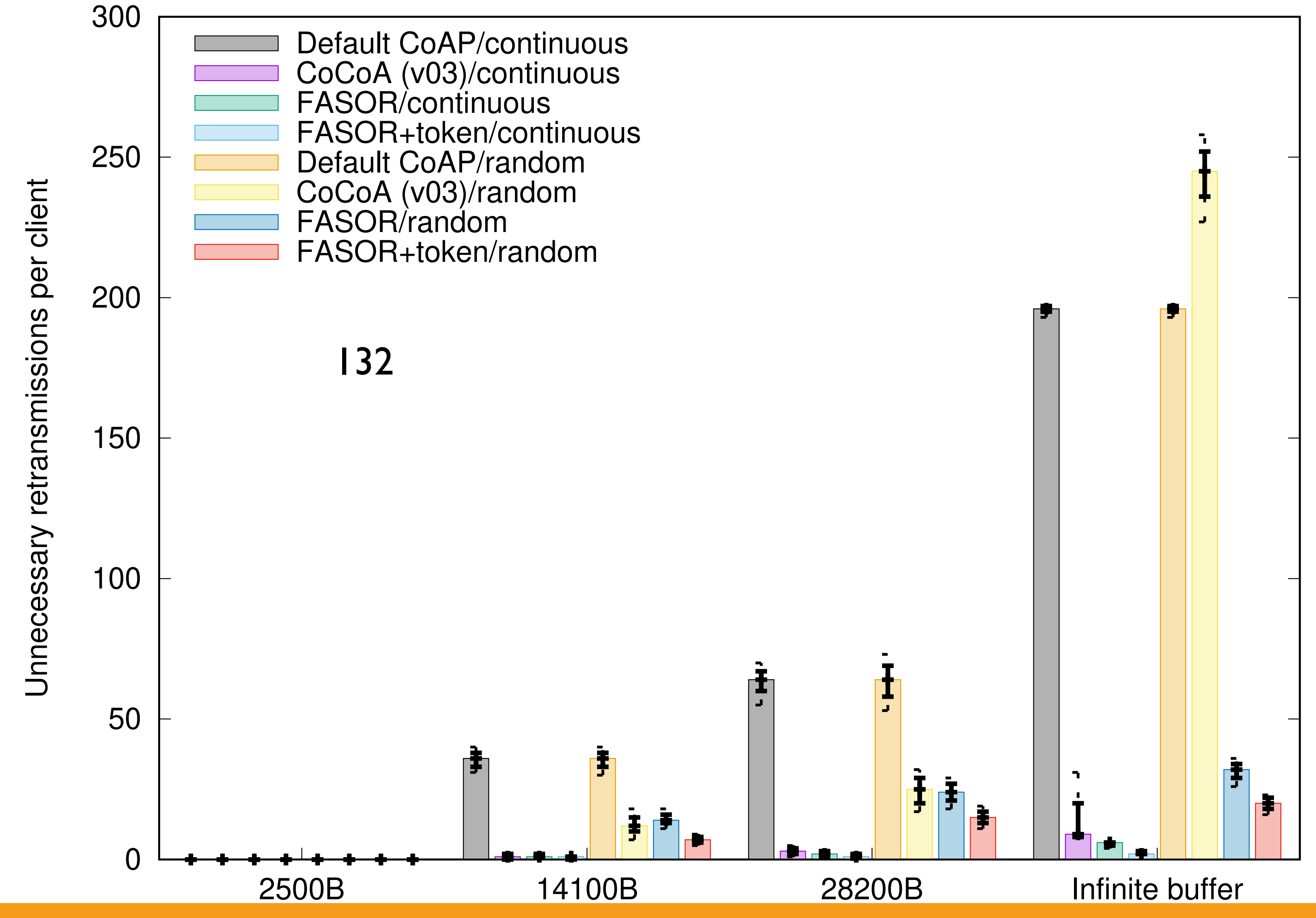




Backup Slides: 400 Parallel Flows







- fasor ready to adopt? Hmm: https://datatracker.ietf.org/ipr/3227/

- Do not discuss any patent claim now or on mailing list

- WG members need to form opinion and decide whether that is an obstacle to WG adoption

- (Claim owner can choose to speed up the process by providing more information.)

All times are in time-warped ICT (UTC+07:00)

Thursday (60 min)

- 11:20–11:24 Intro, Agenda
- 11:24–11:36 OSCORE base, ERT, actuator
- 11:36–11:56 Active drafts
- 11:56–12:16 FASOR
- 12:16–12:20 Other new work

Old

Signed assertions are expressed as X.509 certificates

New

Signed assertions are expressed as CWTs (RFC 8392) protected by COSE (RFC 8152)

CoIDs

(Concise IDs)
To replace X.509, fill in the small gaps left:
draft-birkholz-core-coid-00

— (Henk Birkholz, Carsten Bormann, Max Pritikin, Robert Moskowitz)