

A publish-subscribe architecture for the Constrained Application Protocol (CoAP)

draft-ietf-core-coap-pubsub-13

Jaime Jiménez, Ericsson

Michael Koster, KTC

Ari Keränen, Ericsson

IETF 118 meeting - Prague- November 9th, 2023

Draft History

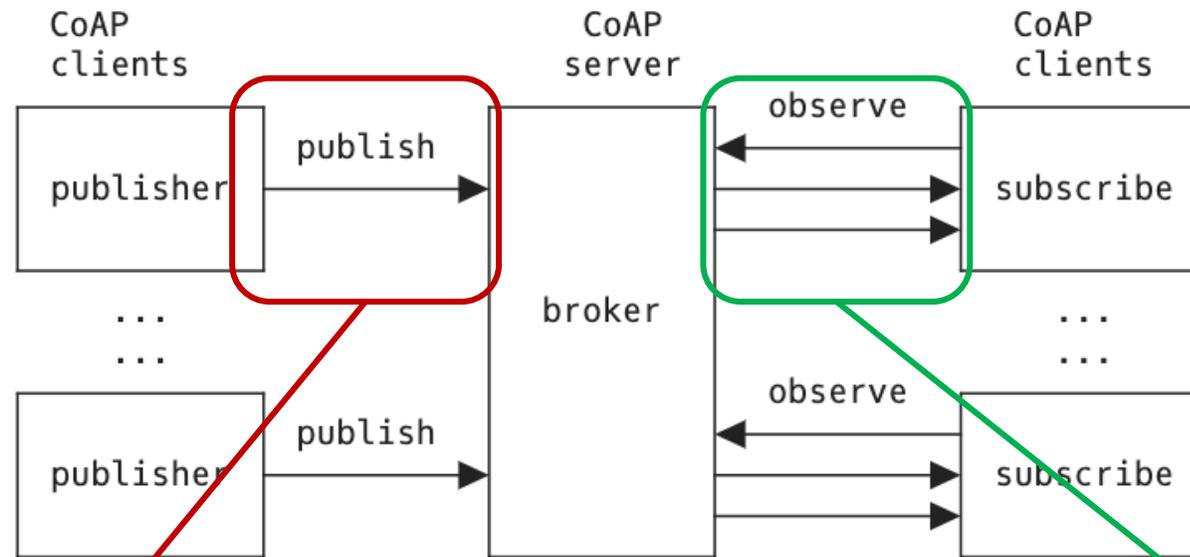
- Together with *core-interfaces* and *core-dynlink* among the “senior” working group drafts we have in CoRE (2016).
- Current design is inspired by [hartke-t2trg-coral-pubsub](#) and [ietf-ace-oscore-gm-admin](#)

Versions:



- Version (v12) introduced “topic configuration” operations. The publish-subscribe over CoAP principle remain very similar.
- Easy to implement, very complete CoAP implementations out there nowadays.

Recap: publish-subscribe in CoAP



```
0.03 PUT <broker_URI>/ps/data/225acdd =>
{
  "n": "temperature",
  "u": "Cel",
  "t": 1621452122,
  "v": 23.5
}
2.04 Changed <=
```

```
<= 0.01 GET <broker_URI>/ps/data/225acdd
Observe: 0
=> 2.05 Content Observe: 10001
[... Payload data...]
=> 2.05 Content Observe: 10002
[... Payload data...]
...
```

API Overview

Topic Collection resource

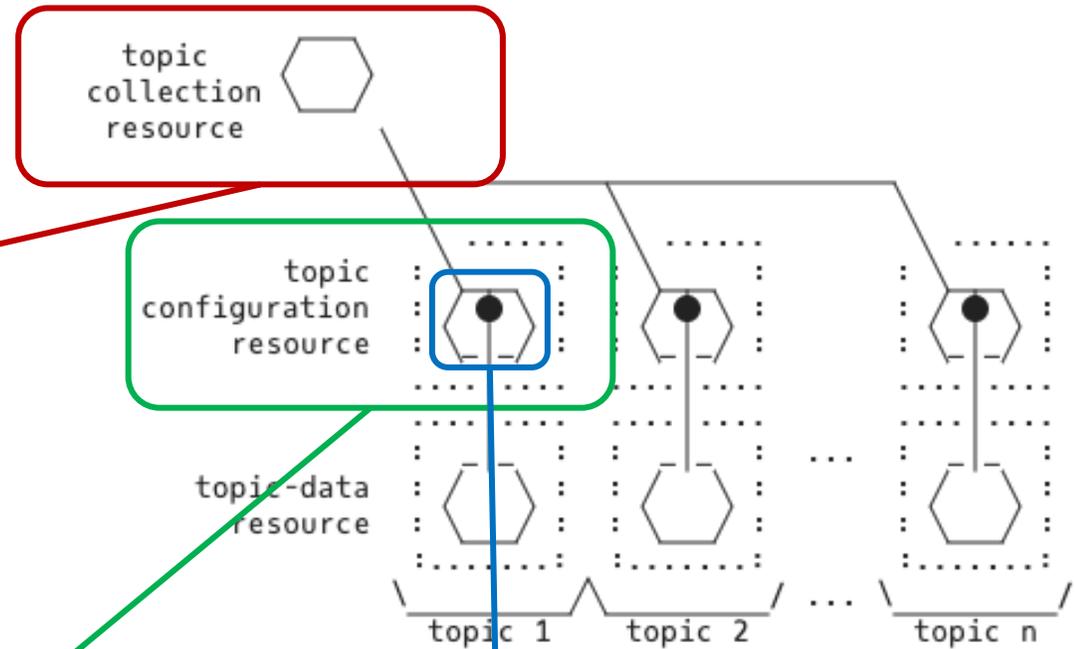
- Retrieve (GET) the list of topics
- Retrieve (FETCH) topics by properties
- Create (POST) a topic resource

Topic resource (configuration)

- Retrieve (GET) a topic resource
- Retrieve (FETCH) part of a topic with a filter
- Update (PUT) whole topic
- Update (PATCH) part of a topic with a filter
- Delete (DELETE) a topic resource

Topic Properties

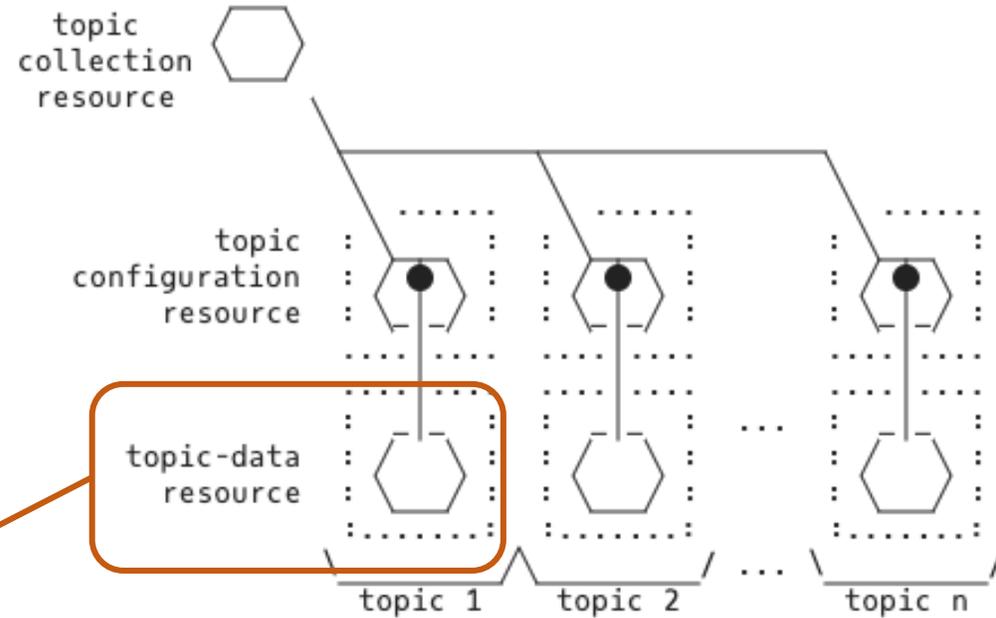
- Configuration parameters written by the administrator of the topic.
- Optional informational parameters (e.g., max-subscribers)



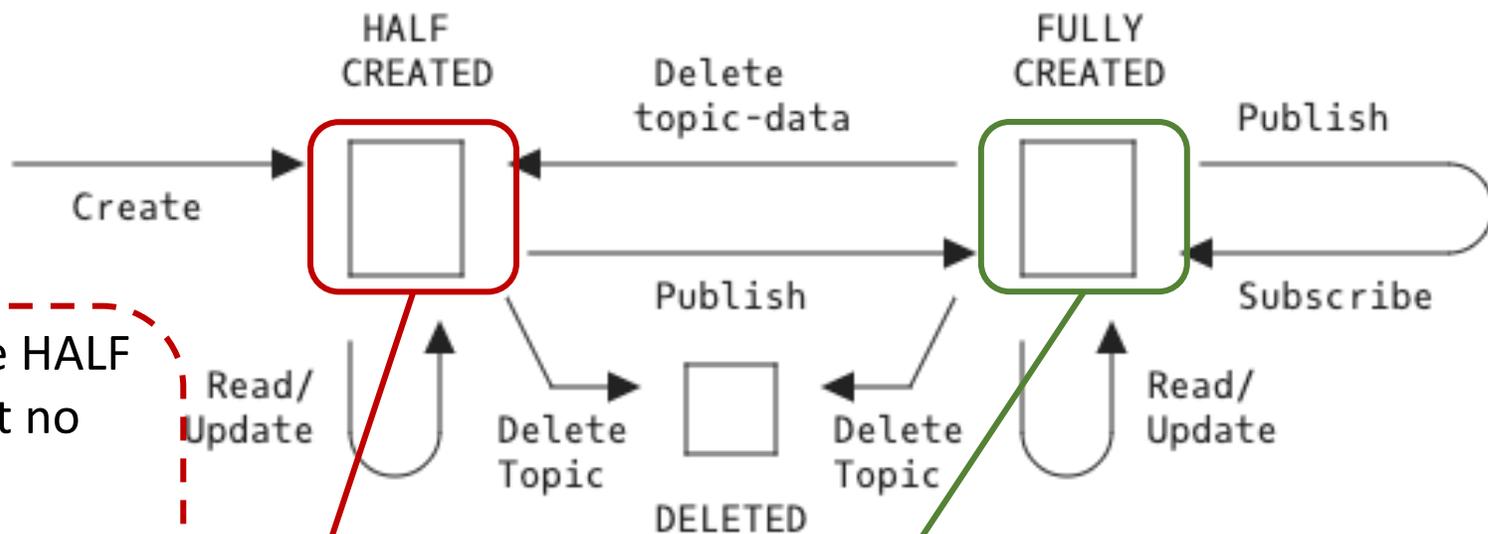
API Overview

Topic Data resource

- Publish (PUT) to a topic data (URI)
- Subscribe (GET + obs=0) to a topic data (URI)
- Unsubscribe (GET + obs=1) from a topic data (URI)
- Read latest value (GET)
- Delete (DELETE) a topic data



Topic Lifecycle



Topic configuration interactions, in the HALF CREATED state the topic is created but no data has been published to it.

=> POST /ps

```
{"topic-name": "Room Temperature Sensor", "resource-type": "core.ps.conf", "media-type": "application/json", "topic-type": "temperature", "expiration-date": "2023-04-05T23:59:59Z", "max-subscribers": 200, "observer-check": 86400}
```

<= 2.01 Created
location: /ps/9b7888

```
{  
  "topic-name": "Room Temperature Sensor",  
  "topic-data": "ps/data/7a0e64d",  
  "resource-type": "core.ps.conf",  
  "media-type": "application/json",  
  "topic-type": "temperature",  
  "expiration-date": "2023-04-05T23:59:59Z",  
  "max-subscribers": 200,  
  "observer-check": 86400  
}
```

A publisher publishes on the topic data resource **ps/data/7a0e64d**

=> PUT /ps/data/7a0e64d

```
{n: temperature,u: Cel,t:  
1621452122,v: 21.4}
```

The state changes to FULLY CREATED.
Subscribers can now subscribe and publish on that resource.

Hackathon Implementation

github.com/jaimejim/aiocoap-pubsub-broker

IETF 117

A simple python implementation of the topic discovery, configuration and pub-sub topic data interactions on top of [aiocoap](#).

The broker implements the following resource classes:

- CollectionResource: The collection resource /ps for storing topics.
- TopicResource: A resource for [topic configurations](#).
- TopicDataResource: A resource for topic data and for the [publish-subscribe interactions](#) over CoAP.

IETF 118

- Updates based on spec changes
- iPATCH to partially update topic configuration
- FETCH on the topic collection
- DELETE
- rt-based discovery (core.ps.coll, core.ps.conf, core.ps.data)

Next Steps for v13

- IANA section
- Use all of max-age, etc, correctly.
- Security section + references to ACE draft
- Use CBOR on the implementation.
- Implement missing operations.

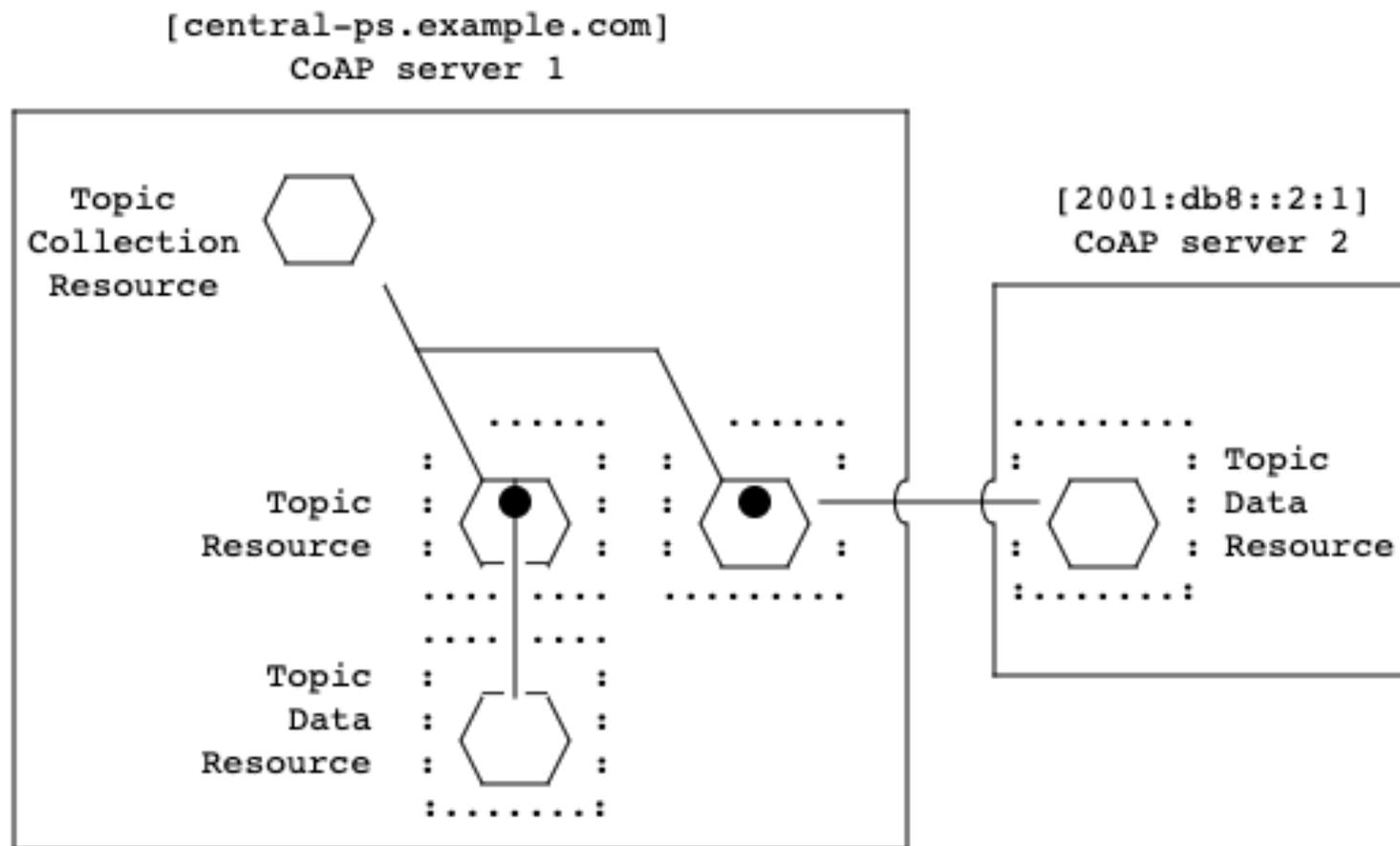
... and ...

- Topic configuration and data resources can be hosted on different servers, reflect that on the draft.

Next Steps for v13 - [diff here](#)

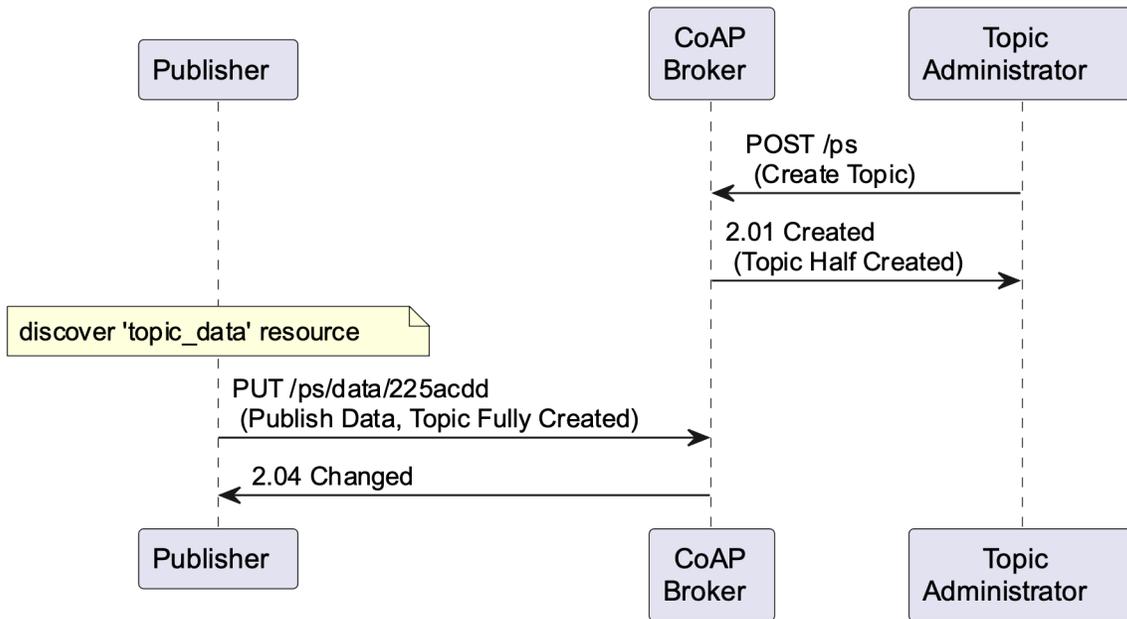
- IANA section ✓
- Use all of max-age, etc, correctly ✓
- Security section + references to draft-ietf-ace-key-groupcomm, draft-ietf-ace-pubsub-profile ✓
- Use CBOR on the implementation. ✓
- Implement missing operations. ✓
- Added “observer-check” field to regulate subscriber’s list ✓
- Added Topic collection discovery section ✓
- Added & updated examples ✓
- Addressed items on github issue tracker ✓
- Clarified much of the draft text ✓
- Addressed Marco’s (now contributor!) and Oscar’s review comments ✓
- Topic configuration and data resources can be hosted on different servers, reflect that on the draft. **X suggest to keep topic-data at broker**
 - Too complex as state then needs to be kept between the host of the topic-data resource and the broker. New protocol interactions needed for that.
 - **Suggestion:** make sure this draft allows for it but let’s define that in another draft.

(opt slide) Next Steps for v13 - [diff here](#)



(opt slide) Workflow Example

Create a Topic



Interact with a Topic

