Industrial IoT

Towards Resilience with ICN-Based Pub/Sub Systems

Cenk Gündoğan
cenk.guendogan@haw-hamburg.de

Department Informatik
Internet Technologies Group
HAW Hamburg

December 13, 2016

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Cenk Gündoğan

[dʒeŋc gy'ndo:an]

Freie Universität Berlin

- B.Sc. Computer Science
- M.Sc. Computer Science
- RIOT Collaborator

HAW Hamburg – Aug '16

- Research Assistant at Internet Technologies (INET) WG

Wat haste jemacht mit dein Leben, Wilhelm?

— Hauptmann von Köpenick

Jenkins Gündoğan

[dʒeŋc gy'ndo:an]

Freie Universität Berlin

- B.Sc. Computer Science
- M.Sc. Computer Science
- RIOT Collaborator

HAW Hamburg – Aug '16

- Research Assistant at Internet Technologies (INET) WG

Wat haste jemacht mit dein Leben, Wilhelm?

— Hauptmann von Köpenick

Agenda

Industrial Internet of Things (IIoT)

IPv6 IoT Protocol Suite for the IIoT

Information-Centric Networking (ICN) for the IIoT

Publish — Subscribe in ICN/NDN

Agenda

Industrial Internet of Things (IIoT)

IPv6 IoT Protocol Suite for the IIoT

Information-Centric Networking (ICN) for the IIoT

Publish — Subscribe in ICN/NDN

The Internet of Things (IoT)

Kevin Ashton – 1999 – describes the **Internet of Things (IoT)** as a system where the physical world is **connected to the Internet** via **ubiquitous sensors**.

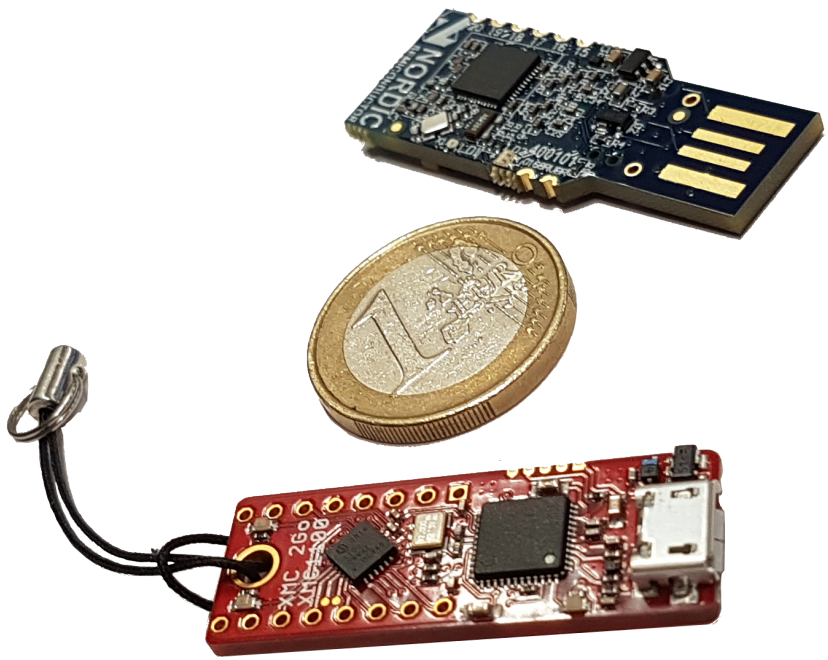
Hardware & Software

Kevin Ashton – 1999 – describes the **Internet of Things (IoT)** as a system where the physical world is **connected to the Internet** via **ubiquitous sensors**.

Hardware & Software

resource-constrained Things

- ROM: $\lesssim$ 250 KiB
- RAM: $\lesssim$ 50 KiB
- CPU: 72 MHz - 216 MHz
- battery-operated

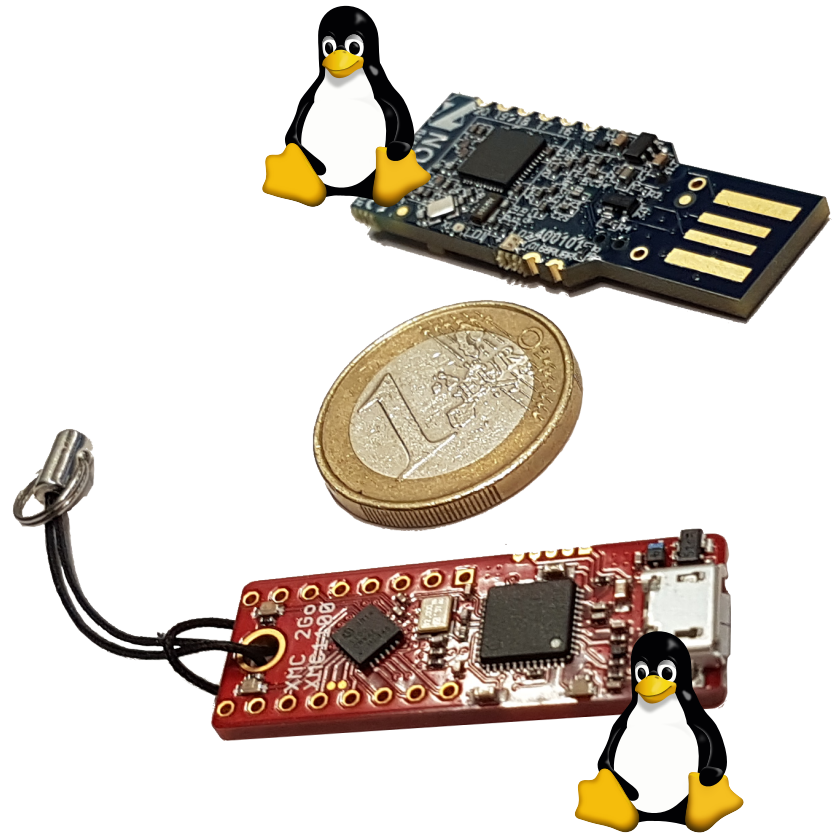


Hardware & Software

resource-constrained Things

- ROM: $\lesssim$ 250 KiB
- RAM: $\lesssim$ 50 KiB
- CPU: 72 MHz - 216 MHz
- battery-operated

Run Linux?



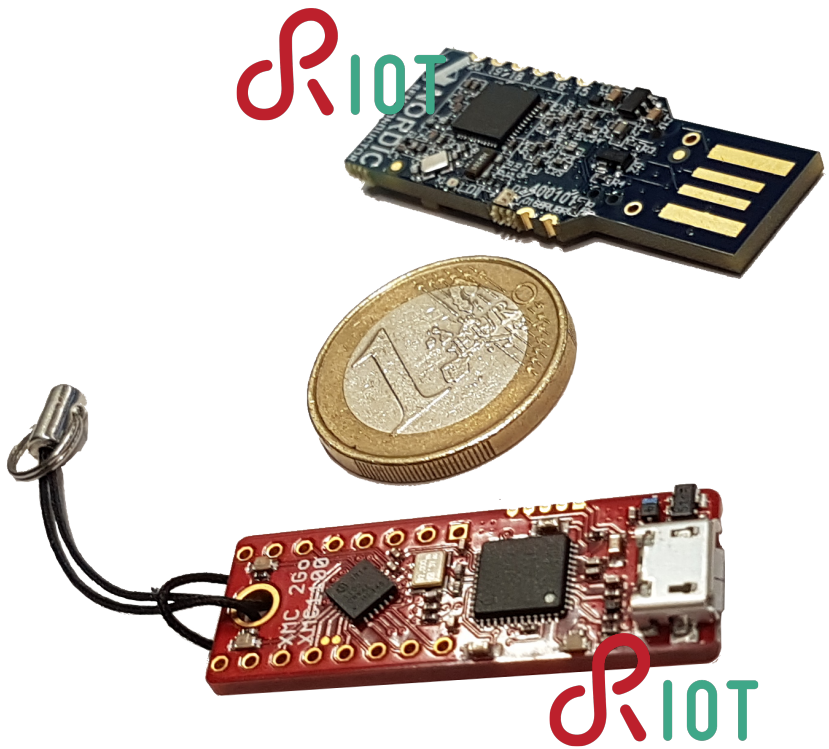
Hardware & Software

resource-constrained Things

- ROM: $\lesssim$ 250 KiB
- RAM: $\lesssim$ 50 KiB
- CPU: 72 MHz - 216 MHz
- battery-operated

~~Run Linux?~~

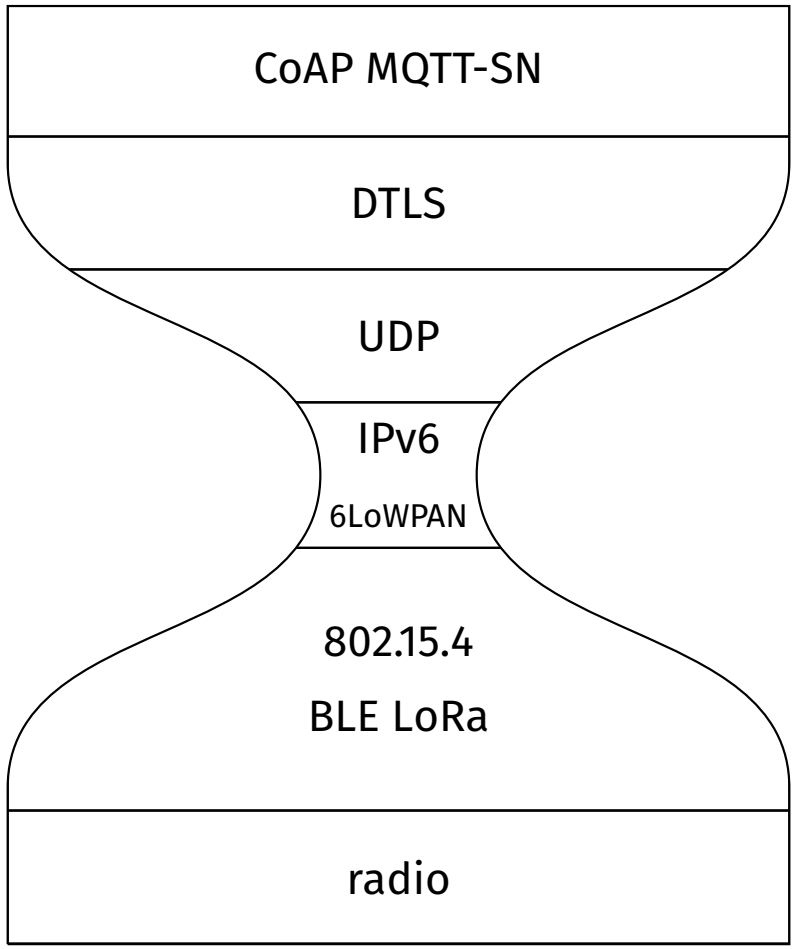
Run RIOT



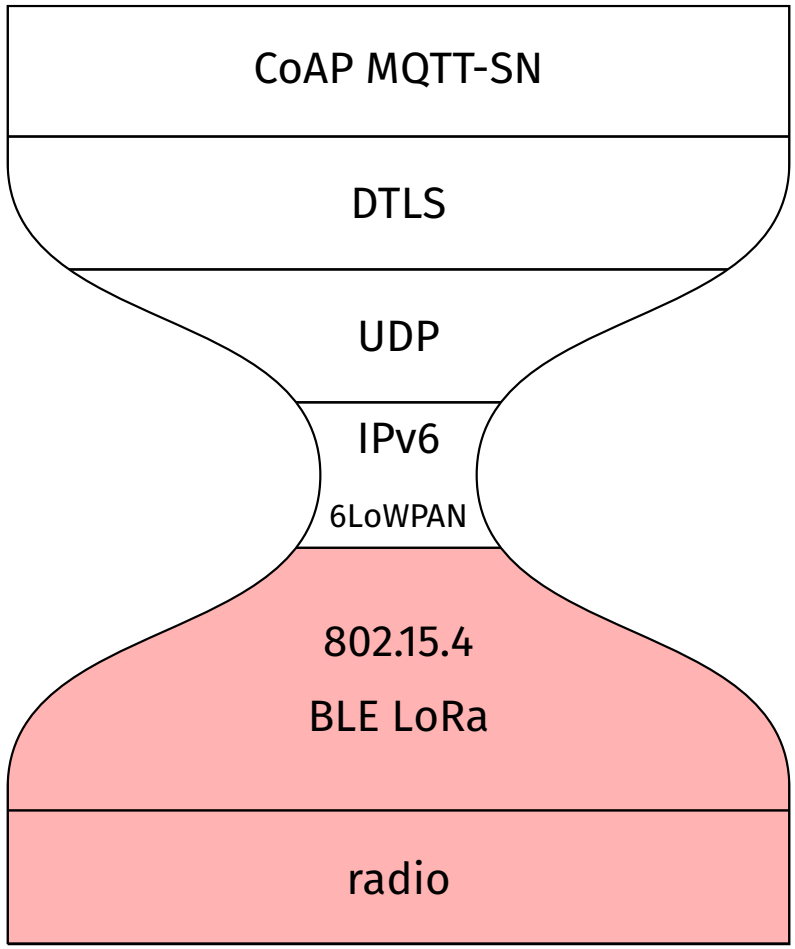
Interconnectivity in IoT

Kevin Ashton – 1999 – describes the **Internet of Things (IoT)** as a system where the physical world is **connected to the Internet** via **ubiquitous sensors**.

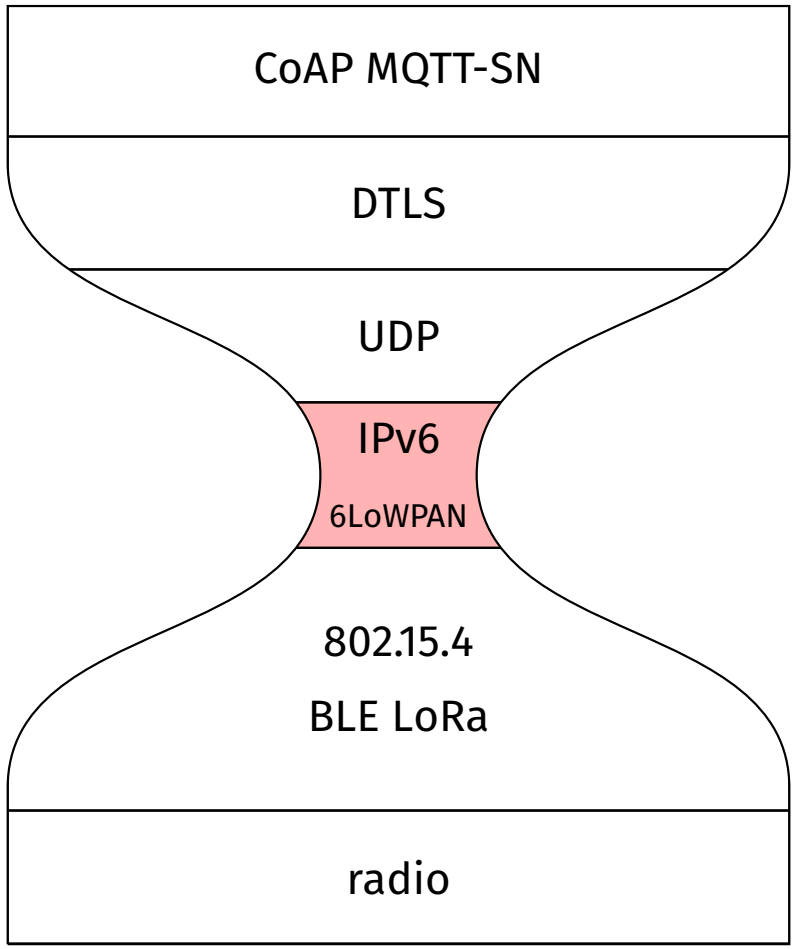
Interconnectivity in IoT



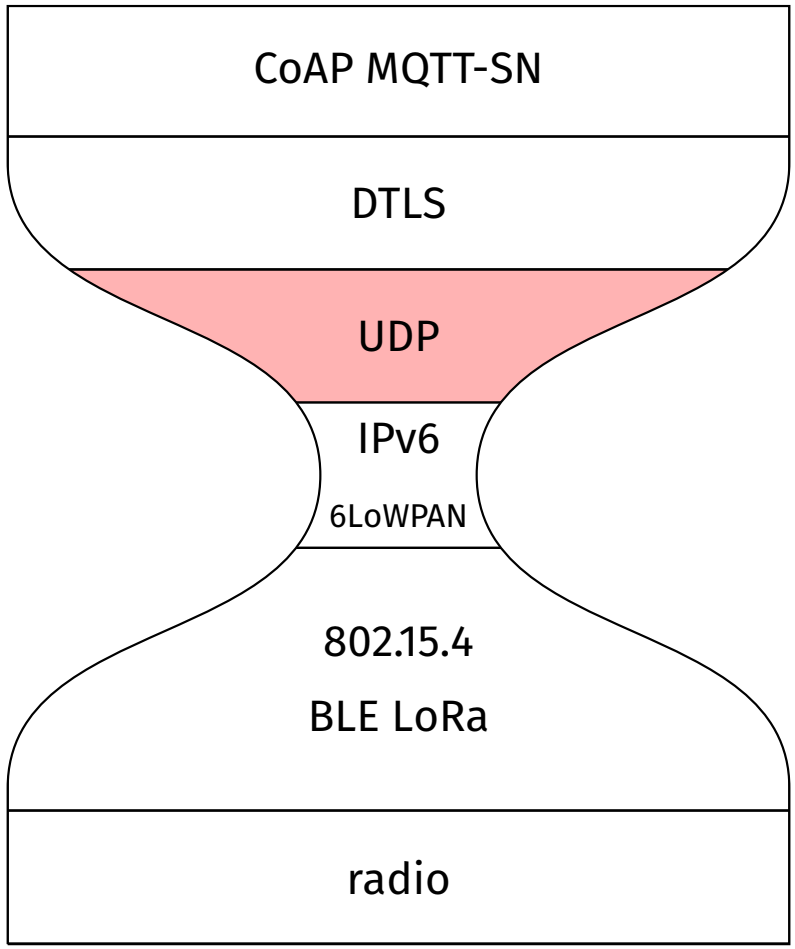
Interconnectivity in IoT



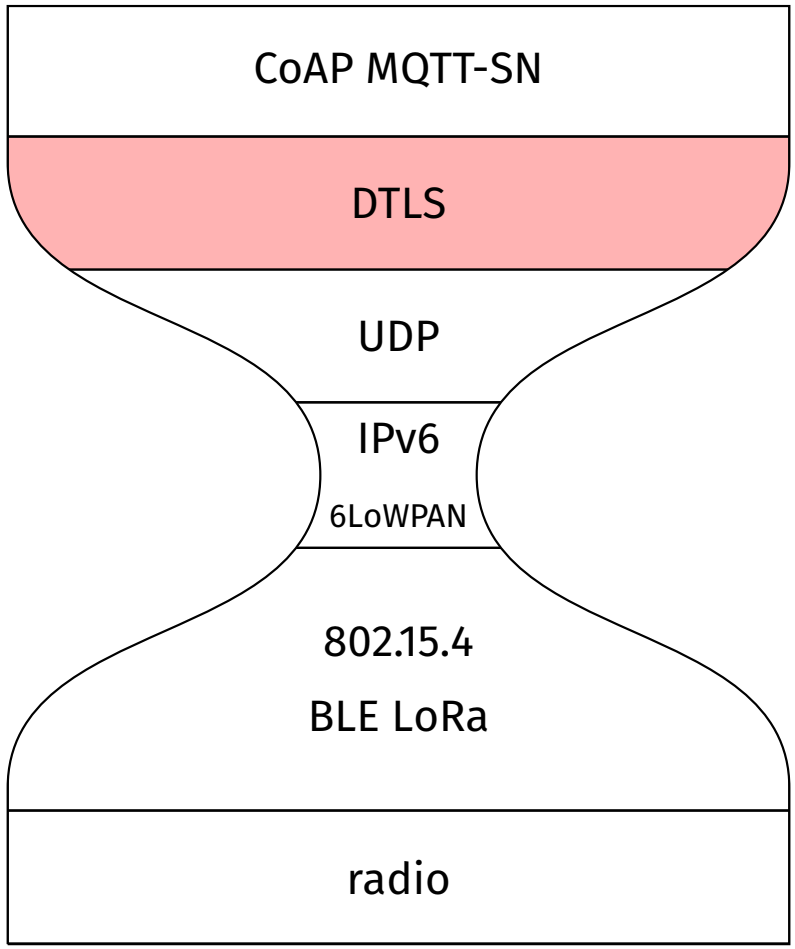
Interconnectivity in IoT



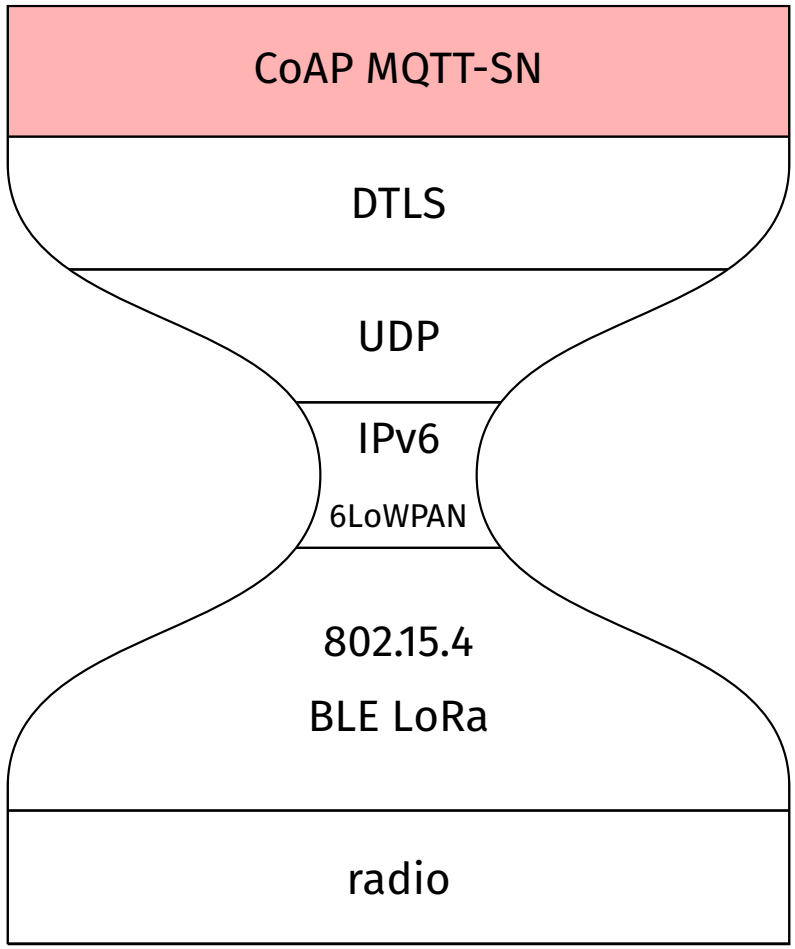
Interconnectivity in IoT



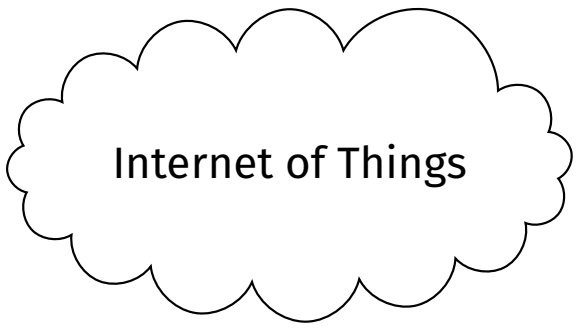
Interconnectivity in IoT



Interconnectivity in IoT



Applications



Agriculture Automation

People & Pet Monitoring

Building Management

Smart Sensing

Telemedicine & Healthcare

Smart Homes & Cities

Security & Surveillance

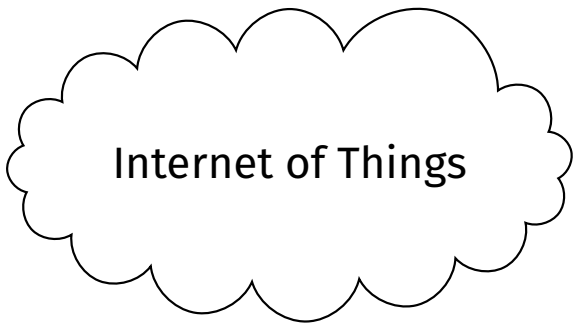
Industrial Automation

Applications

Agriculture Automation

People & Pet Monitoring

Building Management



Smart Sensing

Telemedicine & Healthcare

Smart Homes & Cities

Security & Surveillance

Industrial Automation

Industrial Automation

- Mission-/Safety-Critical Requirements
- Auto-Configuration & Self-Maintainability of Networks
- Support of Stationary & Mobile Devices
- Fault Tolerance
- Security
- Regulations

I3: Information-Centric Networks for the Industr. Internet

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Freie Universität Berlin

MSA
The Safety Company

I3
- Security
- Standardization
- Robustness

IIoT Scenarios

Scenario I: Data Retrieval

- semi-portable gas leak sensors
- portable sensors attached to workers
- mission protocols and logs

Scenario II: Alarm Propagation

- alarm notifications to nearby workers
- high priority traffic

Agenda

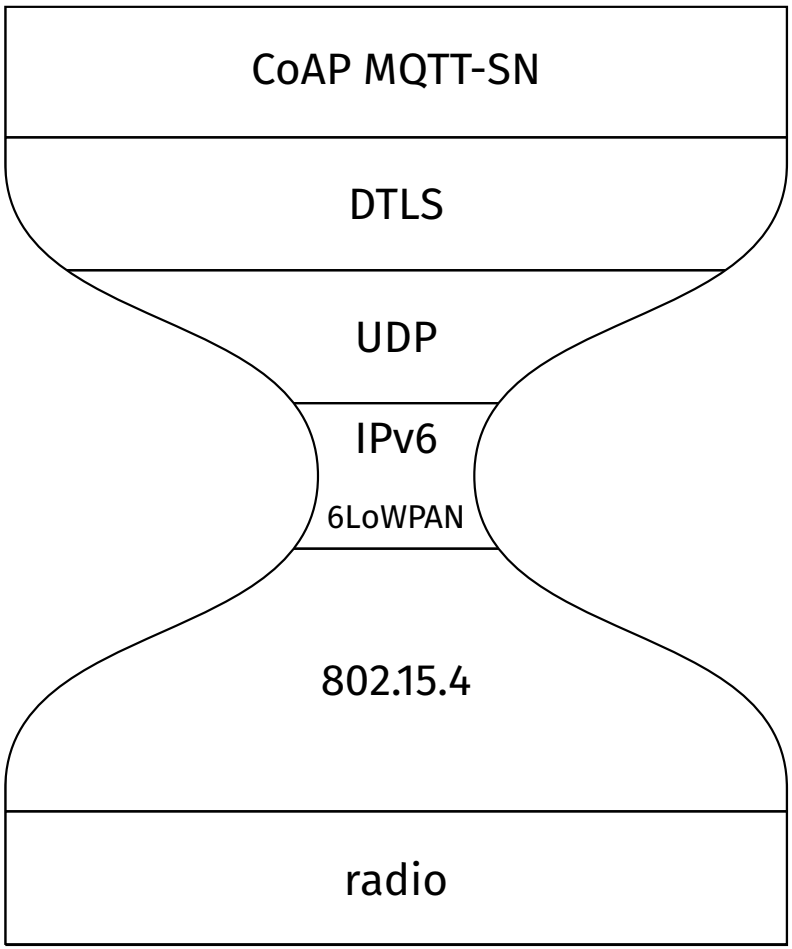
Industrial Internet of Things (IIoT)

IPv6 IoT Protocol Suite for the IIoT

Information-Centric Networking (ICN) for the IIoT

Publish — Subscribe in ICN/NDN

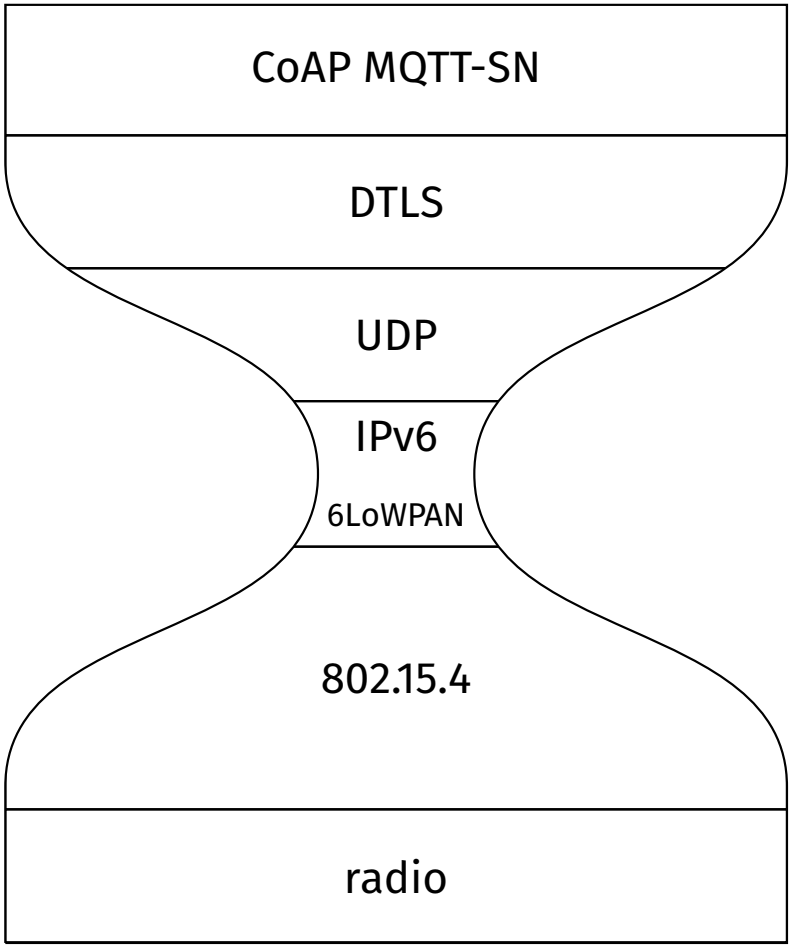
IPv6 IoT Protocol Overview



A name indicates what we seek.
An address indicates where it is.
A route indicates how we get there.

— Jon Postel (RFC 791, 1981)

IPv6 IoT Protocol Overview



A name indicates what we seek.
An address indicates where it is.
*A **route** indicates how we get there.*

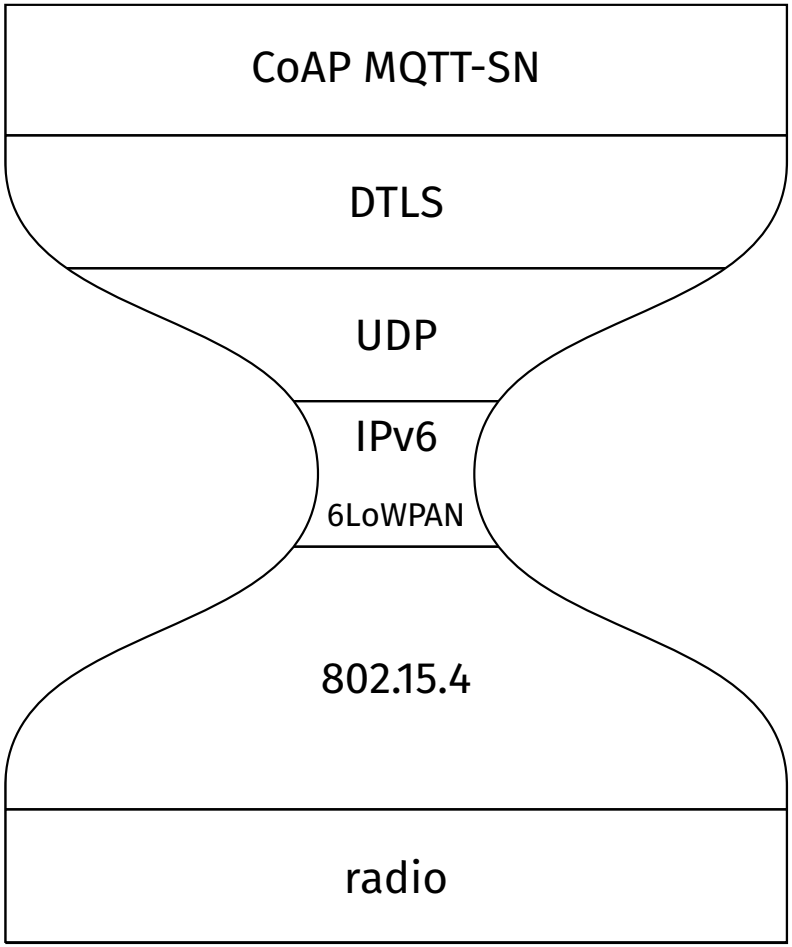
— Jon Postel (RFC 791, 1981)

Routing (IETF RoLL WG)

RPL Routing over LLNs

MPL Multicast Protocol for LLNs

IPv6 IoT Protocol Overview



A name indicates what we seek.
An address indicates where it is.
*A **route** indicates how we get there.*

— Jon Postel (RFC 791, 1981)

Routing (IETF RoLL WG)

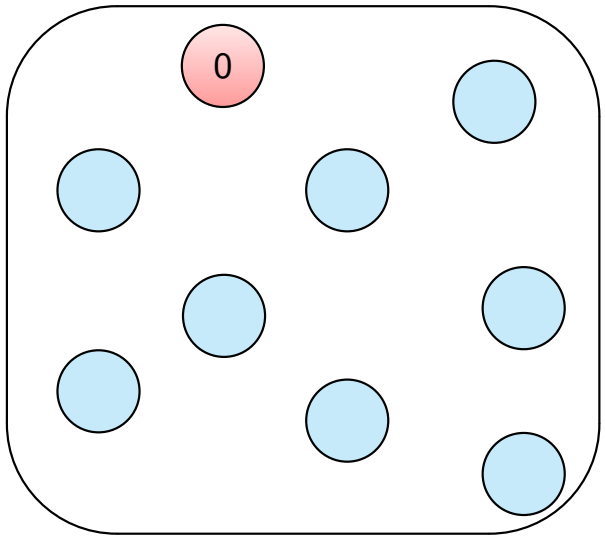
RPL Routing over LLNs

MPL Multicast Protocol for LLNs

Routing Protocol for LLNs (RPL)

Specification

- Lightweight Routing Protocol
- Dest. Oriented Directed Acyclic Graph (DODAG)
- ICMPv6 Control Messages
  - DODAG Information Object (DIO)
  - DODAG Information Solicitation (DIS)
  - Dest. Advert. Object (DAO & DAO-ACK)

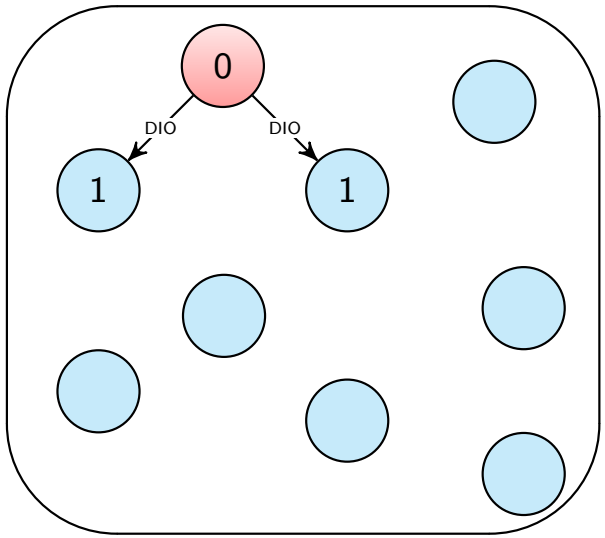


Formation of DODAG

Routing Protocol for LLNs (RPL)

Specification

- Lightweight Routing Protocol
- Dest. Oriented Directed Acyclic Graph (DODAG)
- ICMPv6 Control Messages
  - DODAG Information Object (DIO)
  - DODAG Information Solicitation (DIS)
  - Dest. Advert. Object (DAO & DAO-ACK)

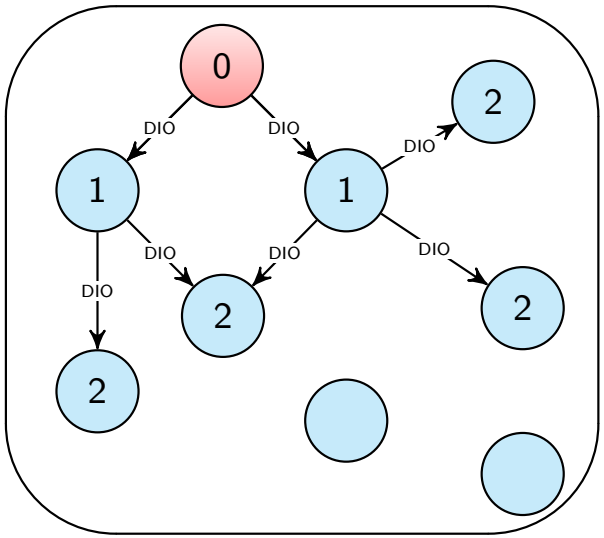


Root Emits DIOs

Routing Protocol for LLNs (RPL)

Specification

- Lightweight Routing Protocol
- Dest. Oriented Directed Acyclic Graph (DODAG)
- ICMPv6 Control Messages
  - DODAG Information Object (DIO)
  - DODAG Information Solicitation (DIS)
  - Dest. Advert. Object (DAO & DAO-ACK)

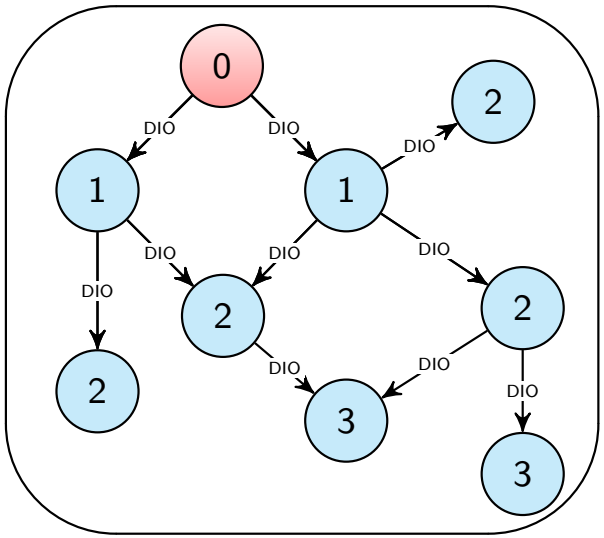


DIOs Flow Downward

Routing Protocol for LLNs (RPL)

Specification

- Lightweight Routing Protocol
- Dest. Oriented Directed Acyclic Graph (DODAG)
- ICMPv6 Control Messages
  - DODAG Information Object (DIO)
  - DODAG Information Solicitation (DIS)
  - Dest. Advert. Object (DAO & DAO-ACK)

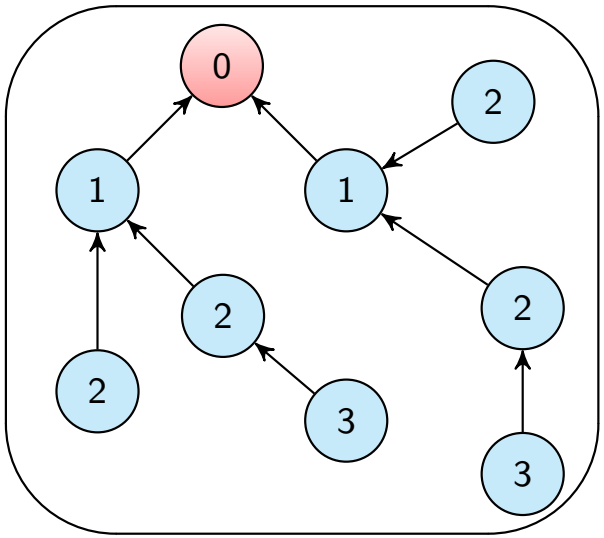


All Nodes Participate

Routing Protocol for LLNs (RPL)

Specification

- Lightweight Routing Protocol
- Dest. Oriented Directed Acyclic Graph (DODAG)
- ICMPv6 Control Messages
  - DODAG Information Object (DIO)
  - DODAG Information Solicitation (DIS)
  - Dest. Advert. Object (DAO & DAO-ACK)

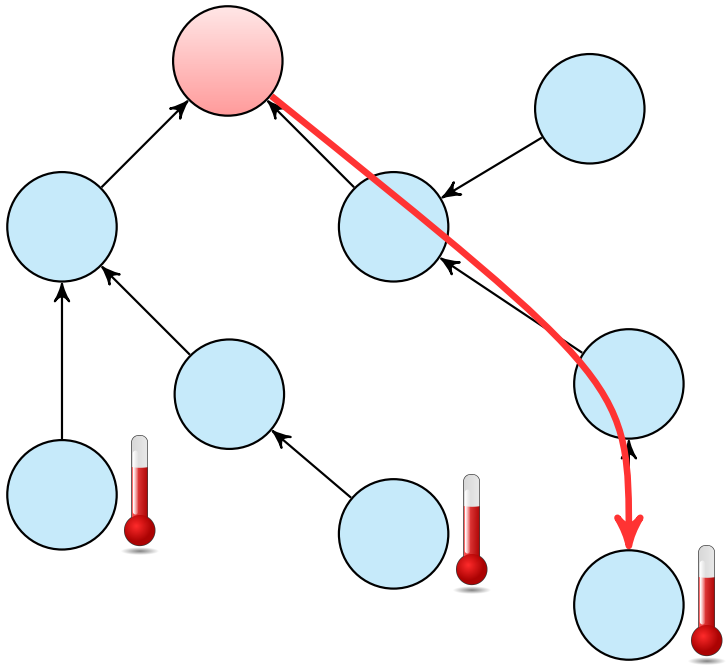


Converged DODAG

Scenario I: Data Retrieval

Unicast

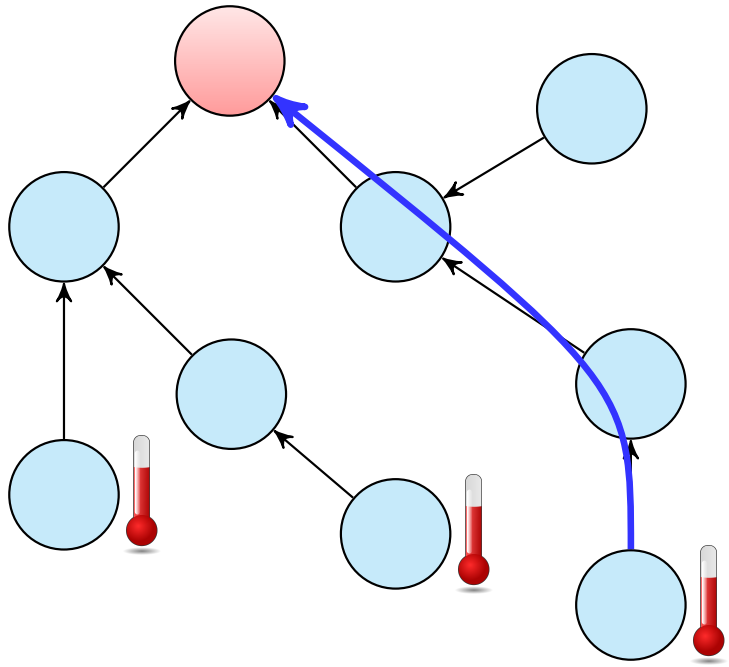
```
GET coap://[2001::DEAD:BEEF:CAFE:1]/temp
```



Scenario I: Data Retrieval

Unicast

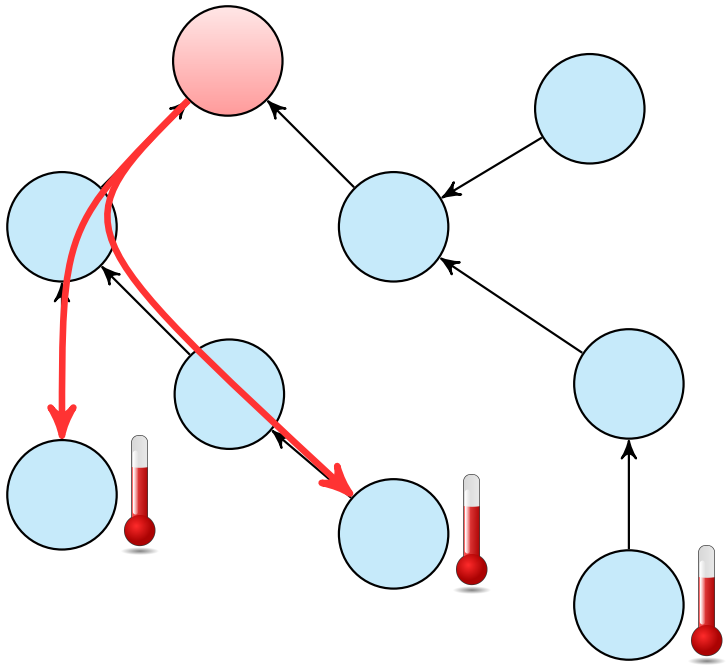
```
GET coap://[2001::DEAD:BEEF:CAFE:1]/temp
```



Scenario I: Data Retrieval

Multicast

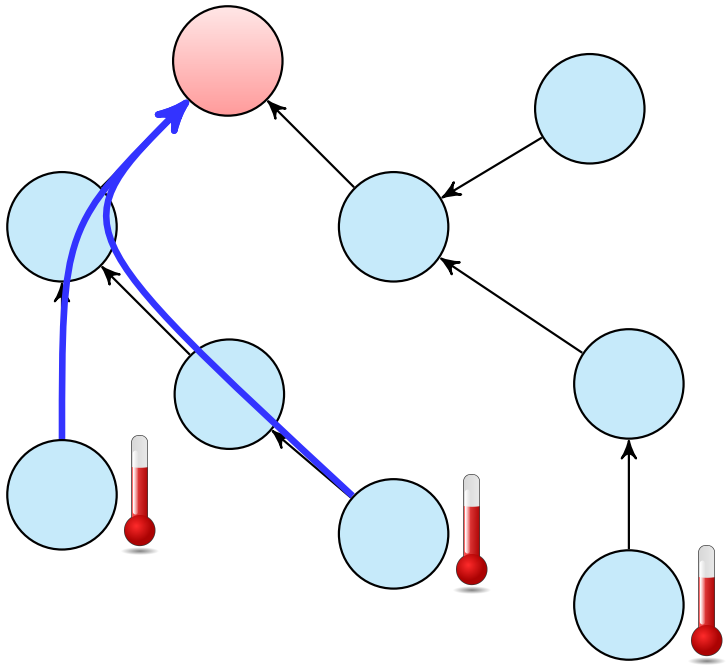
```
GET coap://[FF04::DEAD:BEEF:CAFE:1]/temp
```



Scenario I: Data Retrieval

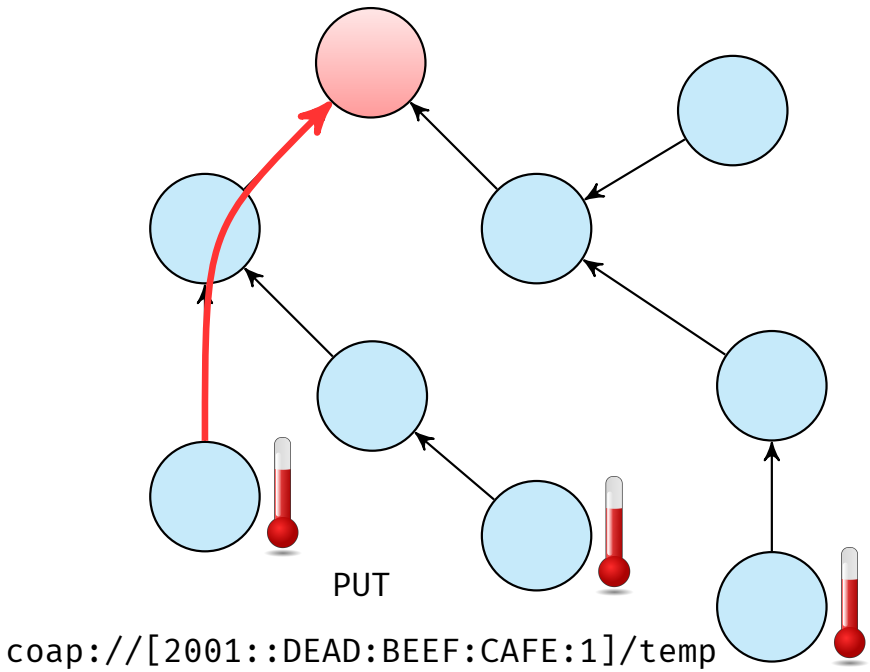
Multicast

```
GET coap://[FF04::DEAD:BEEF:CAFE:1]/temp
```



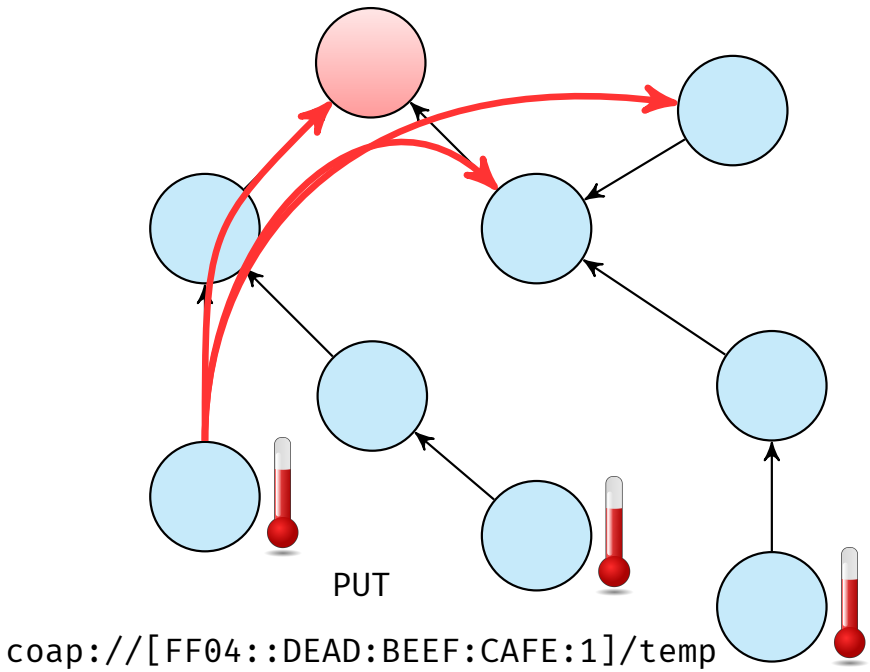
Scenario II: Alarm Propagation

Unicast



Scenario II: Alarm Propagation

Multicast



Agenda

Industrial Internet of Things (IIoT)

IPv6 IoT Protocol Suite for the IIoT

Information-Centric Networking (ICN) for the IIoT

Publish — Subscribe in ICN/NDN

Receiver Mobility

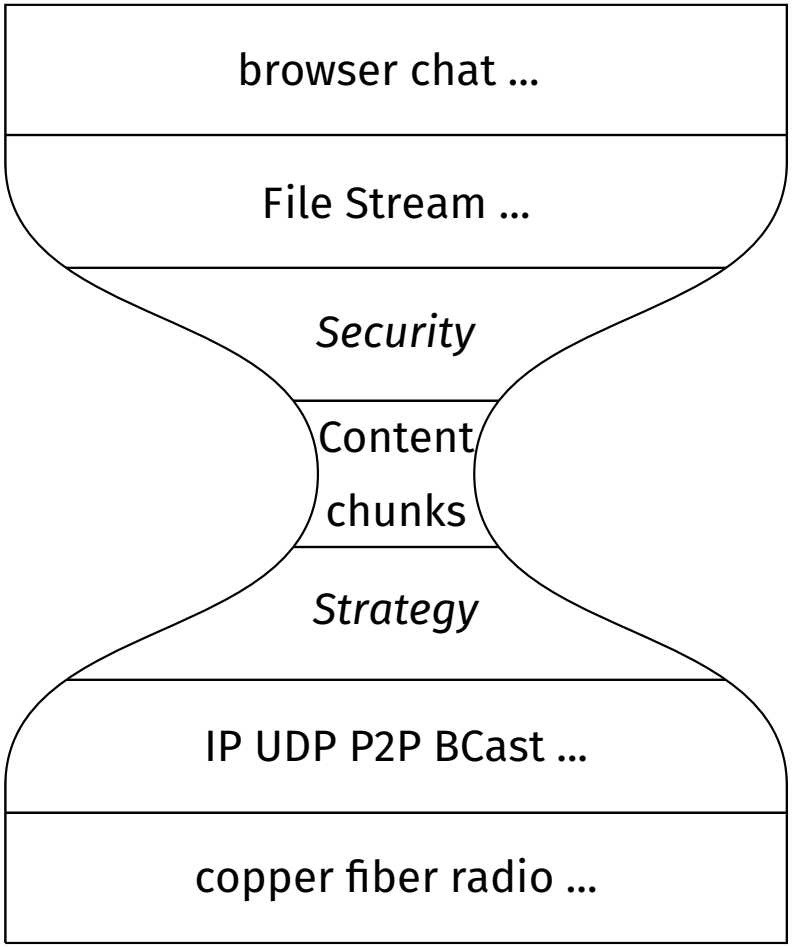
Security at Data Level

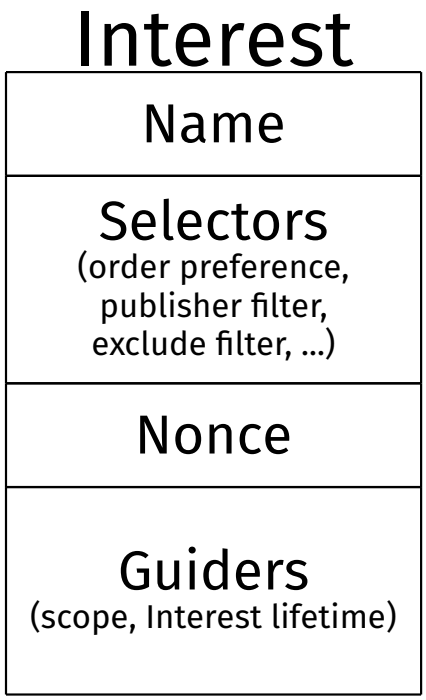
Motivation for ICN in IIoT

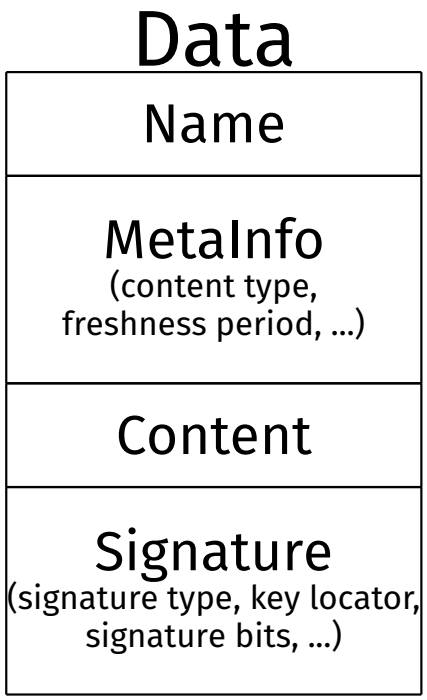
Network Caches

Smaller Memory Footprint

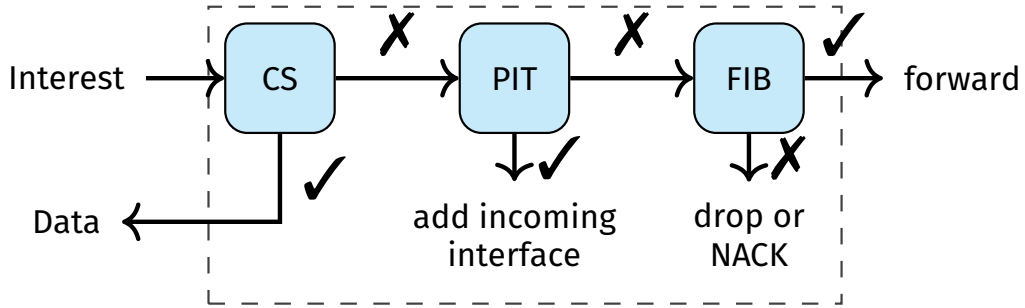
Named Data Networking (NDN)

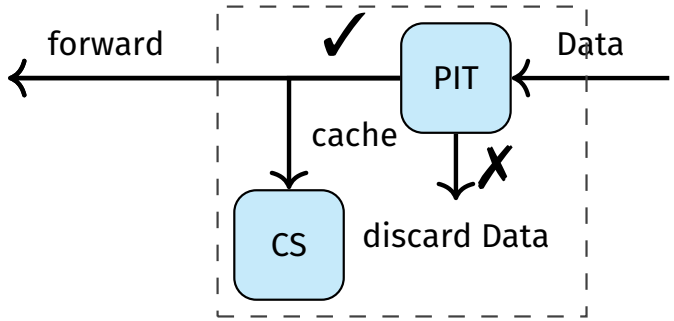




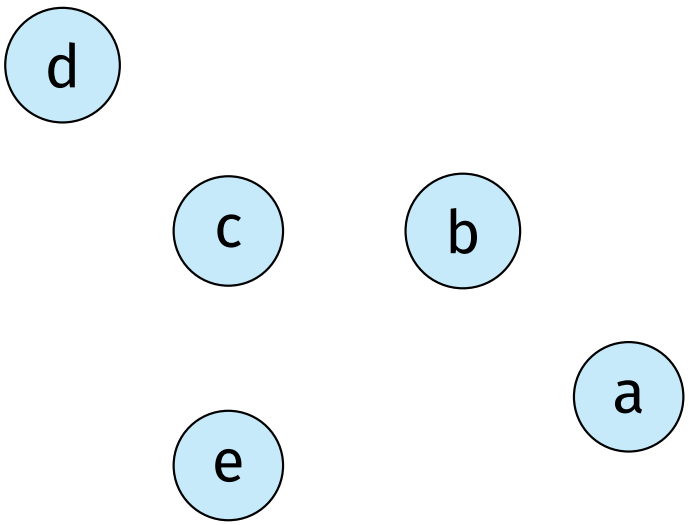


NDN Architecture



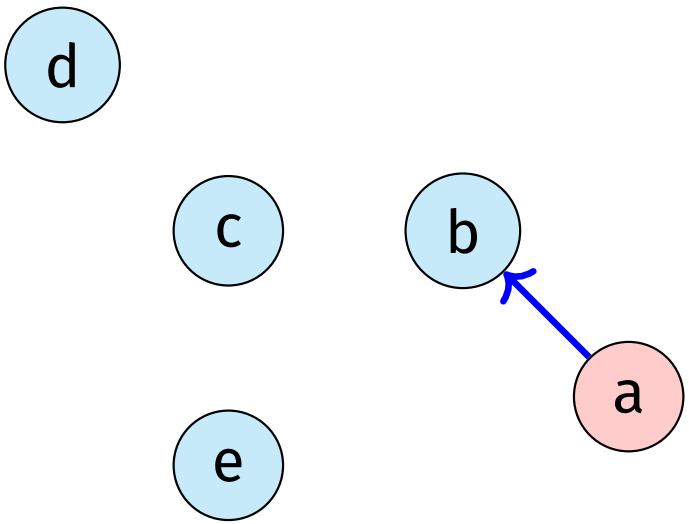


NDN Operation: Interest



a is interested in */haw/temp1*

NDN Operation: Interest



Content Store (CS)

Name	Data
...	...

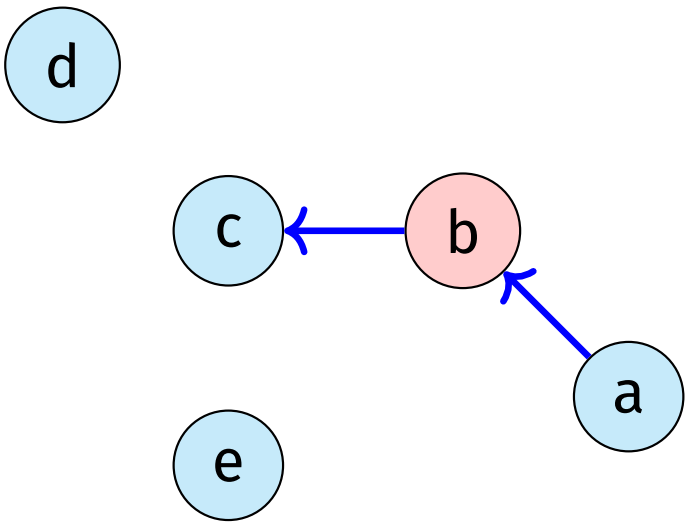
Pending Interest Table (PIT)

Prefix	Face List
/haw/temp1	27

Forwarding Information Base (FIB)

Prefix	Face List
/haw/temp1	b

NDN Operation: Interest



Content Store (CS)

Name	Data
...	...

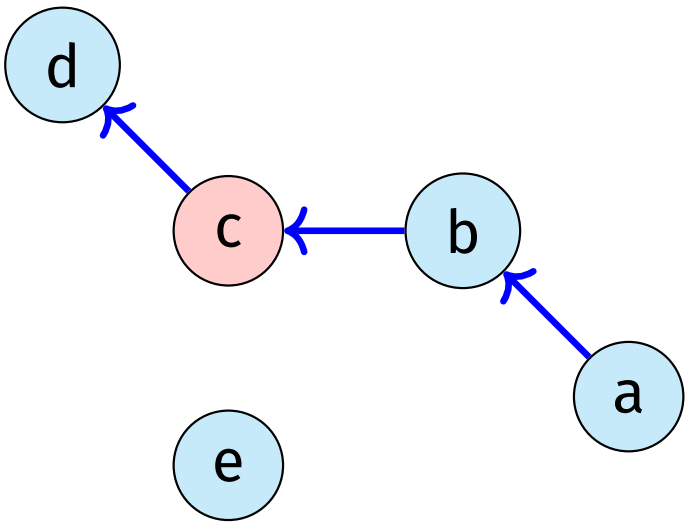
Pending Interest Table (PIT)

Prefix	Face List
/haw/temp1	a

Forwarding Information Base (FIB)

Prefix	Face List
/haw/temp1	c

NDN Operation: Interest



Content Store (CS)

Name	Data
...	...

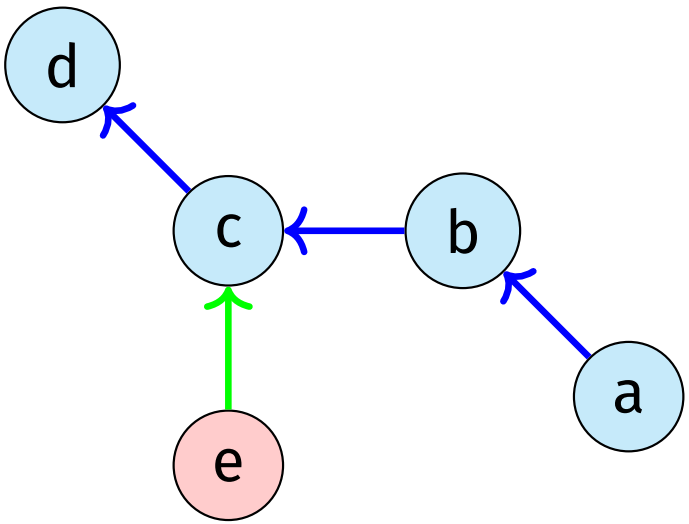
Pending Interest Table (PIT)

Prefix	Face List
/haw/temp1	b

Forwarding Information Base (FIB)

Prefix	Face List
/haw/temp1	d

NDN Operation: Interest



Content Store (CS)

Name	Data
...	...

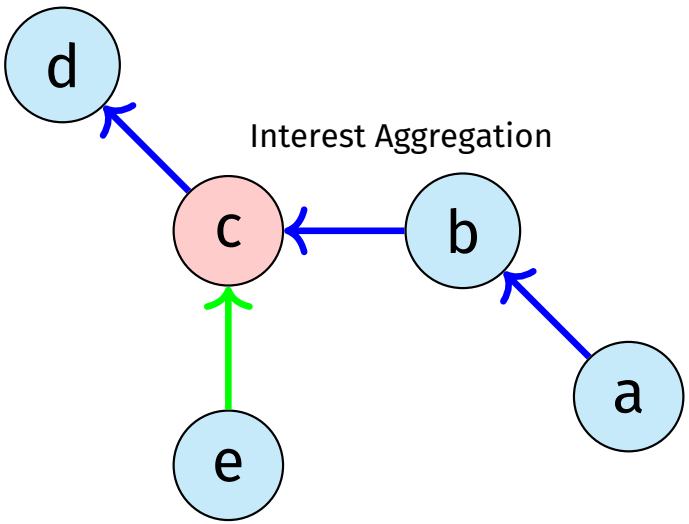
Pending Interest Table (PIT)

Prefix	Face List
/haw/temp1	32

Forwarding Information Base (FIB)

Prefix	Face List
/haw/temp1	c

NDN Operation: Interest



Content Store (CS)

Name	Data
...	...

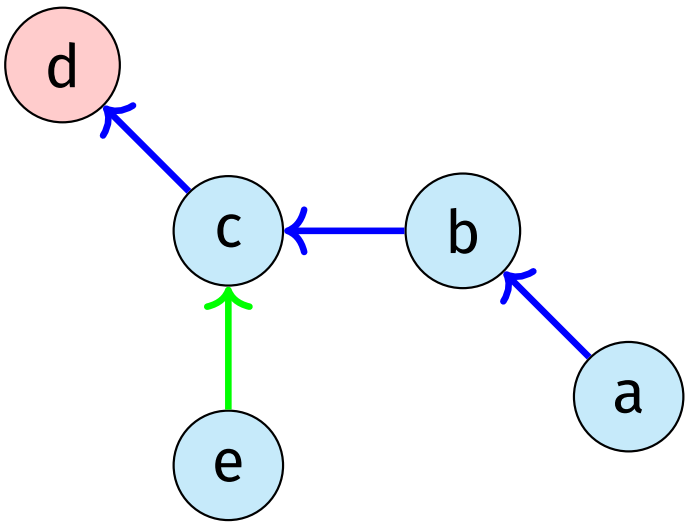
Pending Interest Table (PIT)

Prefix	Face List
/haw/temp1	b, e

Forwarding Information Base (FIB)

Prefix	Face List
/haw/temp1	d

NDN Operation: Interest



Content Store (CS)

Name	Data
/haw/temp1	17 °C

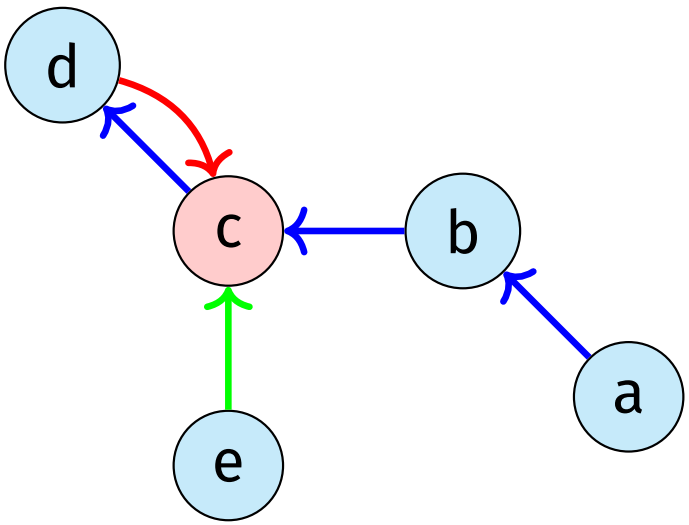
Pending Interest Table (PIT)

Prefix	Face List
...	...

Forwarding Information Base (FIB)

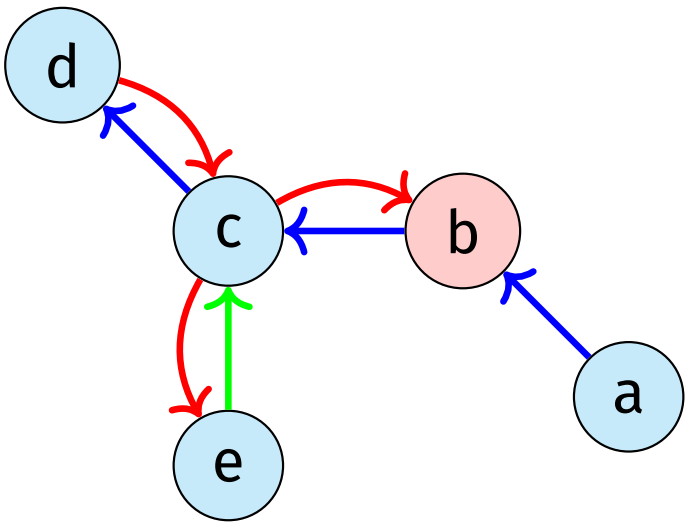
Prefix	Face List
...	...

NDN Operation: Data



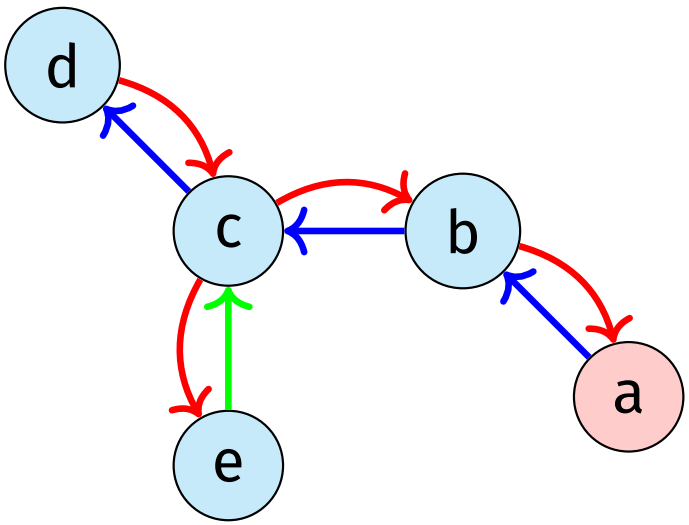
Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	b, e

NDN Operation: Data



Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	a

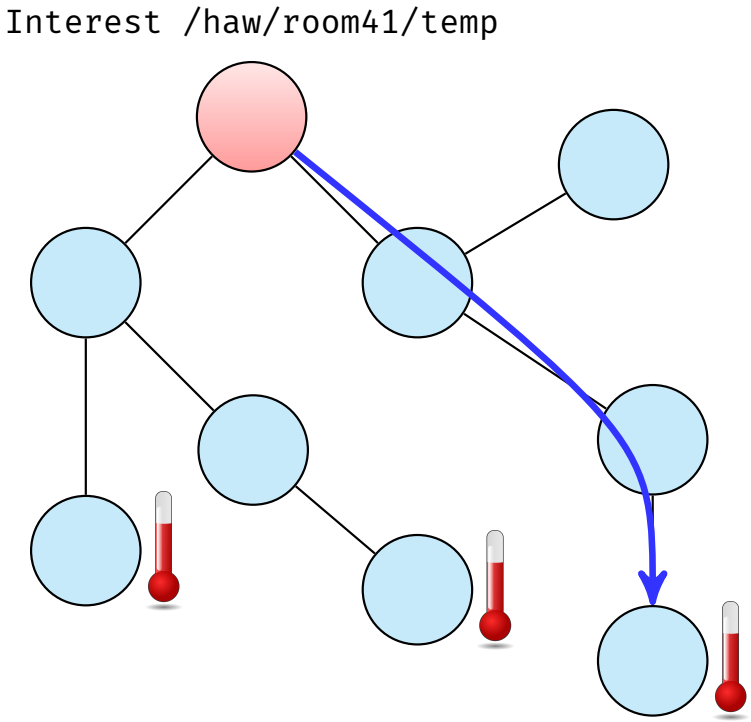
NDN Operation: Data



Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	27

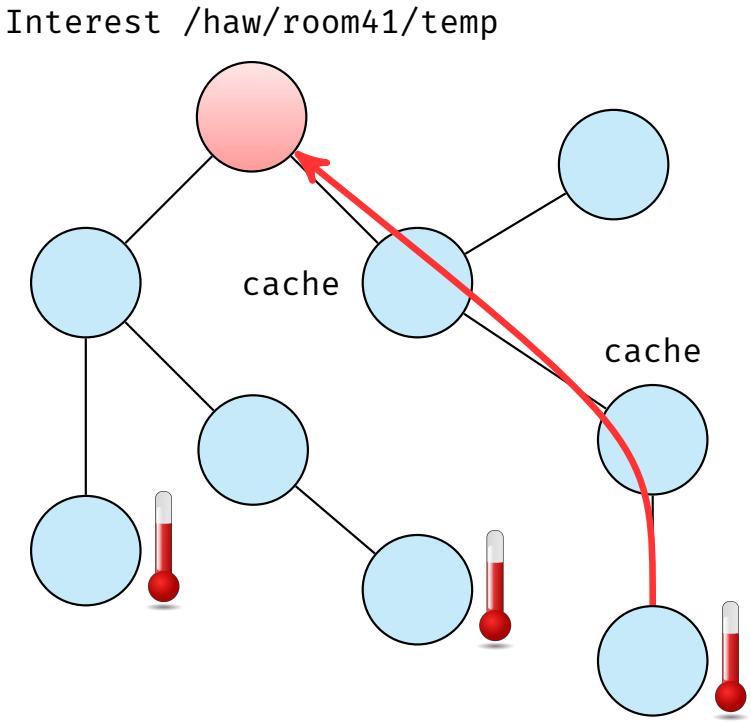
Scenario I: Data Retrieval

Single-Producer



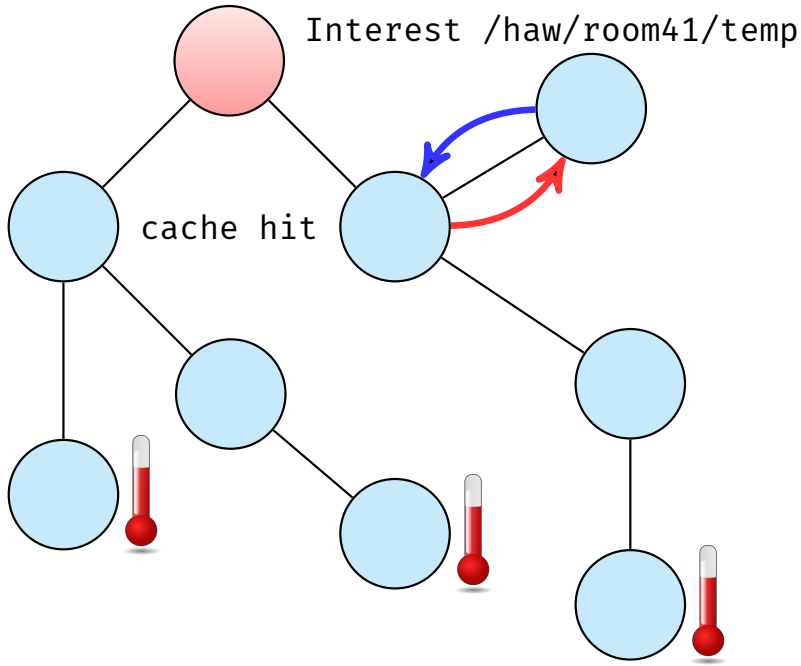
Scenario I: Data Retrieval

Single-Producer



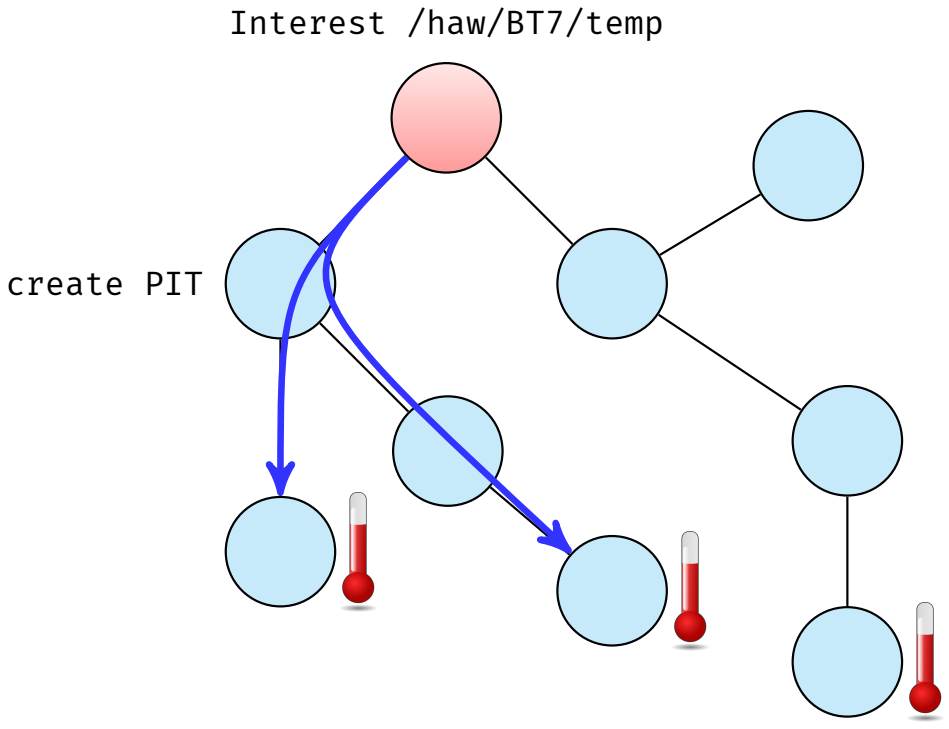
Scenario I: Data Retrieval

Single-Producer



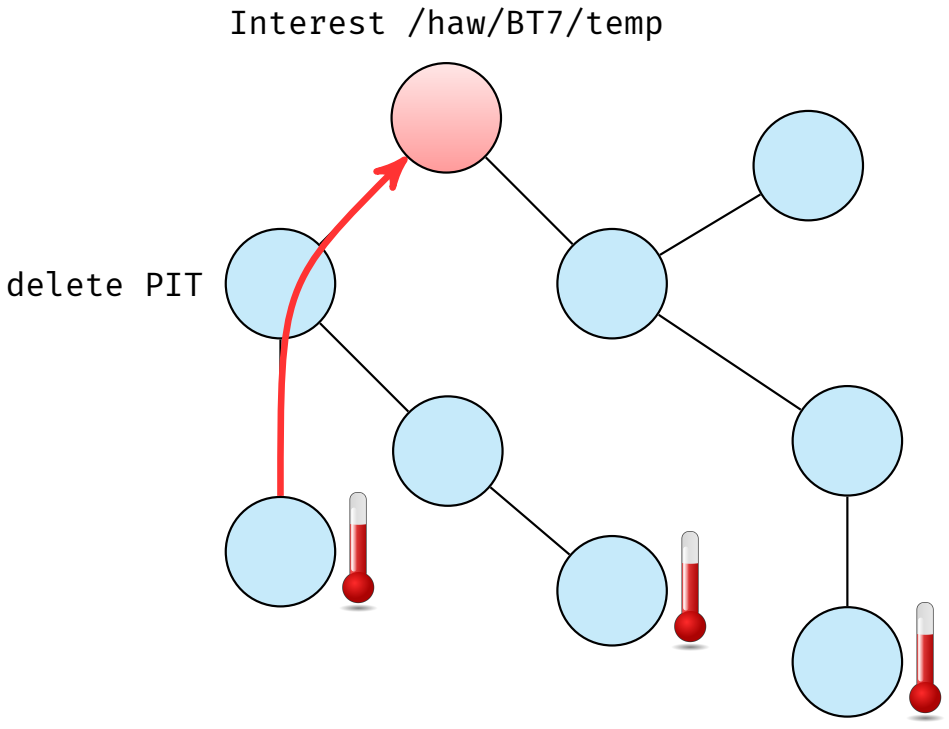
Scenario I: Data Retrieval

Multi-Producer



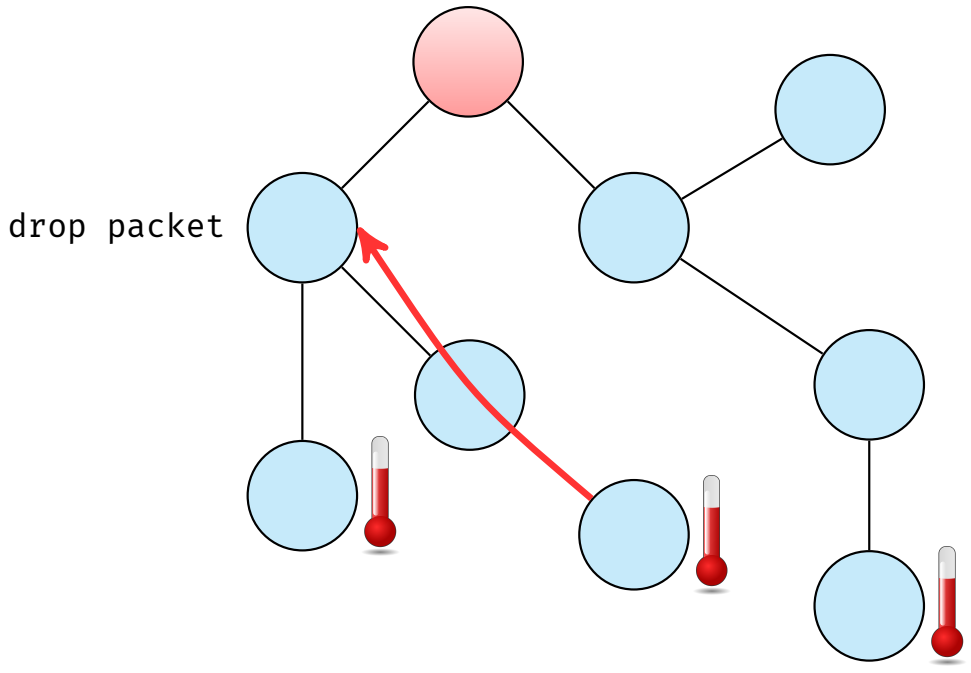
Scenario I: Data Retrieval

Multi-Producer



Scenario I: Data Retrieval

Multi-Producer



Shortcomings for IIoT

Scenario I: Data Retrieval

- No Propagation from N Producers to 1 Consumer

Scenario II: Alarm Propagation

- No *PUSH* support
- No mechanism to deliver unsolicited data

NDN IoT vs IP IoT

IP IoT

- *Pull* / *Push* in application (CoAP)
- *Fragm. & Header Compr.* for higher goodput
- End-to-End retrans. with lossy links
- End-to-End security on transoprt layer

NDN IoT

- *Pull* in network layer / no *Push*
- Less protocol overhead for application reqs.
- Re-Interest may benefit from cache hits
- Security on chunk level (beneficial for caching)

Agenda

Industrial Internet of Things (IIoT)

IPv6 IoT Protocol Suite for the IIoT

Information-Centric Networking (ICN) for the IIoT

Publish — Subscribe in ICN/NDN

Publish – Subscribe Systems

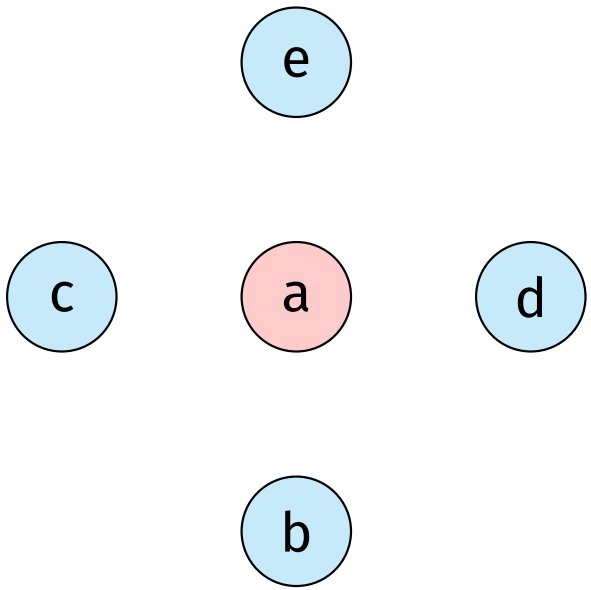
- Loose coupling (locality, temporal, synchronicity)
- Higher scalability
- N to M communication

Taxonomy

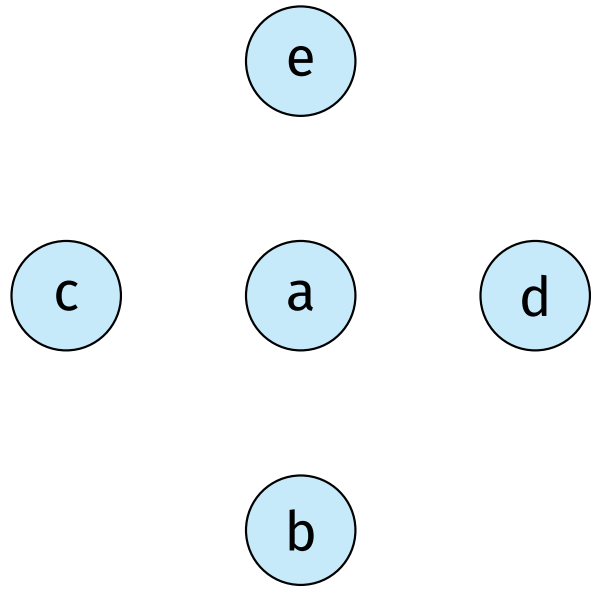
- Subscriber indicates interest in content
- Publisher produces content
- Pub-/Sub-System handles
  - update notifications
  - delivery of content from publisher to subscriber
  - message caching
  - QoS

Routing Strategies

Selective Event Routing
(Rendezvous-Based)

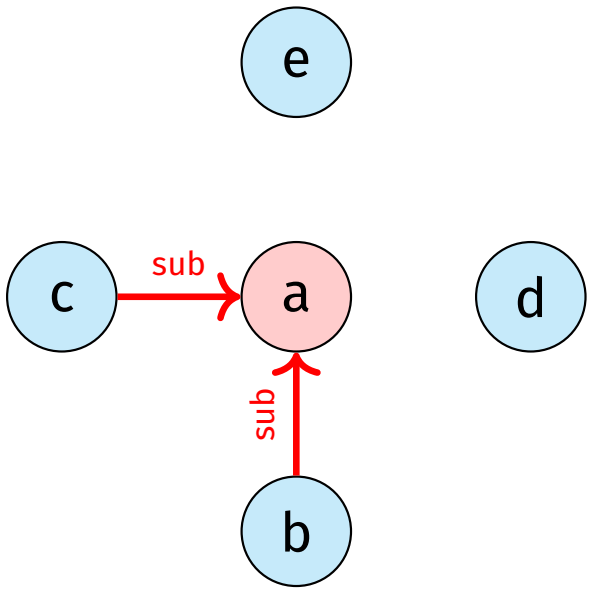


Subscription Flooding

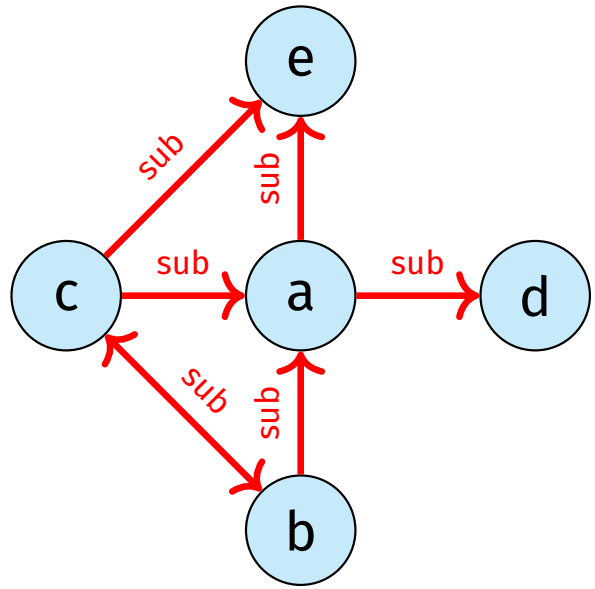


Subscriptions

Selective Event Routing
(Rendezvous-Based)

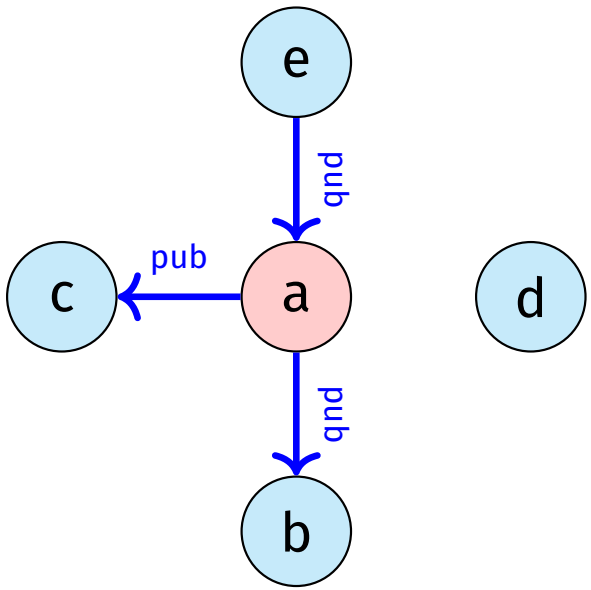


Subscription Flooding

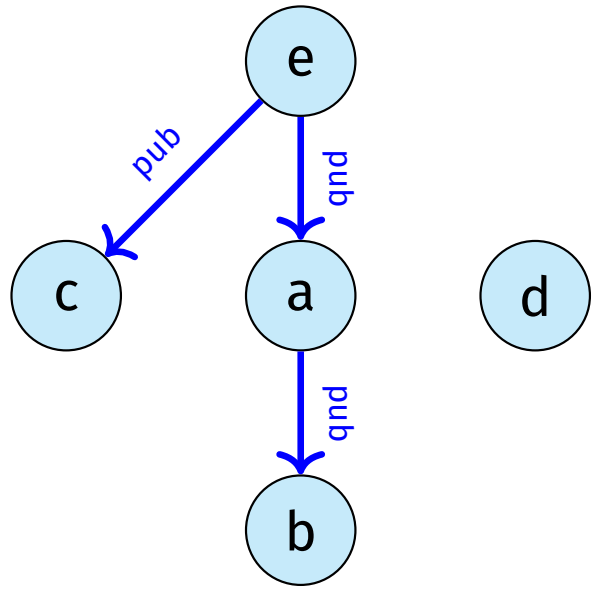


Publish

Selective Event Routing
(Rendezvous-Based)



Subscription Flooding

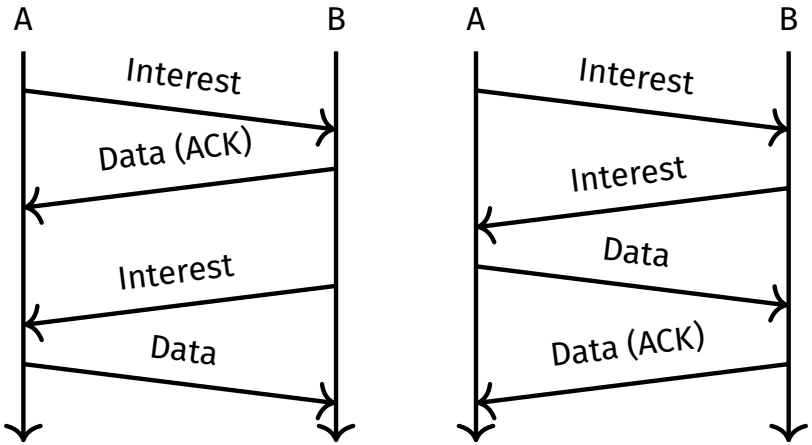


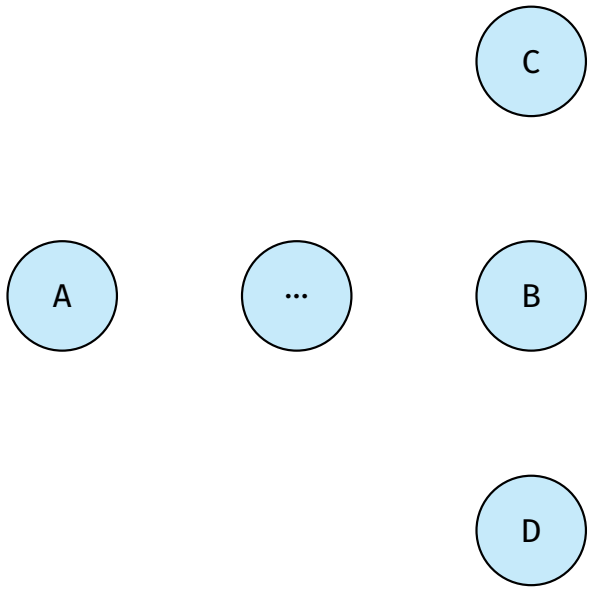
How to Build a Publisher in NDN?

No Support for Unsolicited Data in NDN
Proposals:

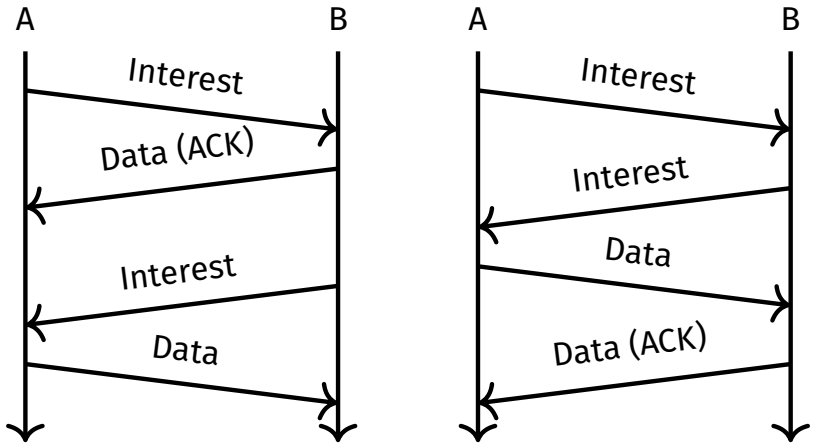
- Interest Polling
- Interest Notification
- Long-Lived Interest
- *PUSH* Message

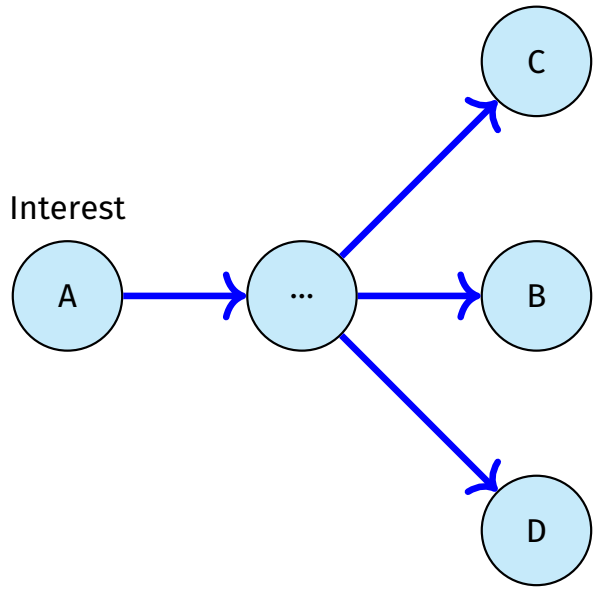
Interest Polling



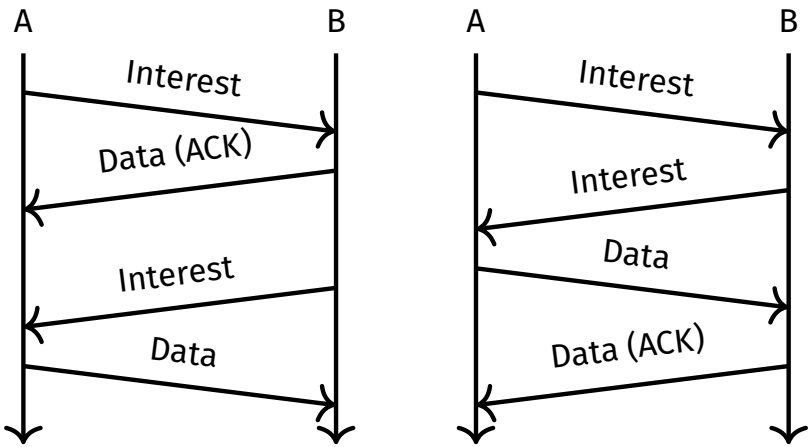


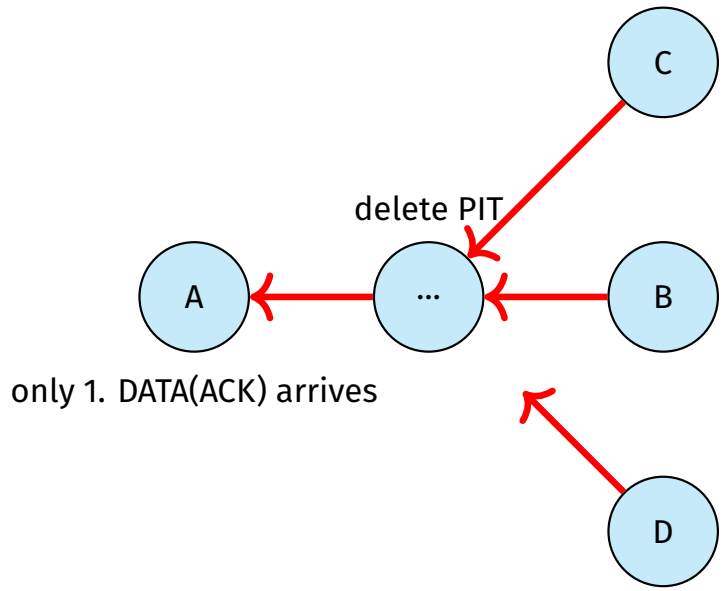
Interest Polling



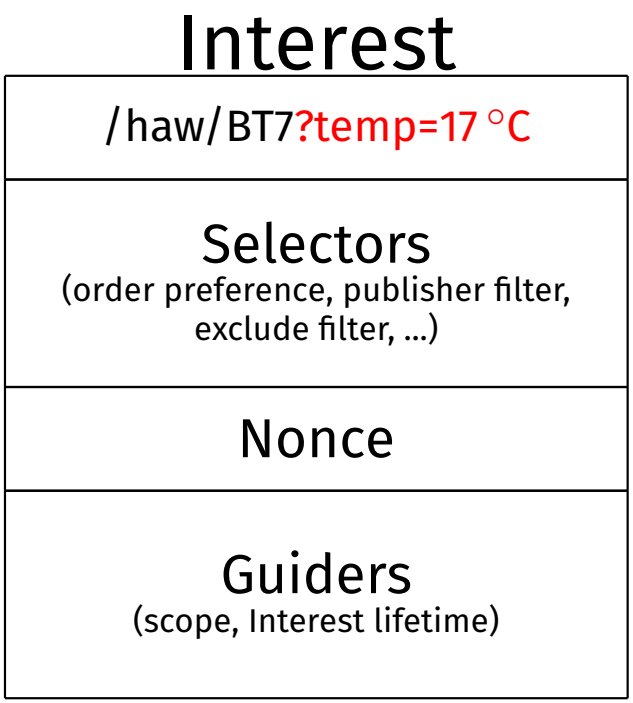


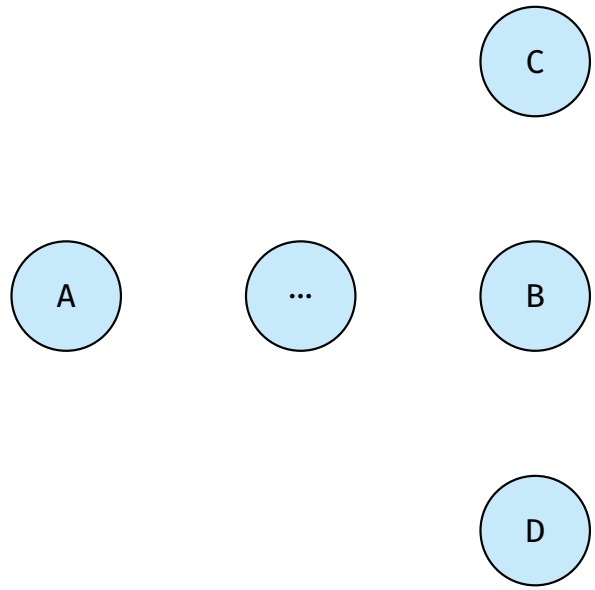
Interest Polling



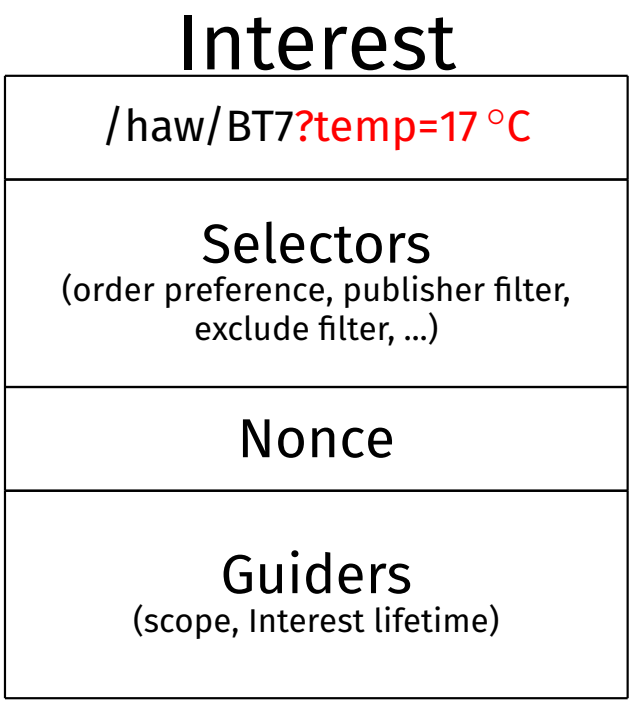


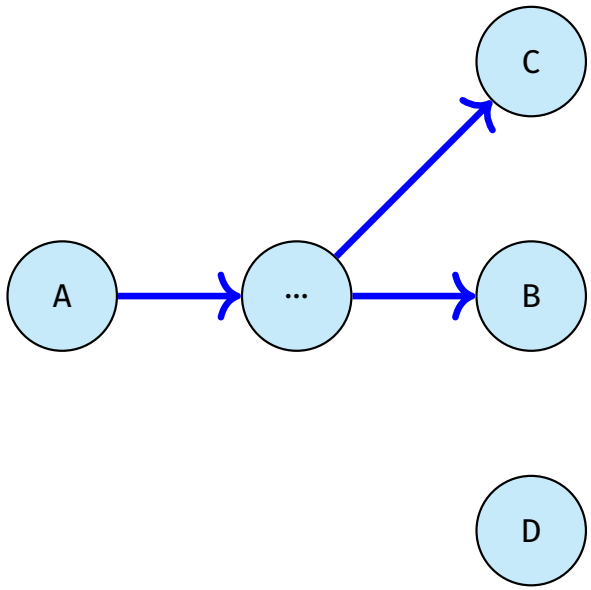
Interest Notification



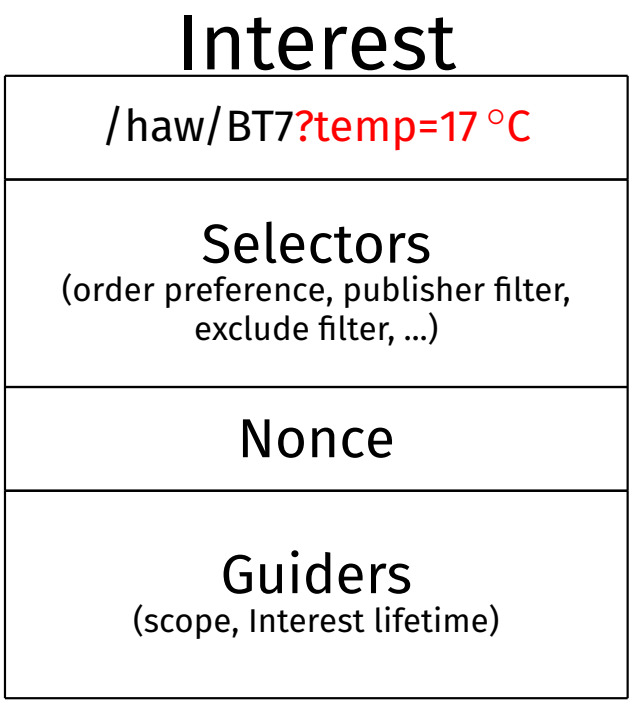


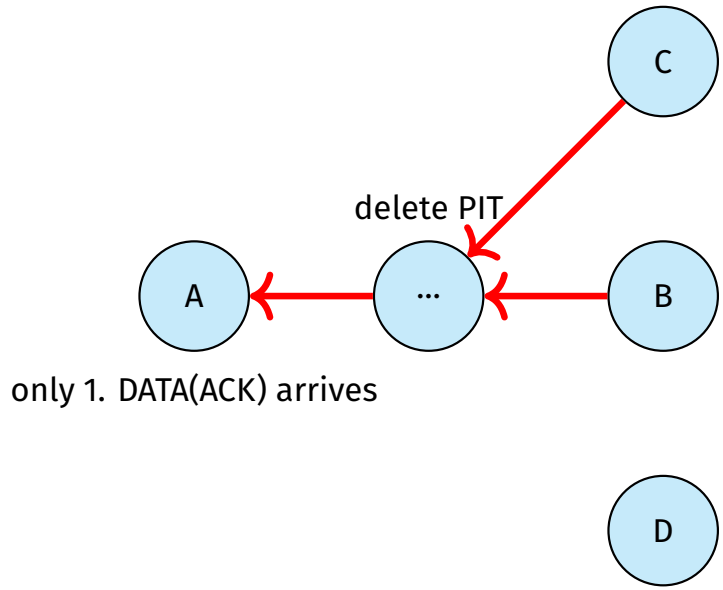
Interest Notification



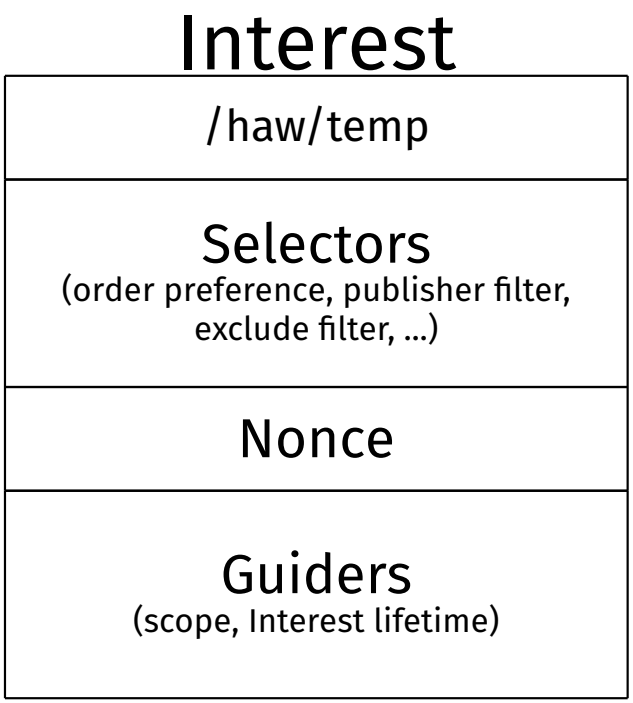


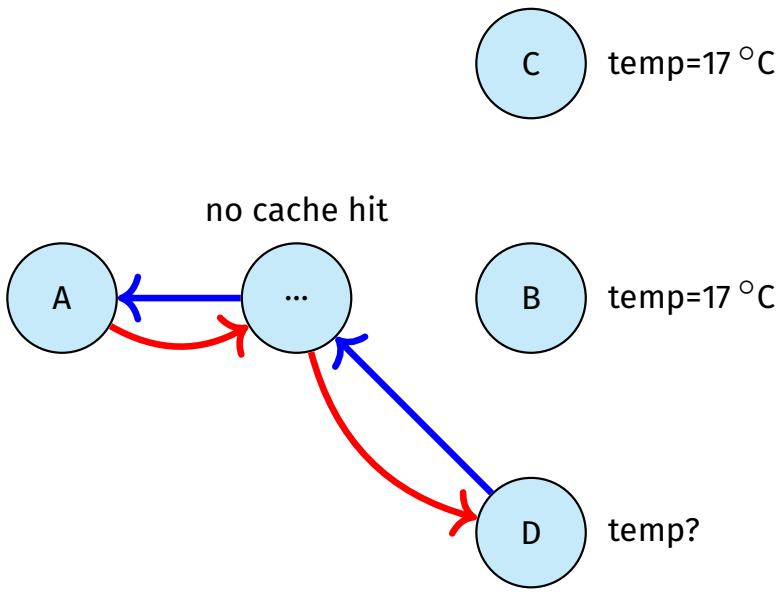
Interest Notification





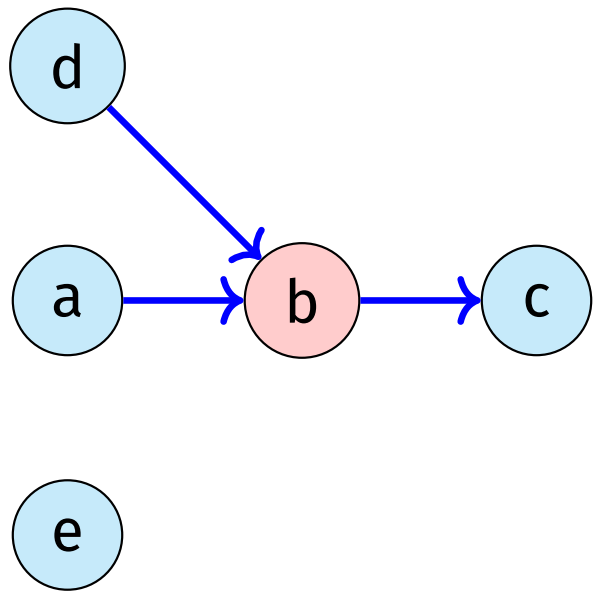
Interest Notification





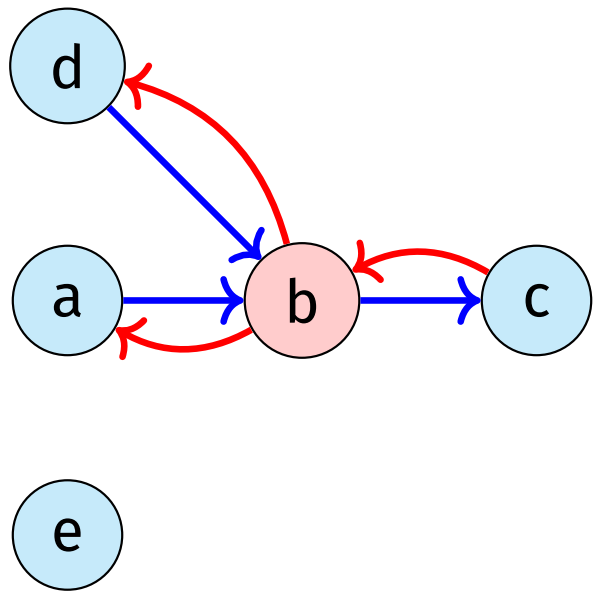
Long-Lived Interest

Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	a, d



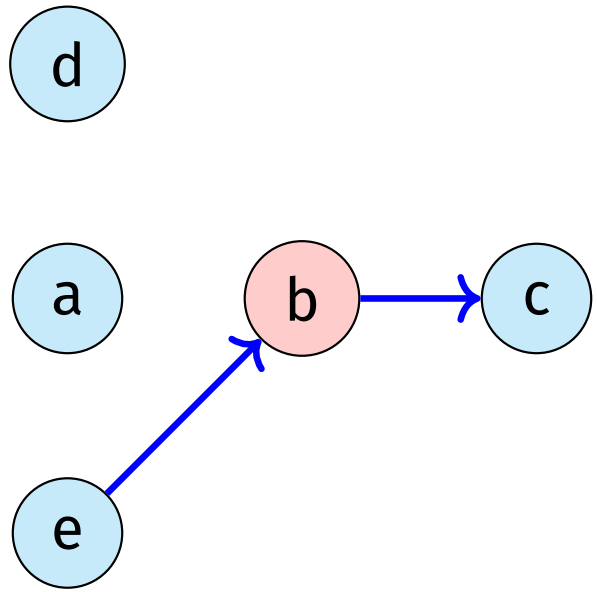
Long-Lived Interest

Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	a, d



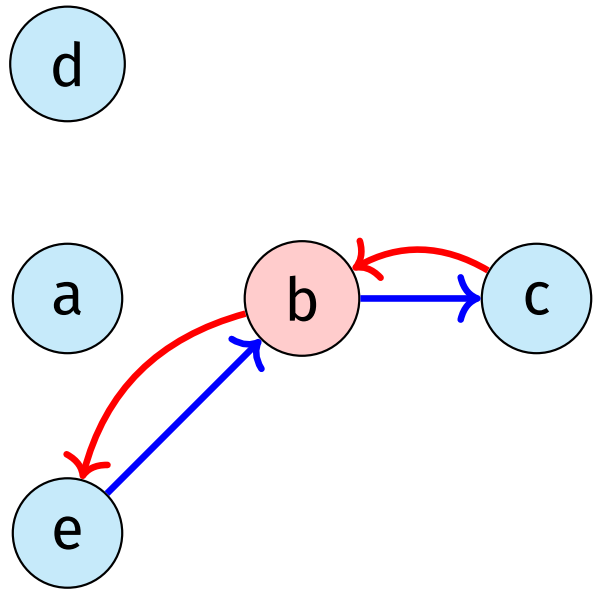
Long-Lived Interest

Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	a, d
/haw/temp2	e



Long-Lived Interest

Pending Interest Table (PIT)	
Prefix	Face List
/haw/temp1	a, d
/haw/temp2	e



PUSH Message

Characteristics

- New packet format

Shortcomings

- Needs flow control handling
- Popularity-based cache placement strategies are difficult to realize

Current Ideas and Decisions

- *publish* translates to *push* and *content replication*
- content is cached off-path
- *content replication* eases routing efforts for subscribers
  - more replications $\Rightarrow$ less routing states
  - more replications $\Rightarrow$ higher QoS
- default routes from publisher to content caches reduce routing state at publisher
- PANINI can provide prefix-specific default routes (Name Collectors, NAC)

Open Research Questions

Questions

- Which *push* strategy fits our need?
- Which proactive cache replication strategy to use?
  - must be aware of content prioritization
- How to realize *subscribe* / *unsubscribe*?
  - refresh subscription lifetimes
  - subscription cleanup in case of faults

Thanks for Listening!

Any Questions?