

# A Broker-based Pub/Sub System for NDN

**Haksuh Kim, Sung-Hyuk Byun, Namseok Ko, and Sun-Me Kim**

Electronics and Telecommunications Research Institute (ETRI)

{tuple, shbyun, nsko, kimsunme}@etri.re.kr

Dec. 1, 2020

# Motivations

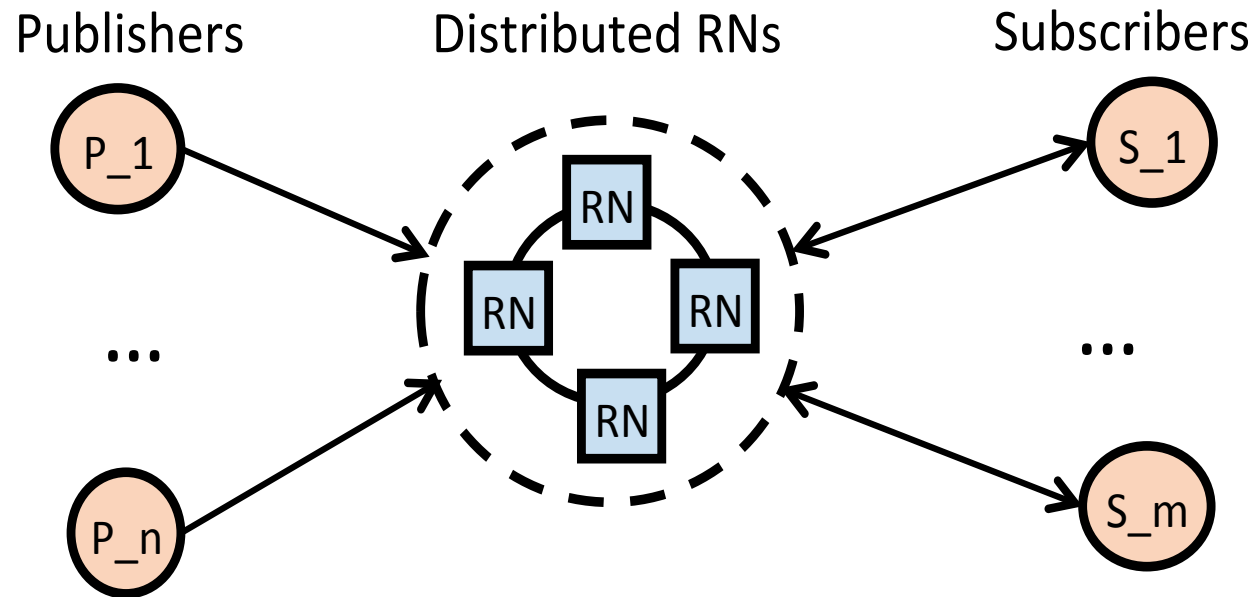
- ▶ A few Pub/Sub mechanisms for current NDN
  - Partial-sync mode of Psync
  - Periodic requests with long-lived Interests
- ▶ Not scalable and limited especially in low-performance IoT producers
- ▶ Not flexible as in IP-based mechanisms : no wildcard topic matches
- ▶ We need a flexible and scalable pub/sub architecture for NDN

# Design Directions

- ▶ To cope with the issues on low-performance producers
  - Broker-based approach
- ▶ To support scalability
  - Multiple brokers
- ▶ To support flexibility
  - MQTT-like wildcard Topic matches
    - Single-level wildcard (+), e.g., /prefix/etri/7d+/temp
    - Multi-level wildcard (#), e.g., /prefix/etri/7d/#
    - Multiple wildcards can be supported in a Topic
  - ※ Topics are defined by subscribers

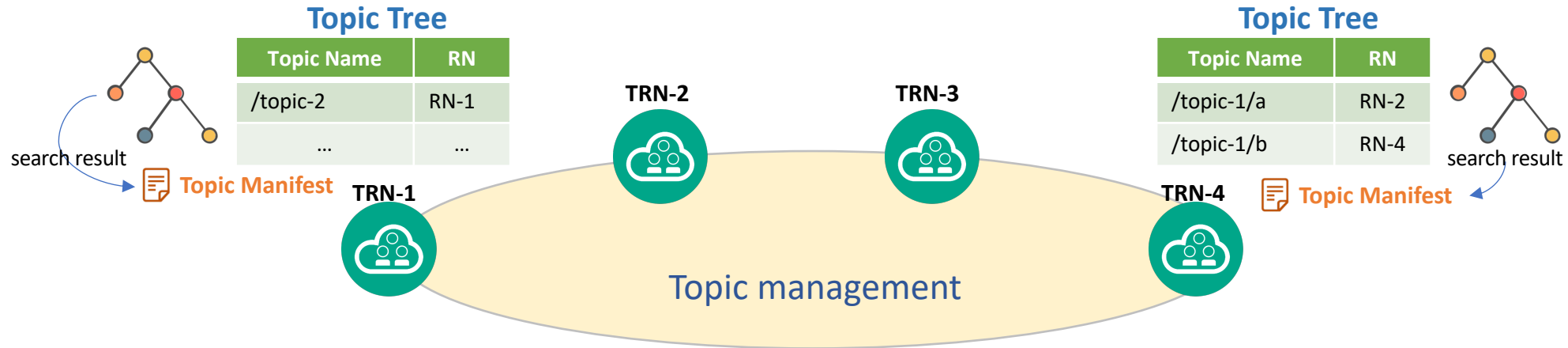
# Architecture Overview

- ▶ Multiple brokers (or Rendezvous Nodes)
  - Do brokering of publishers and subscribers
  - Store published data for limited performance publishers
    - ⊗ Note that data can be stored in devices themselves and other external repositories
  - Manage published data (names) based on DHT

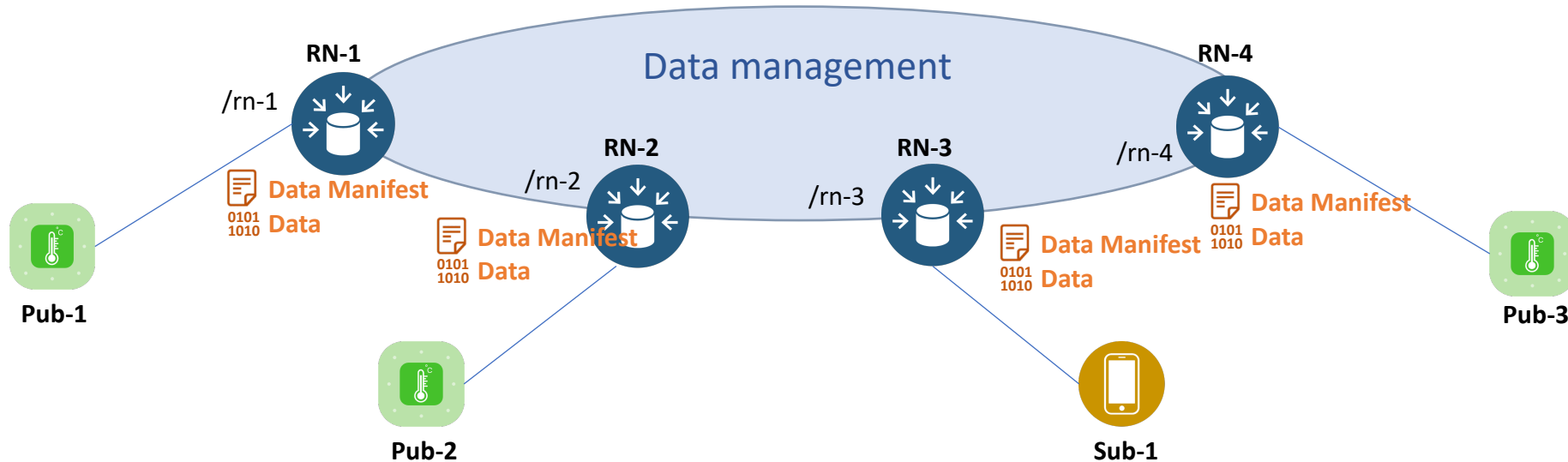


High-level Architecture

# Architecture Overview



Logical separation of Topic and Data management



# Architecture Overview

- ▶ Logical separation of Topic and Data management
  - A Data RN is an RN for data management and a Topic RN is an RN for topic management
  - Pub/sub service has a service prefix, **rn** and each RN has its own node prefix, **rn-x**
- ▶ Naming scheme
  - Data
    - /<data stream name>/[sequence-number] : e.g., /etri/7d/room385/temp/1
  - Command
    - /<service prefix>/<command>/<data name>: e.g., /rn/PA/etri/7d/room385/temp

# Protocol Messages (Commands)

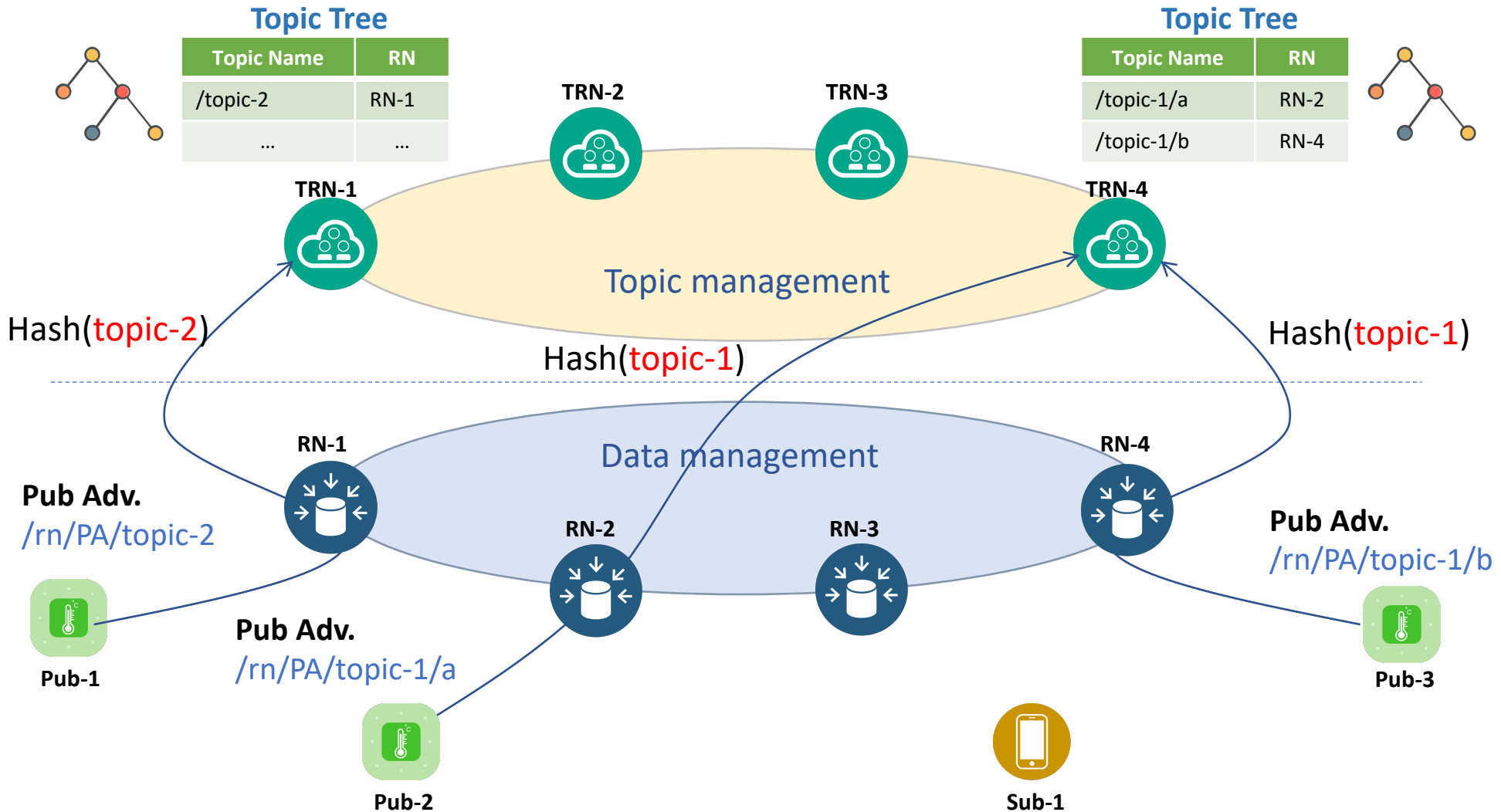
## ▶ Publish Procedure

- Publish Advertisement (PA): Advertise the name of a data stream to publish
- Publish Unadvertisement (PU): Revoke the publish of a data stream
- Publish Data (PD): publish a data

## ▶ Subscribe Procedure

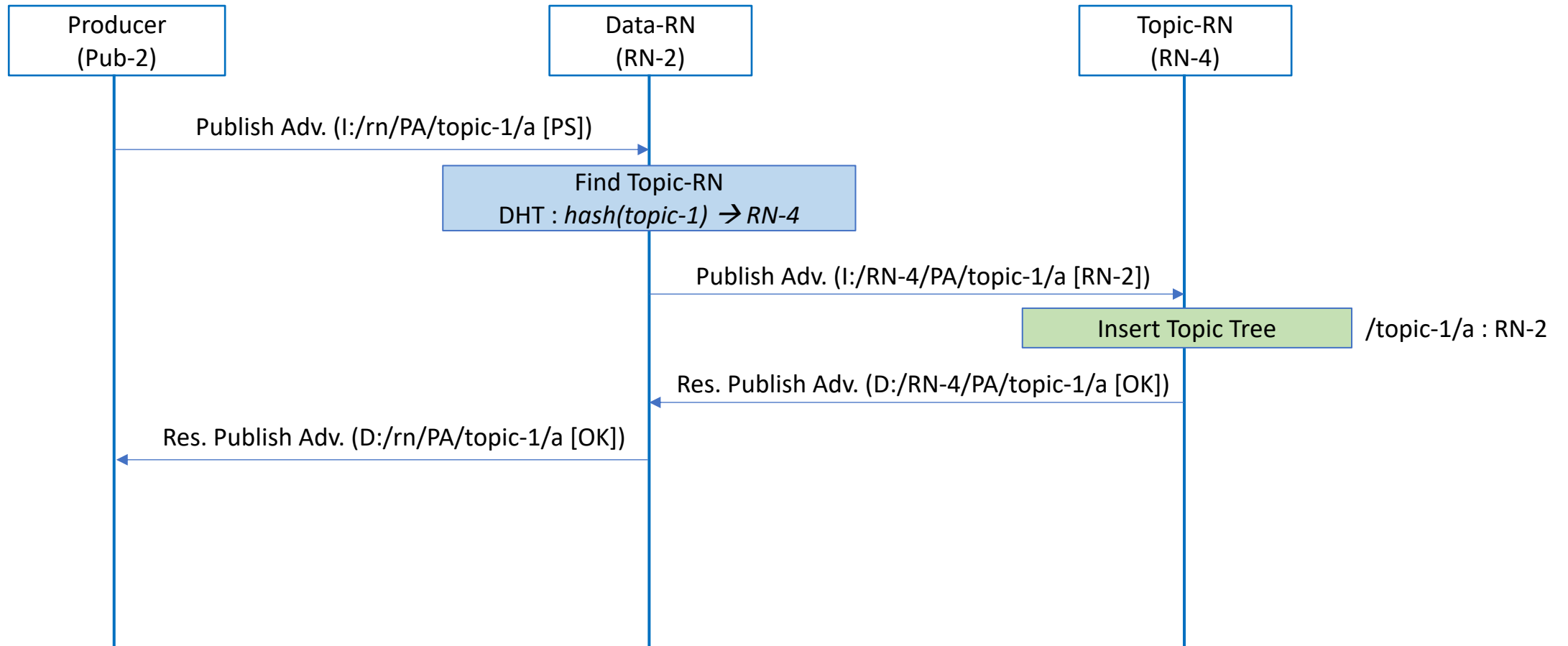
- Subscribe Topic Subscription (ST): Subscribe to a topic (request a topic manifest)
  - \* Topic manifest : a list of data RN holding subscribed data streams
- Subscribe Manifest Request (SM): Request a data manifest (to a specific Data RN)
  - \* Data manifest : data names for a data stream
- Subscribe Data Request (SD): Request a data (to a specific Data RN)

# Publish Advertisement

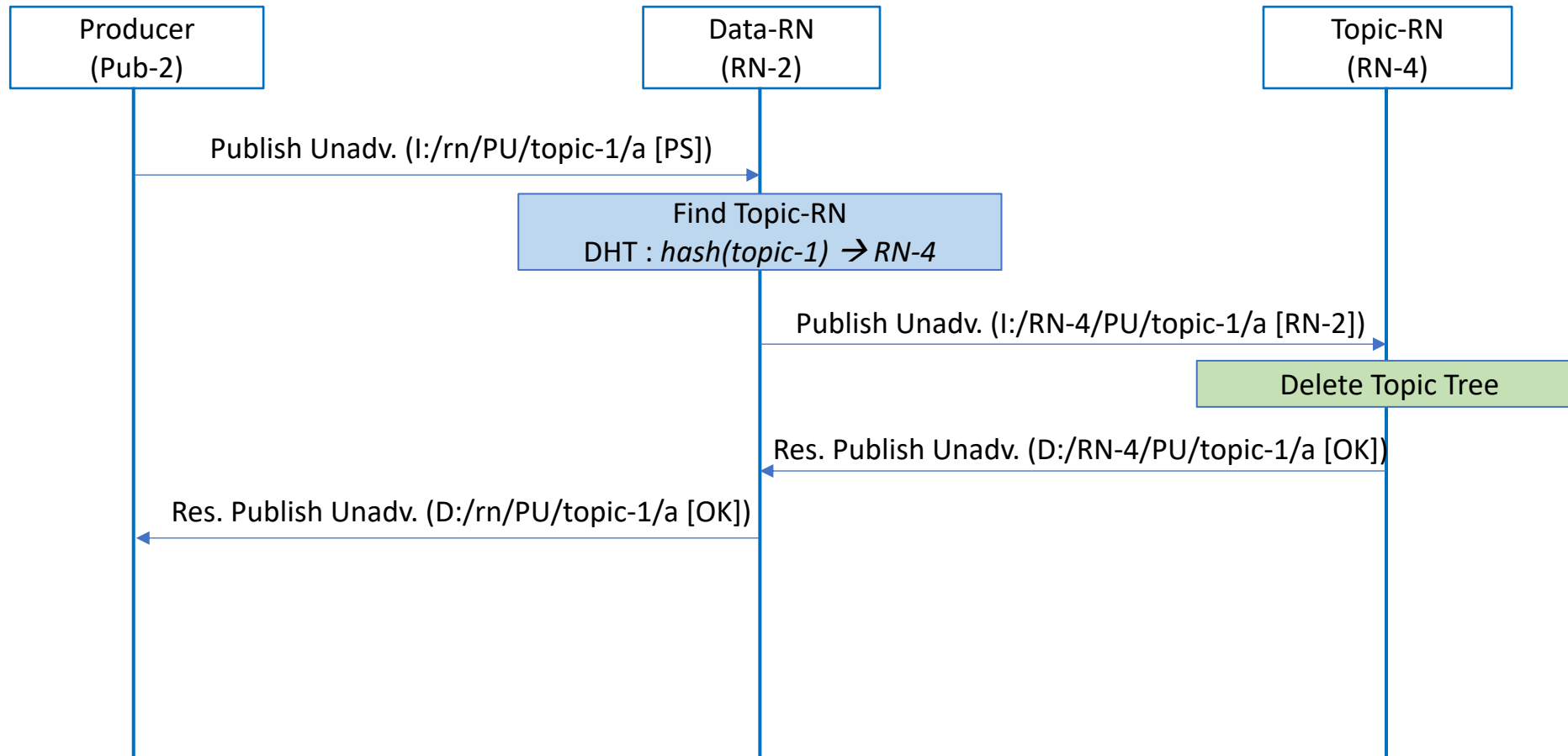




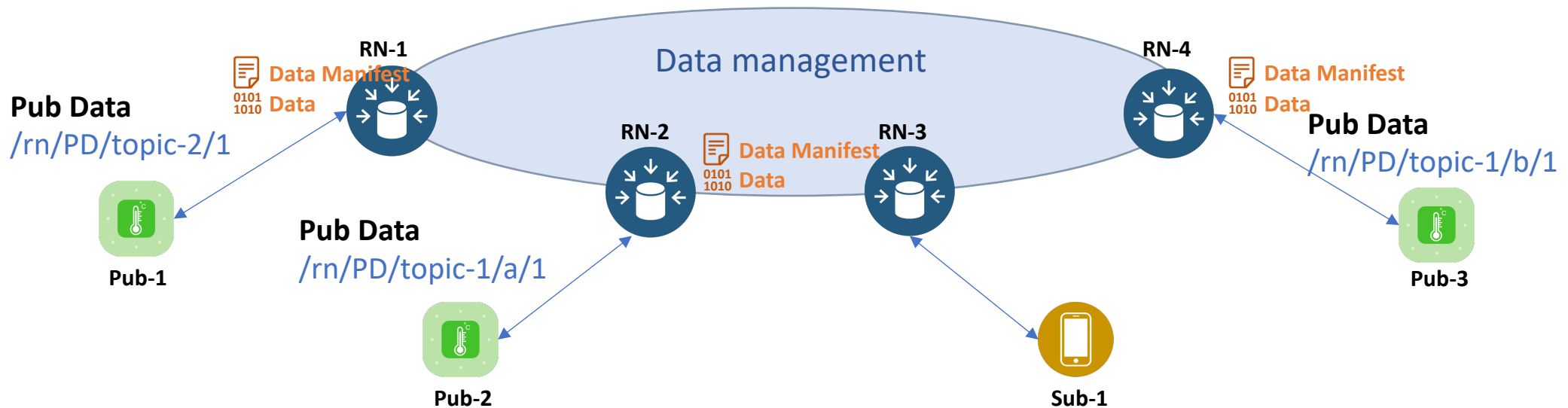
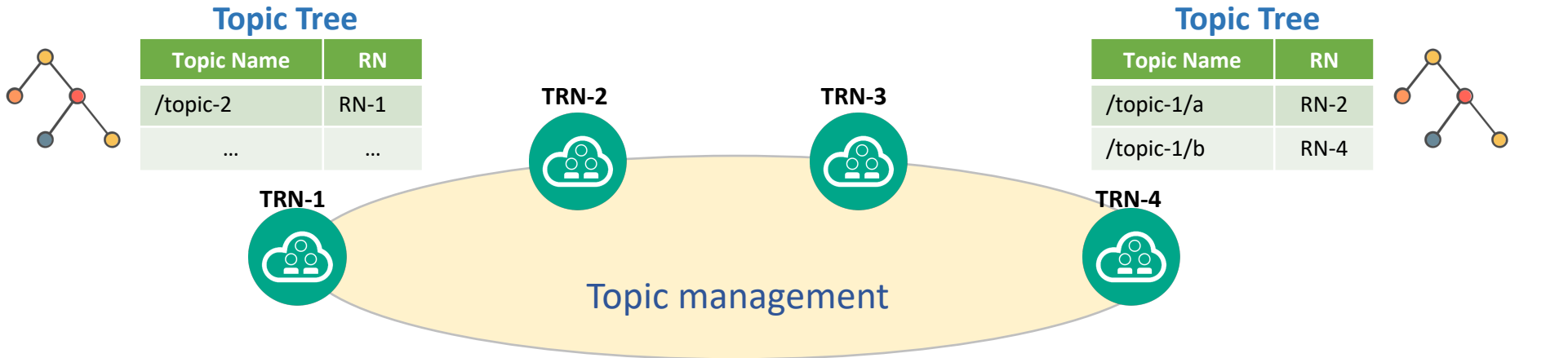
# Publish Advertisement



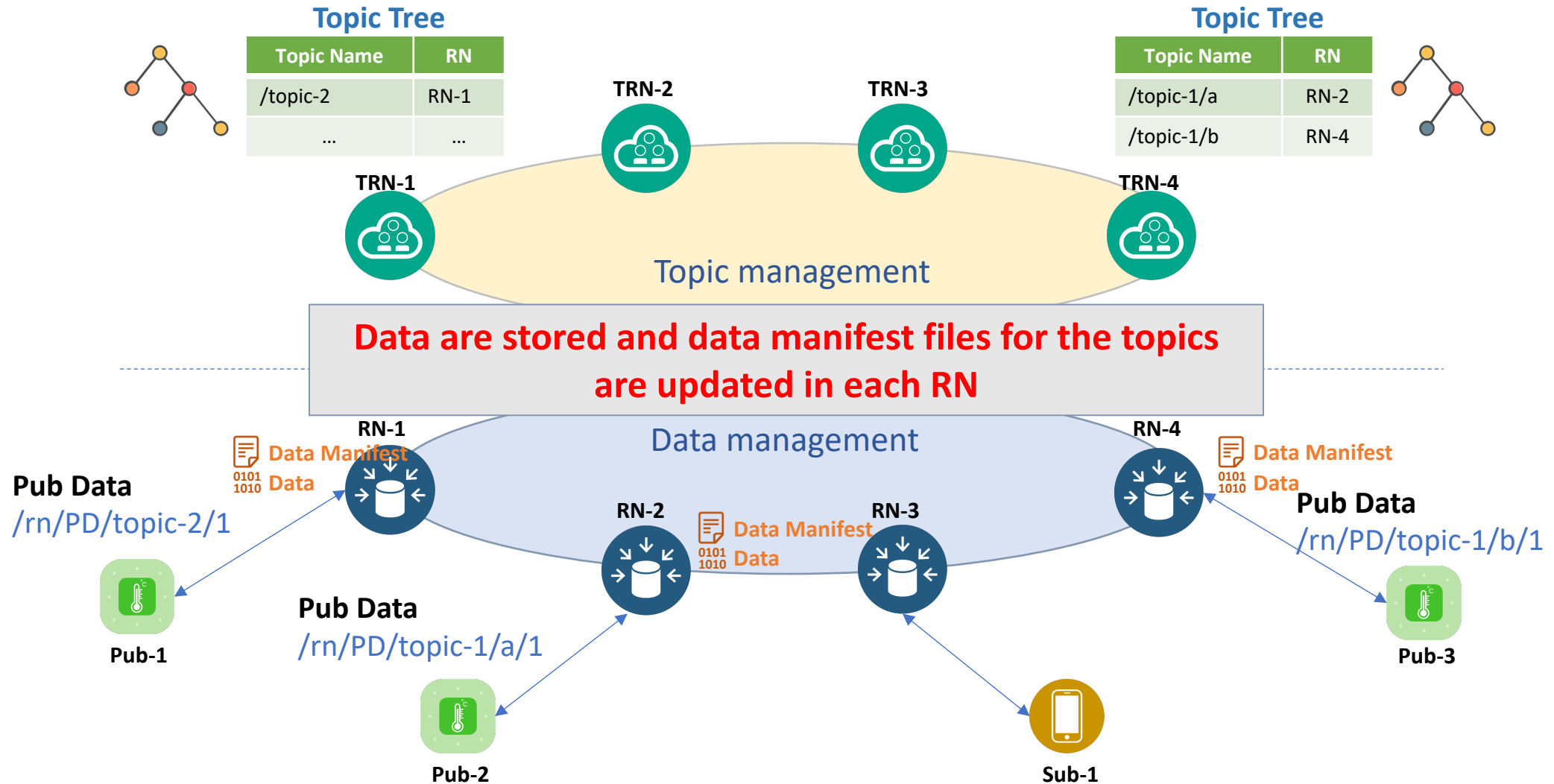
# Publish Unadvertisement



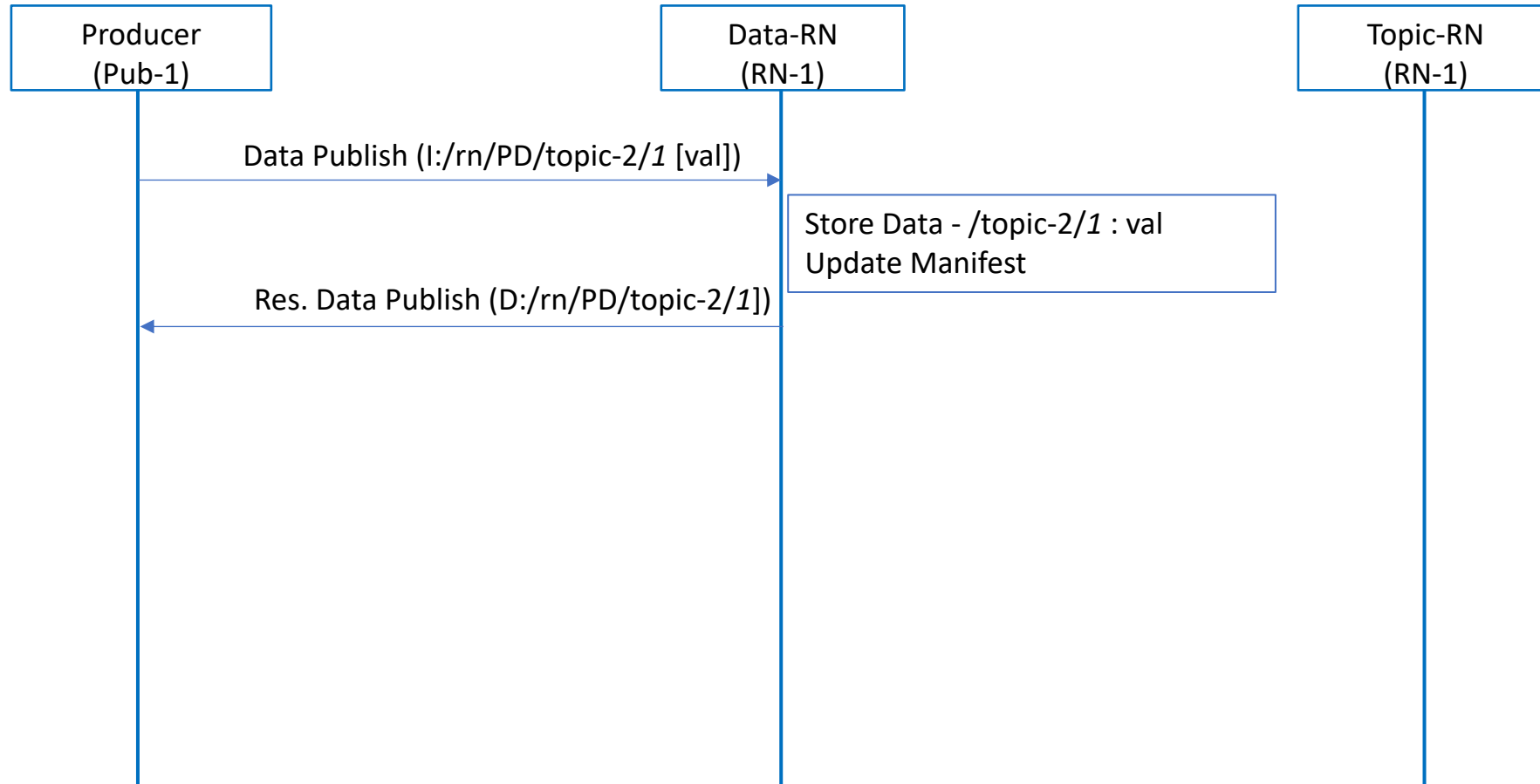
# Publish Data



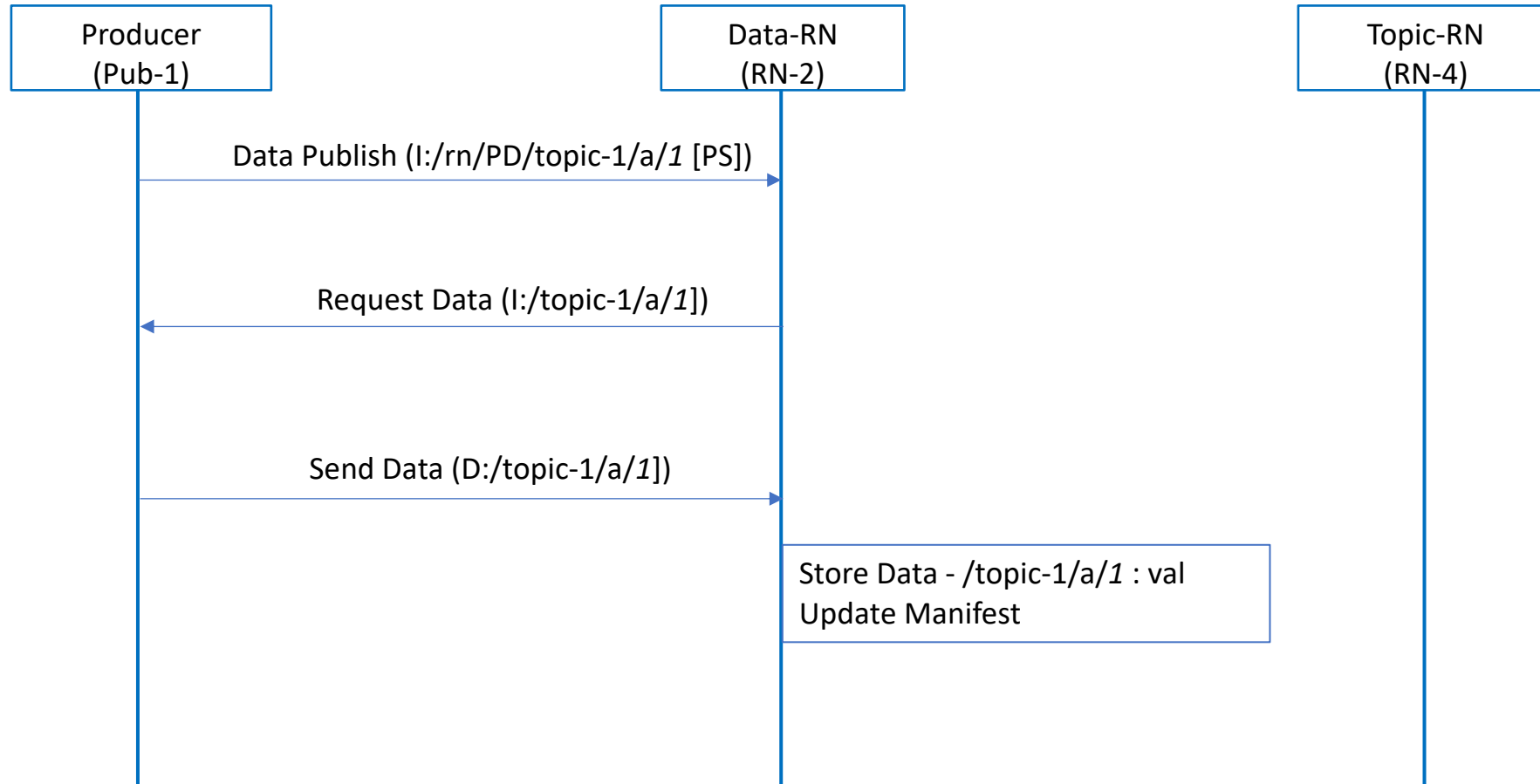
# Publish Data



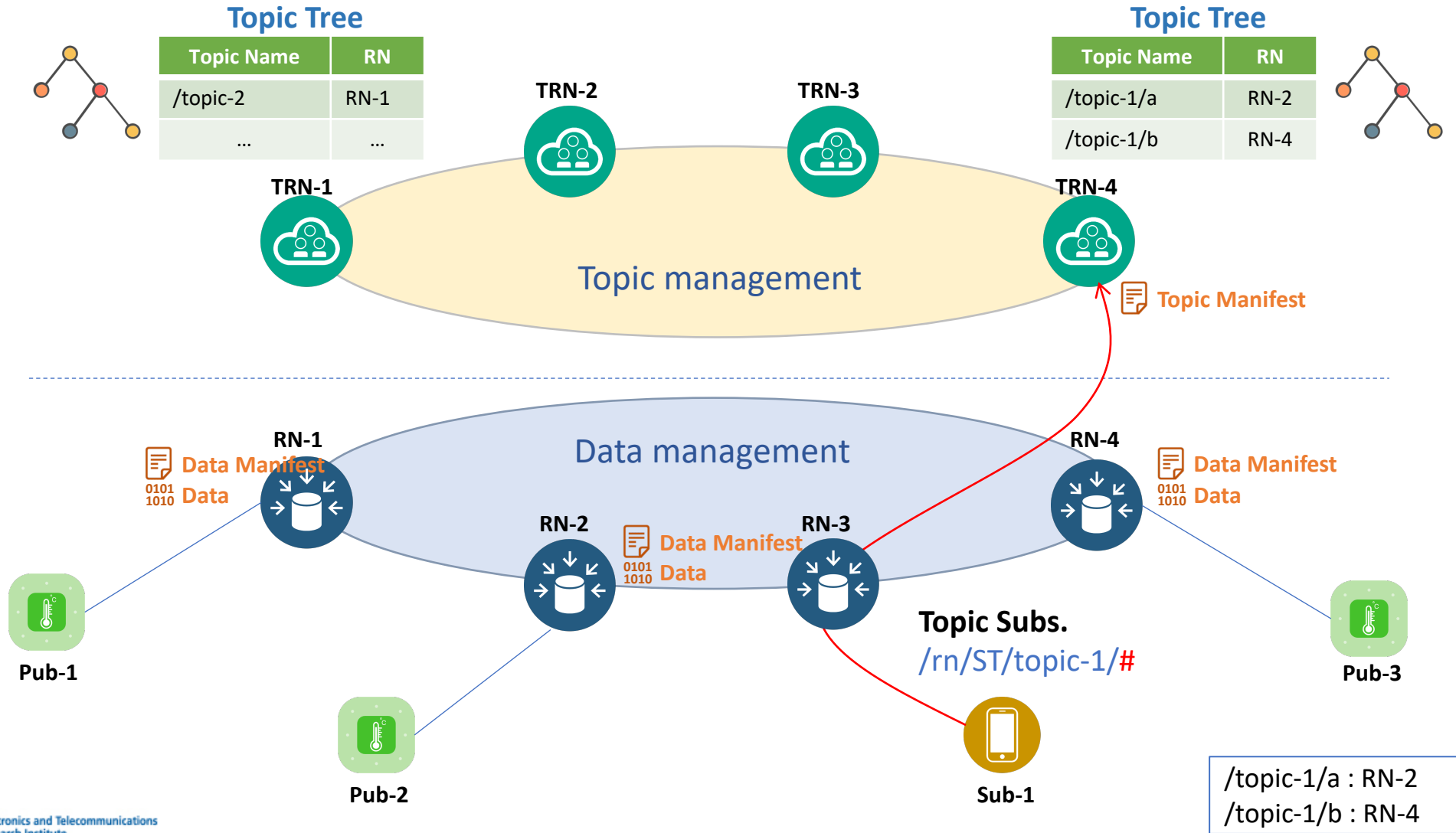
# Publish Data (small data)



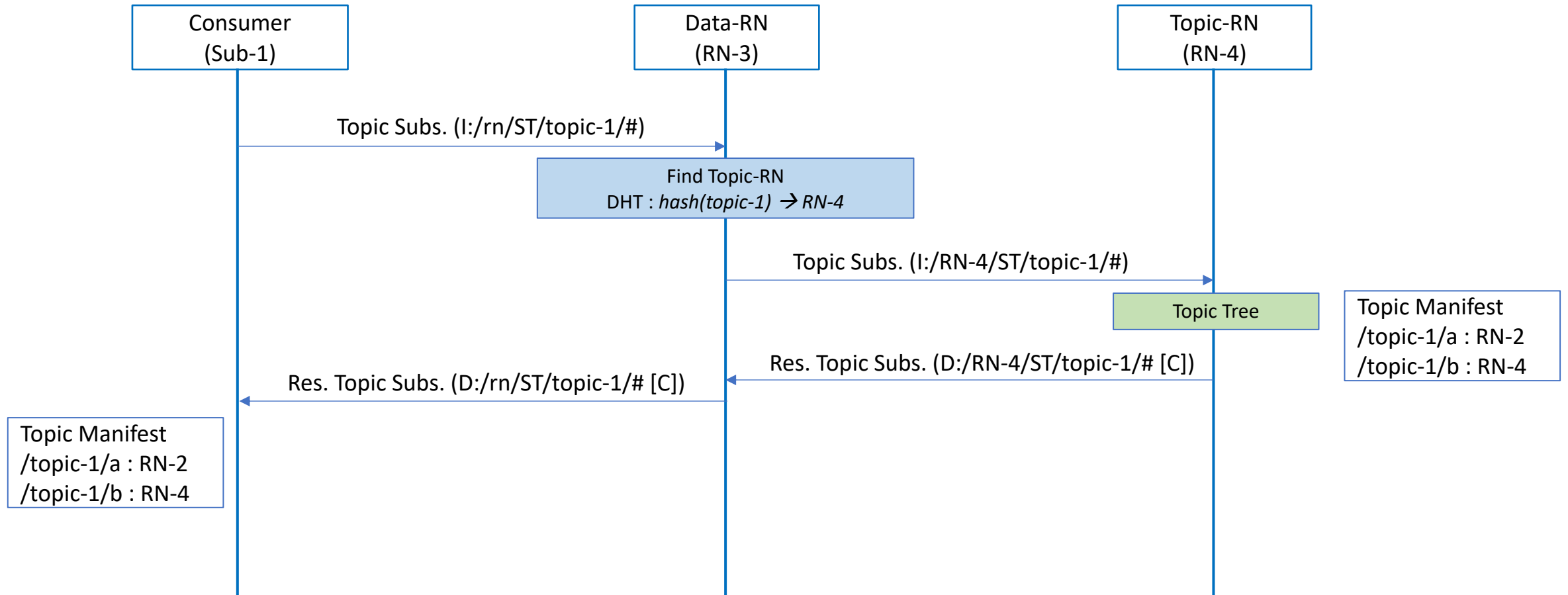
# Publish Data (large data)



# Topic Subscription

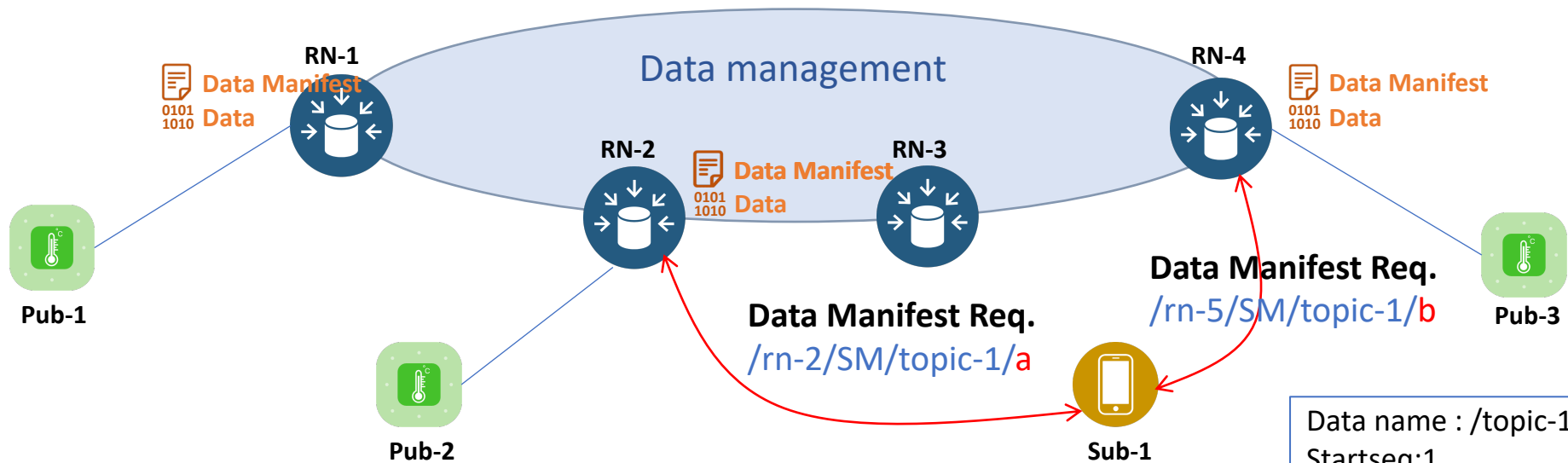
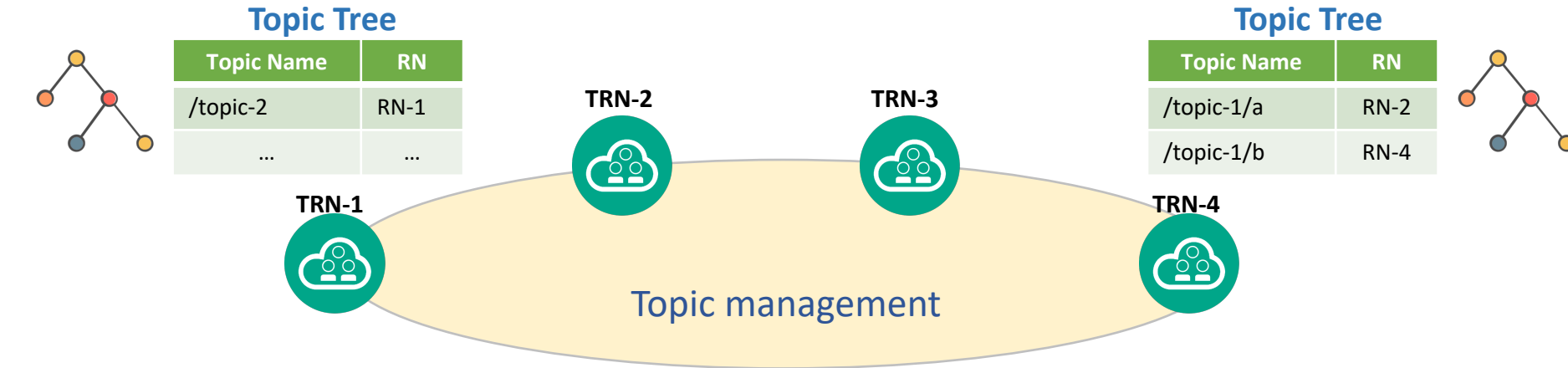


# Topic Subscription



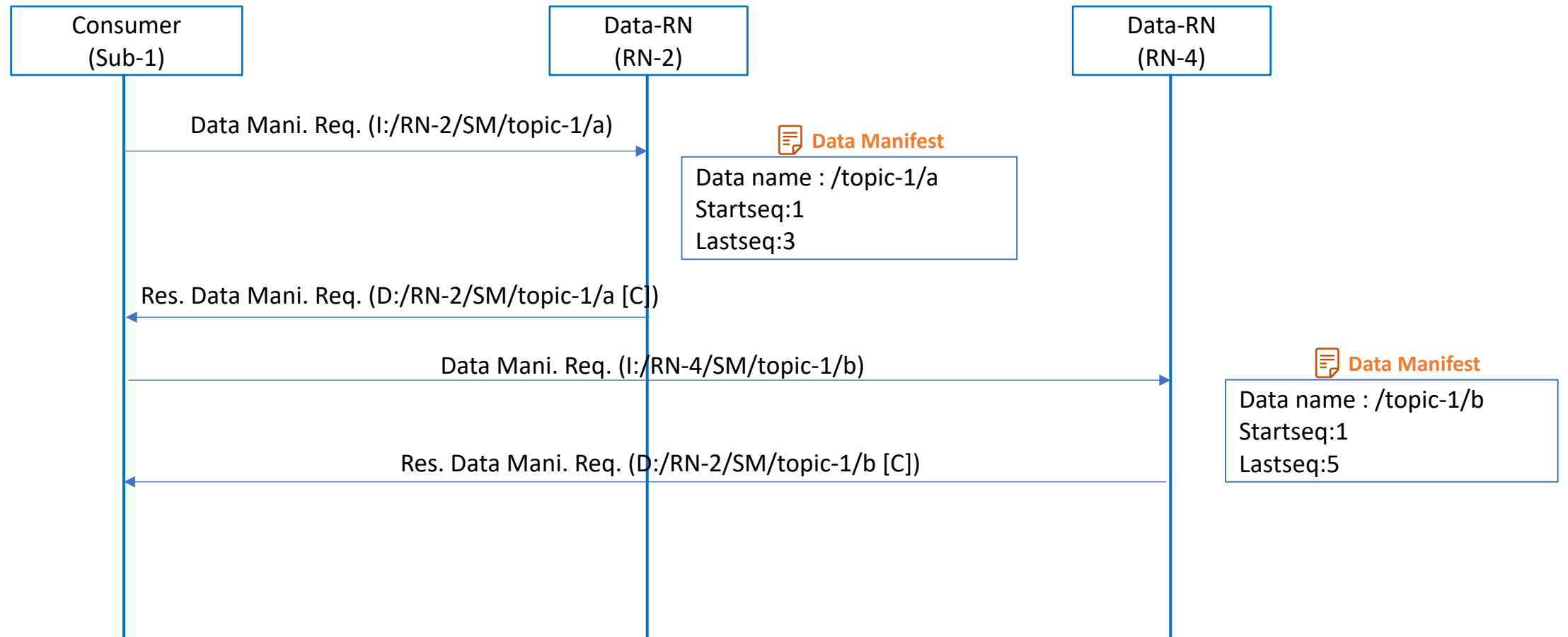


# Data Manifest Request

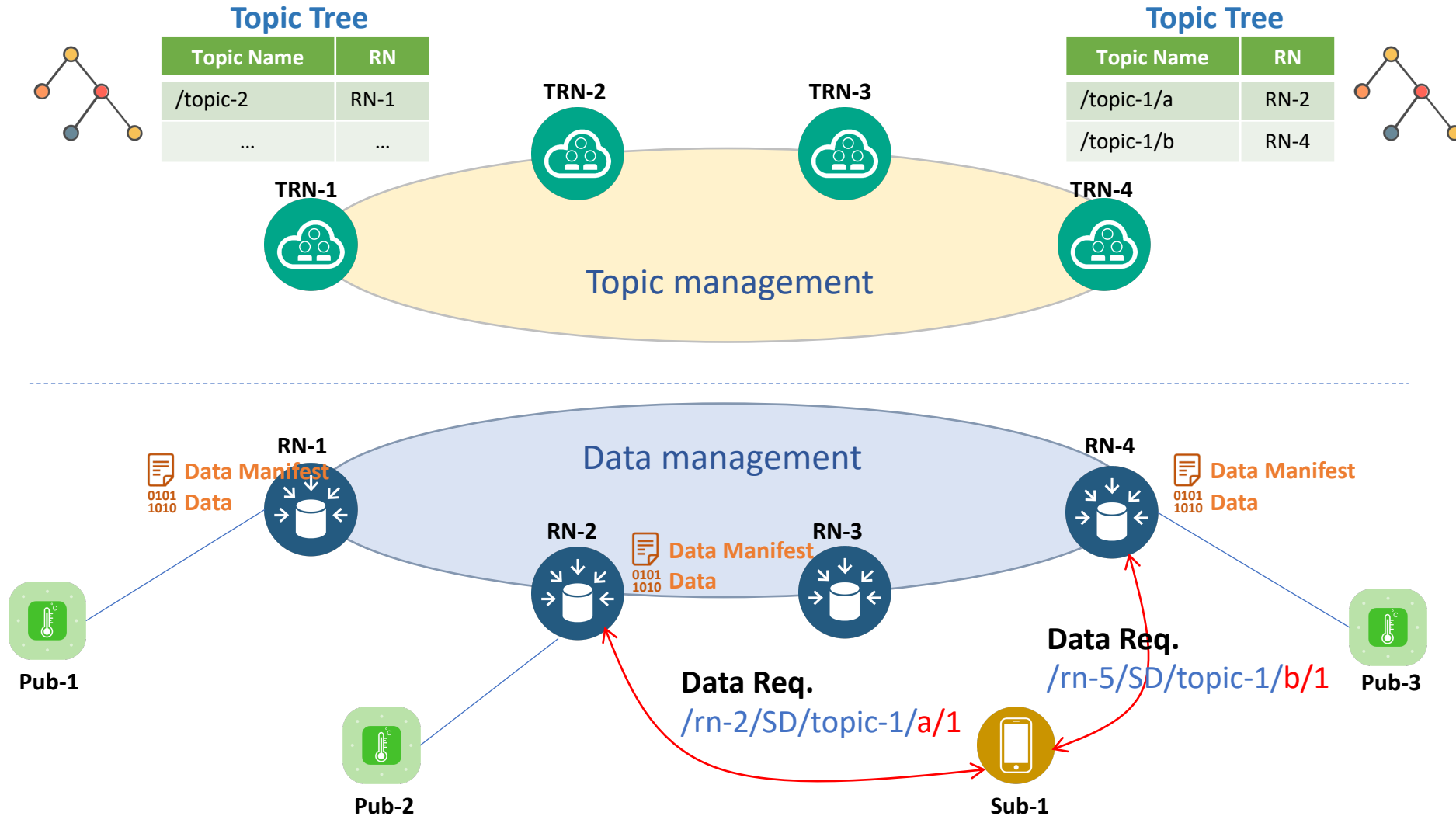


Data name : /topic-1/a  
 Startseq:1  
 Lastseq:3

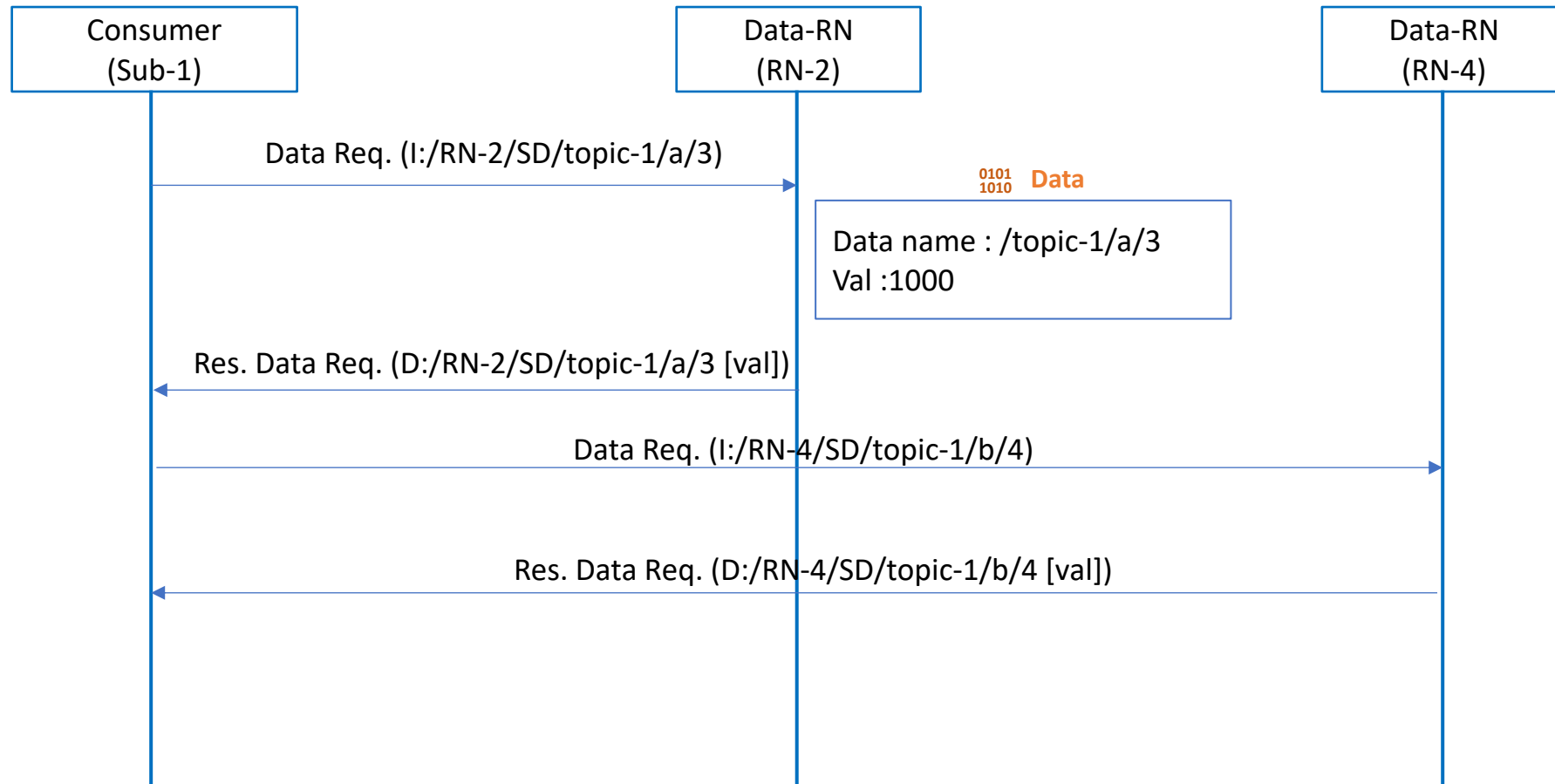
# Data Manifest Request



# Data Request



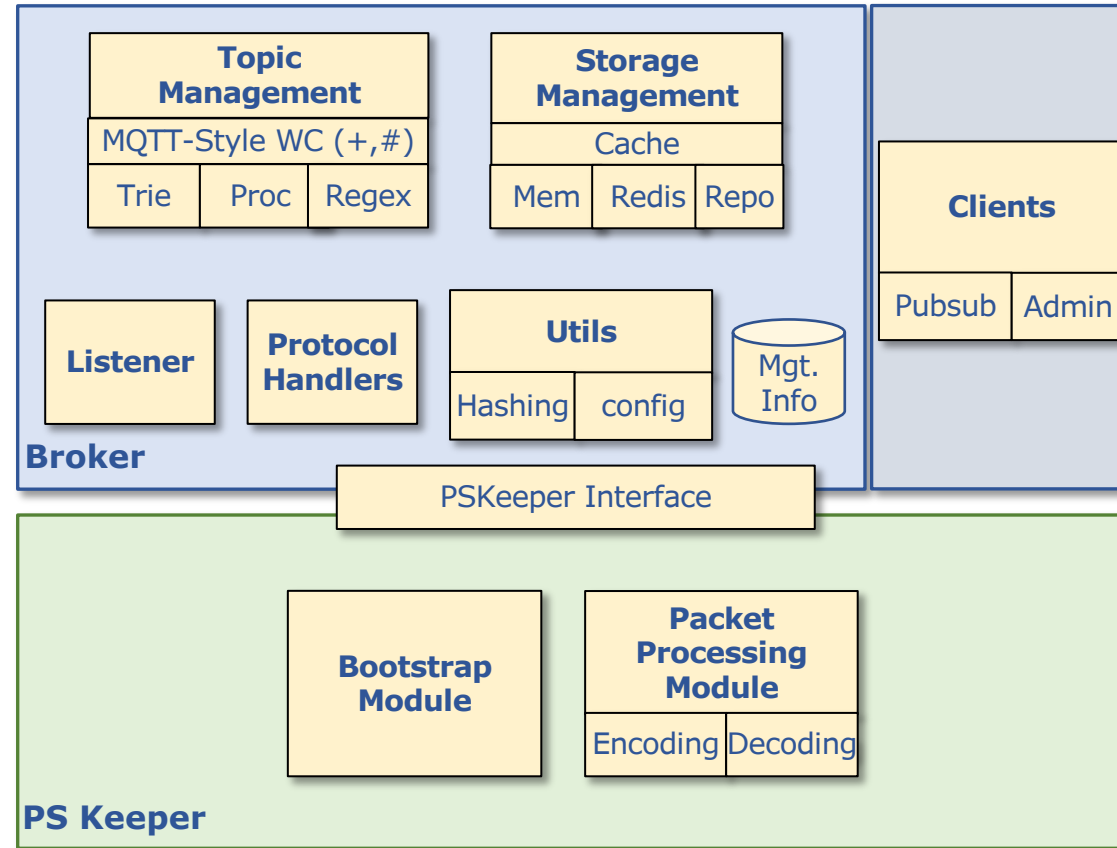
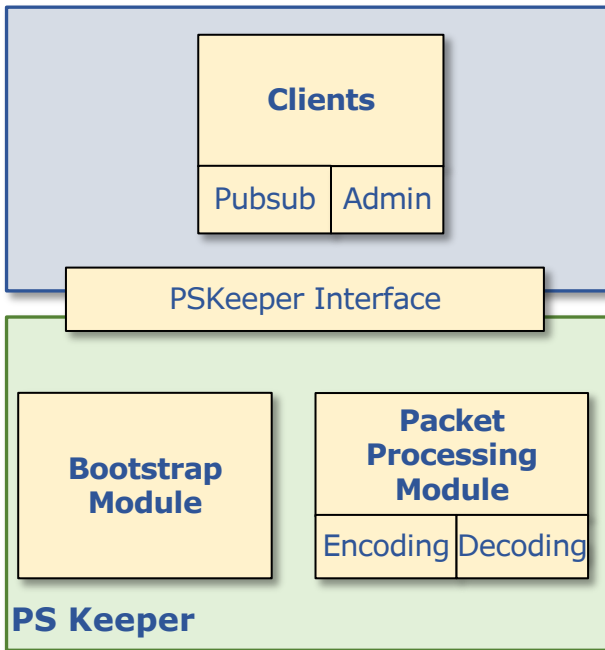
# Data Request



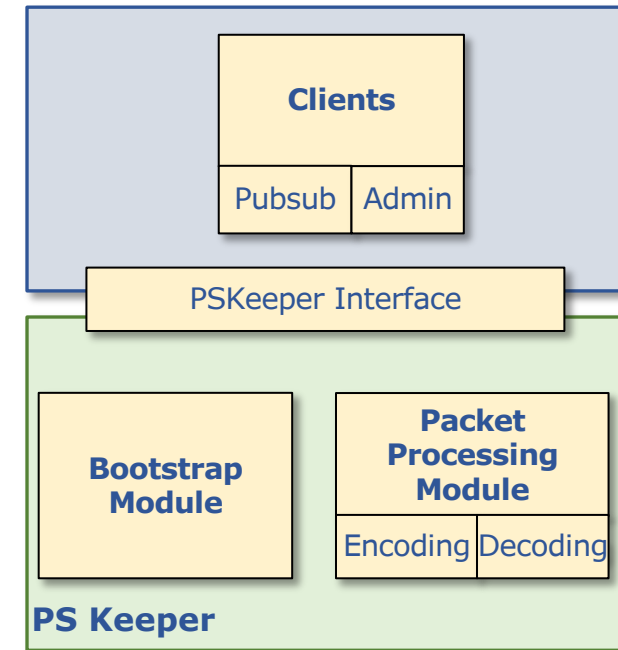
# SW Function Blocks

## Brokers

### Clients

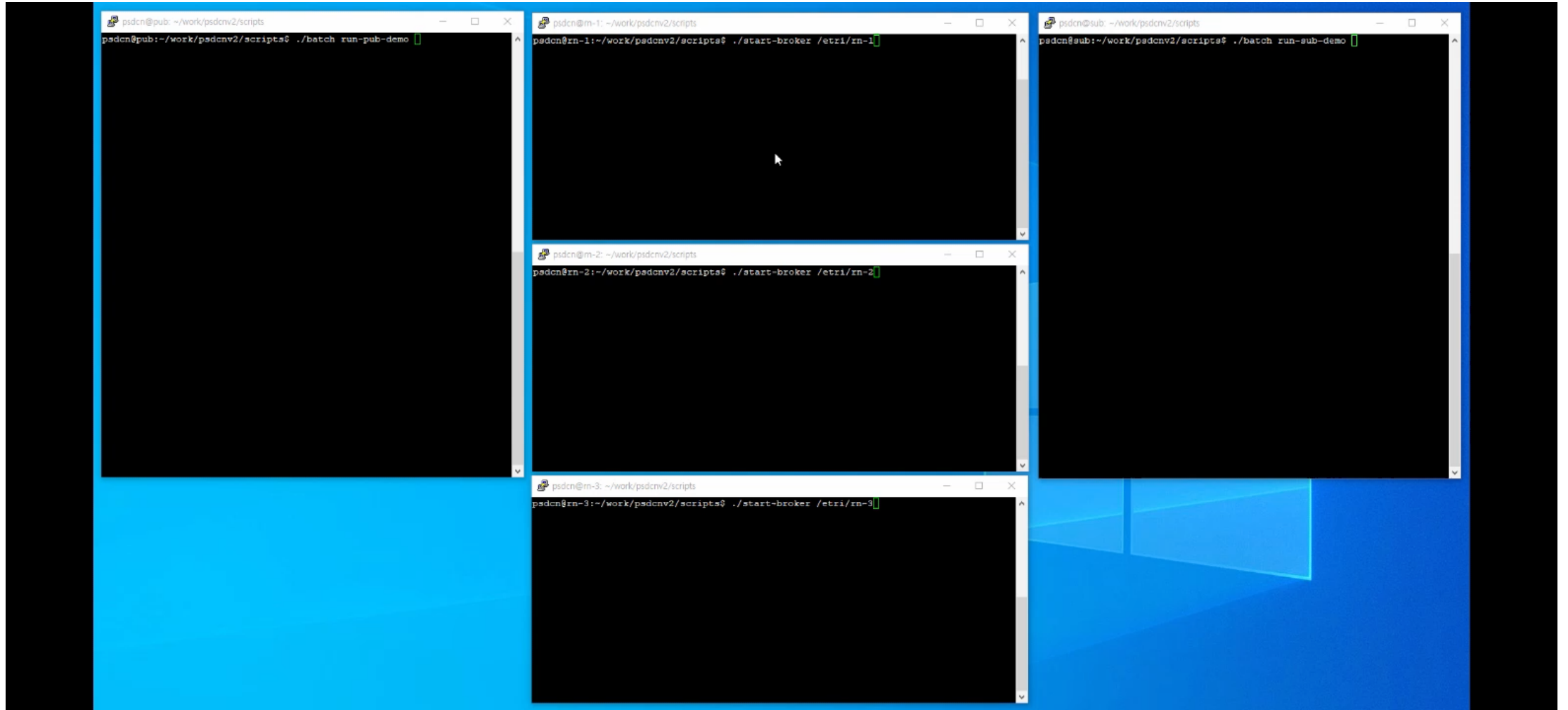


### Subscribers



NDN

# Demo



# Summary

- ▶ Broker-based Pub/Sub for NDN
- ▶ Scalable and flexible than existing mechanisms
- ▶ Plan to release our code as open source software soon

Thank You!