

Subscription to Multiple Stream Originators

draft-zhou-netconf-multi-stream-originators

Tianran Zhou

Guangying Zheng

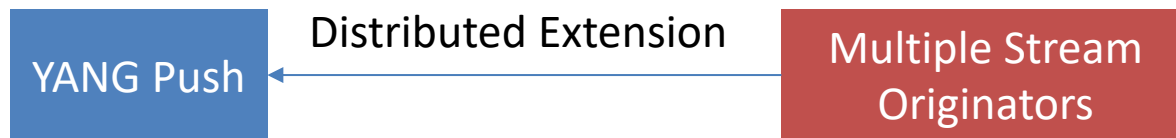
Eric Voit

Alexander Clemm

Andy Bierman

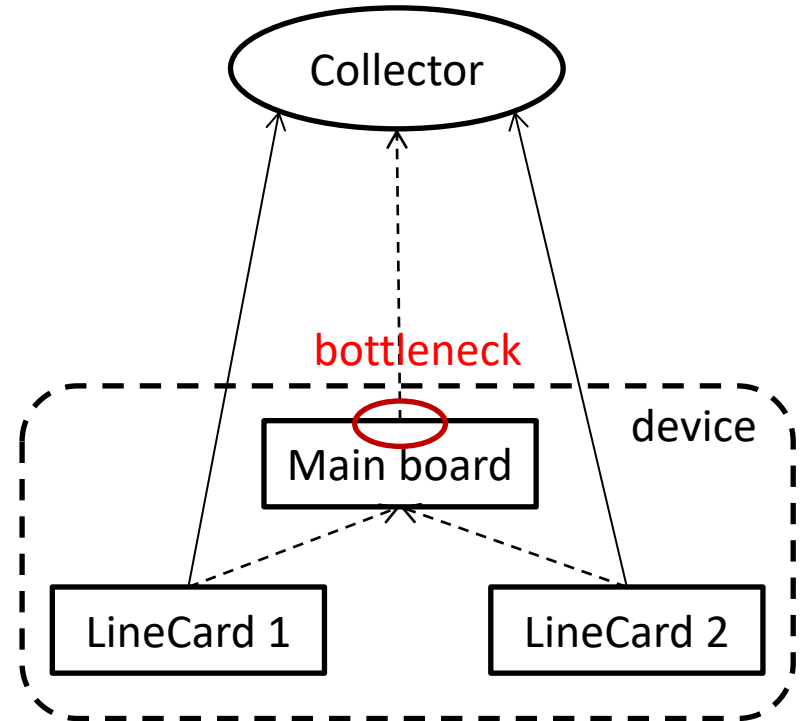
Introduction

- **Distributed data collection** mechanism that allows multiple data streams to be managed using a **single subscription**.
- Transport independent



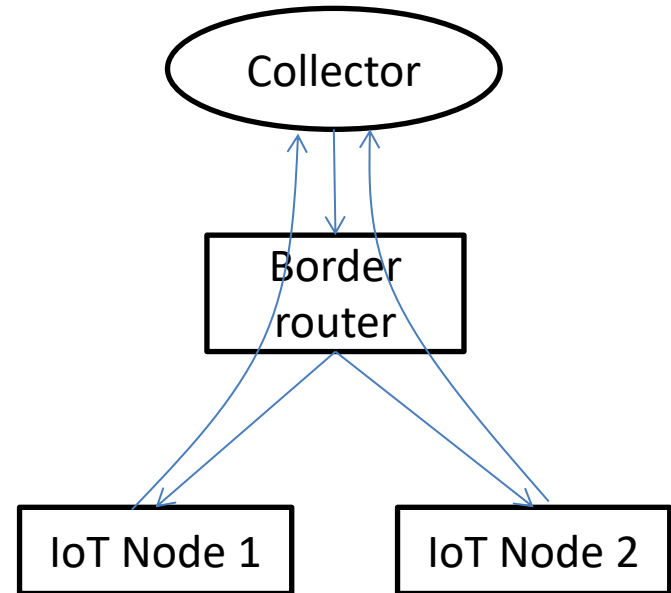
Use Case 1

- Large amount of data collection from devices with main board and line cards.
- Existing solution consider only one push server reside in the main board.
 - Result in performance bottleneck when data are forwarded to the main board and converged to one consolidated stream.
- Request for **distributed data collection mechanism** which can directly push data from line cards to a collector.



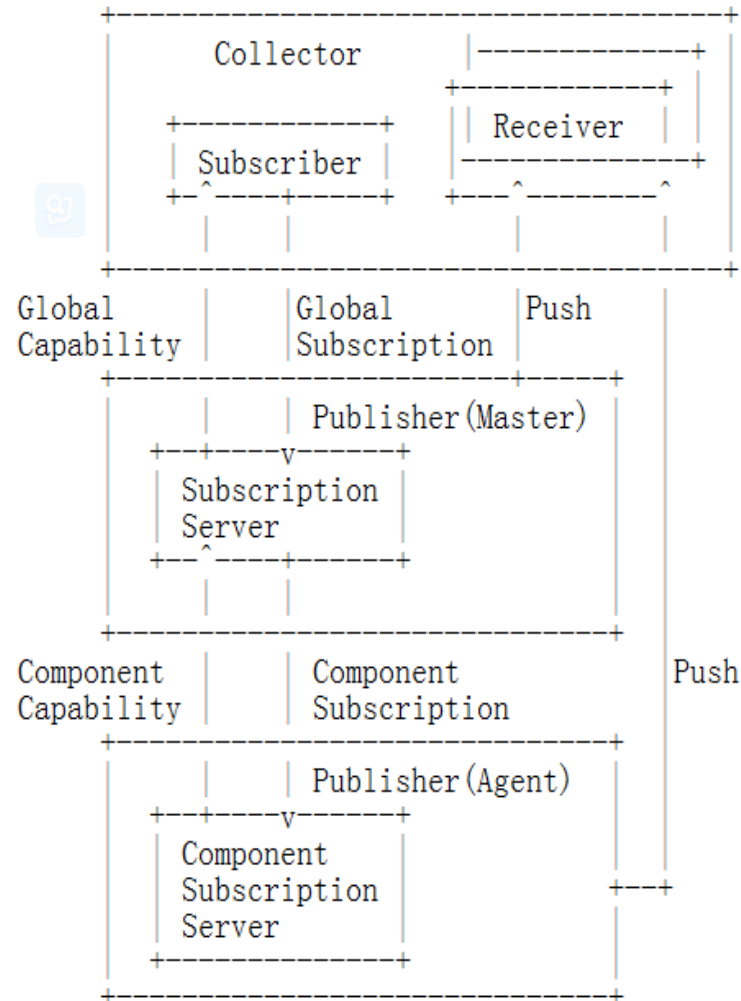
Use Case 2

- IoT data collection
- Collector cannot subscribe/access data directly from IoT nodes.
 - subscribe data from border router
 - border router distribute the subscription to Nodes.
 - IoT Nodes stream data to the collector through BR.
 - Collector assembles the subscription data.
- **The border router does not assemble data as a broker.**



Solution Overview

- Collector
 - Subscriber
 - Receiver
- Distributed Publisher
 - Master with the Subscription server
 - Agent with the Component subscription server
- Assumption
 - The connection between the master and the agents exist
 - Master knows the resource-location map.



Subscription Decomposition

- The Master
 - expose the Global Capability that can be served by multiple Publishers;
 - disassemble the Global Subscription to multiple Component Subscriptions, and distribute them to the corresponding telemetry sources;
 - notify on changes when portions of subscription moving between different Agents over time.
- The Master may need a data structure, typically a Resource-Location Table, to keep track of the mapping between the resource and the corresponding location of the Subscription Server which commits to serve the data.

Publication Composition

- The Receiver recognizes data records associated with one subscription according the **Subscription ID**.
- The Receiver assembles data generated at the same time period based on the **recording time** consisted in each data record.
- Receiver need to know the **number of Component Subscriptions** which the Global Subscription is decomposed to.
 - **Propose to add a list of Publisher ID**
 - The "subscription-started" and "subscription-modified" notification

Subscription State Change Notifications

- Two options:
 - Each agent sends its own notification to the subscriber
 - **Master sends a notification to acknowledge the Globe Subscription.**
- All the subscription state change notifications **MUST** be delivered by the **Master Publication Channel** which is the session between the Master Publisher and the Receiver.
- When the subscription decomposition result changed, the "subscription-modified" notification **MUST** be sent to indicate the new list of Publishers.

Support Multiple Transports

```

+--rw subscription* [id]
  +--rw id
  |   subscription-id
  ...
  +--rw transport?                transport
  |   {configured}?
  +--rw encoding?                 Encoding
  ...
  +--rw receivers
  |   +--rw receiver* [name]
  |       +--rw name                string
  |       |...
  |       Augment Address and port?

```

ietf-subscribed-notifications@2019-01-16.yang

Next

- Need dynamic subscription?
- Any other issues need to consider for this distributed extension of the YANG-Push work?
- Ask for the WG Adoption.

Thank you