

Adaptive Subscription and Bulk Subscription

draft-wang-netconf-adaptive-subscription-01

draft-wang-netconf-bulk-subscribed-notifications-02

Qin Wu (bill.wu@huawei.com)

Wei Song (songwei80@huawei.com)

Liang Geng(gengliang@chinamobile.com)

Peng Liu(liupengyjy@chinamobile.com)

Hui Cai (caihui@chinamobile.com)

Document Status

- draft-wang-netconf-adaptive-subscription
 - v-00 was first presented in the IETF 107 meeting, and it was suggested to setup design team to progress this work.
 - It was suggested to align with ECA model (draft-wwx-netmod-event-yang-08)
 - provision a ratio of the new measurement interval
 - provision "5" means do it five times faster than the original period
 - The characterization of on-change subscription as a degenerative case of periodic should be fixed
- draft-wang-netconf-bulk-subscribed-notifications
 - V-01 was secondly presented in the IETF 107 meeting, and it was suggested to setup design team to facilitate the discussion.
 - It was suggested to add default value of 'max-bundle-latency' for the publisher
 - Criteria to classify subscriptions based on different subscriber transport sessions, encoding, dscp, weight was discussed
- The latest update of draft-wang-netconf-adaptive-subscription is v-(01), changes compared to previous versions:
 - Add usage example of adaptive subscription;
 - Align with ECA model and data path, data, condition expression and ratio(i.e.,count) parameters based on earlier discussion;
 - Highlight the motivation to add adaptive subscription support;
- The latest update of draft-wang-netconf-bulk-subscribed-notifications is v-02, changes compared to previous versions:
 - Motivation polishing
 - Add bundle size to the model structure
 - Subtrees and data nodes path fixing

Adaptive Subscription Recap

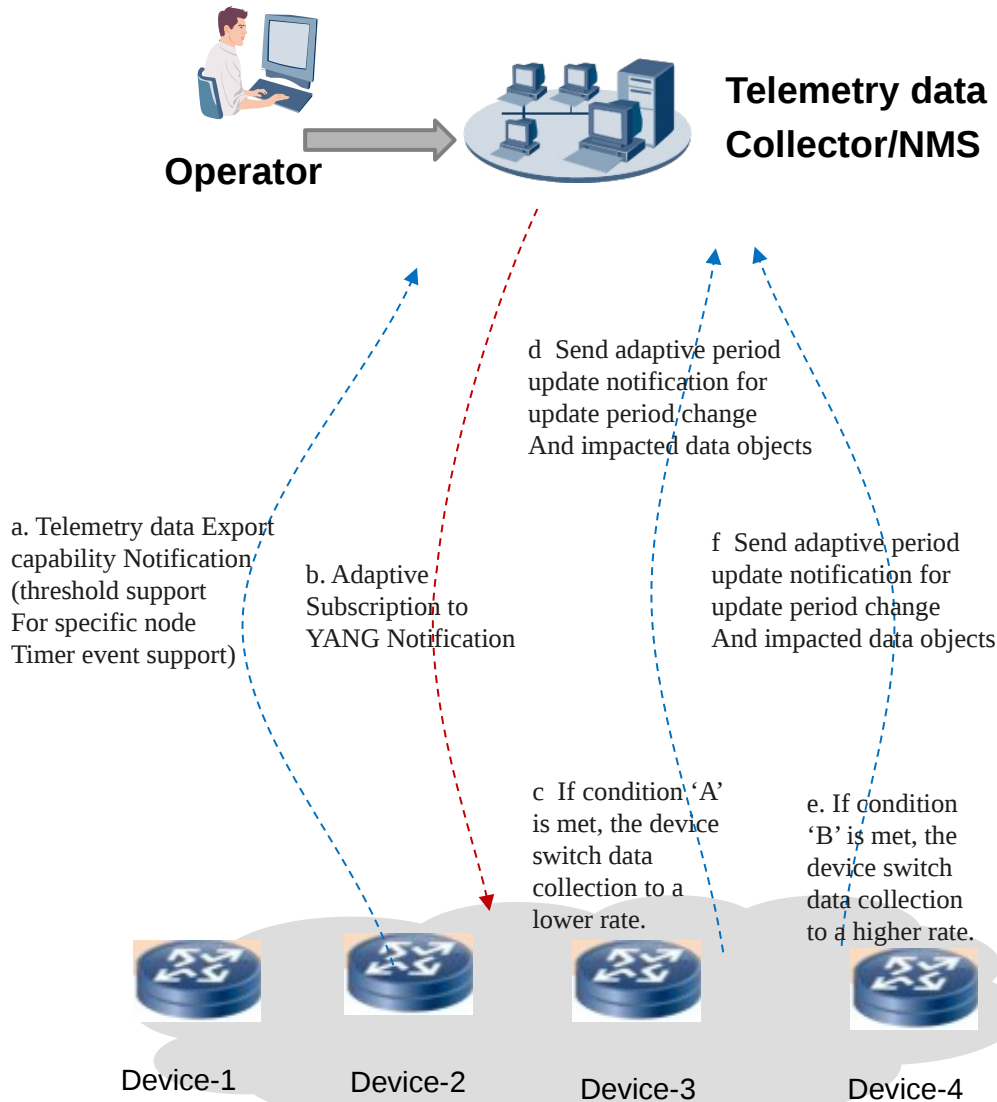
- Motivation:

- Where an increased data collection rate is being used, it becomes more likely that a burst of streamed data may temporarily overwhelm a receiver and consume expensive network resource(e.g., air interface resource).
- If the rate at which we can collect a stream of data is set too low, these telemetry data are not sufficient to detect and diagnose problems and verify correct network behavior.
- There is a need for a service to configure both collectors and publishers with multiple period intervals, counter threshold and automatically switch to different period intervals according to resource usage,
 - e.g., when the wireless signal strength falls below a configured low watermark, the subscribed data can be streamed at a higher rate
 - while when the wireless signal strength crosses a configured high watermark, the subscribed data can be streamed at lower rate.

- Goal:

- Define a YANG data model and associated mechanism enabling subscriber's adaptive subscriptions to a publisher's event streams.
 - allows both subscriber and publisher to automatically adjust the volume of telemetry traffic sent from publisher to the receivers.

Adaptive Subscription Solution Overview



```

augment /sn:subscriptions/sn:subscription/yp:update-trigger:
  +--rw (adaptive-subscription)?
  +--:(adaptive-subscriptions)
  +--rw adaptive-subscriptions
    +--rw data-path?          yang:xpath1.0
    +--rw target?             <anydata>
    +--rw adaptive-period* [condition-expression]
    +--rw condition-expression string
    +--rw watermark?         uint32
    +--rw period              centiseconds
    +--rw count?             uint16
    +--rw anchor-time?       yang:date-and-time
augment /sn:establish-subscription/sn:input/yp:update-trigger:
  +--:(adaptive-subscription)?
  +--:(adaptive-subscriptions)
  +-- adaptive-subscriptions
    +-- data-path?          yang:xpath1.0
    +-- target?             <anydata>
    +-- adaptive-period* [condition-expression]
    +-- condition-expression string
    +-- watermark?         uint32
    +-- period              centiseconds
    +-- count?             uint16
    +-- anchor-time?       yang:date-and-time

notifications:
  +--n adaptive-period-update
  +--ro id?                  sn:subscription-id
  +--ro period               centiseconds
  +--ro count?              uint16
  +--ro anchor-time?        yang:date-and-time
  +--ro datastore            identityref
  +--ro (selection-filter)?
  +--:(by-reference)
  | +--ro selection-filter-ref          selection-filter-ref
  +--:(within-subscription)
  +--ro (filter-spec)?
  +--:(datastore-subtree-filter)
  | +--ro datastore-subtree-filter?    <anydata> {sn:subtree}?
  +--:(datastore-xpath-filter)
  +--ro datastore-xpath-filter?       yang:xpath1.0 {sn:xpath}?
  
```

- "data-path" identifies data path of the targeted data object.
- "data" identifies the targeted data object that has been subscribed to.
- "xpath-expression" represents a logical expression, which can contain comparisons of datastore values and logical operations in the XPath format.
- "high-watermark" that defines the upper boundary for the targeted data object.
- "period" that defines the duration between push updates, the period can be changed based on trigger condition.
- "count" that specifies the count number of interval that has to pass before successive adaptive periodic push update records for the same subscription are generated for a receiver.

Adaptive Subscription Usage Example

Use Case: Wireless performance monitoring

```
module: example-wifi-mac
  +--rw clients
    +--ro client* [mac]
      +--ro mac          yang:mac-address
      +--ro rssi?        int8
      +--ro snr?         uint8
      +--ro ss?          uint8
      +--ro phy-rate?    uint16
      +--ro channel-support* uint8
      +--ro neighbors
        | +--ro neighbor-bssid?   yang:mac-address
        | +--ro neighbor-channel? uint8
        | +--ro neighbor-rssi?    int8
        | +--ro neighbor-antenna?  uint8
        | +--ro channel-load-report? uint8
      +--ro ssids
        +--ro name?          string
        +--ro enabled?       boolean
        +--ro broadcast-filter? boolean
        +--ro multicast-filter? boolean
        +--ro ipv6-ndp-filter? boolean
        +--ro ipv6-ndp-filter-timer? uint16
        +--ro station-isolation? boolean
```

1. Create Adaptive Subscription: Scan all clients every 5 seconds up to 30 seconds if the rssi value of client is greater than -65dB; scan all client every 60 seconds up to 360 seconds, switch to 60 seconds period value If the rssi value of client is less than -65dB, and then

```
<netconf:rpc message-id="101"
  xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
  <establish-subscription
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications"
    xmlns:yp="urn:ietf:params:xml:ns:yang:ietf-yang-push">
    <yp:datastore
      xmlns:ds="urn:ietf:params:xml:ns:yang:ietf-datastores">
      ds:running
    </yp:datastore>
    <yp:datastore-xpath-filter
      xmlns:ex="https://example.com/sample-data/1.0">
      /ex:example-wifi-mac
    </yp:datastore-xpath-filter>
    <as:adaptive-subscriptions
      xmlns="urn:ietf:params:xml:ns:yang:ietf-adaptive-subscription">
      <as:data-path>as:clients/as:client</as:data-path>
      <as:target>ssid</as:target>
      <as:adaptive-period>
        <as:condition-expression>ssid > -65</as:condition-expressioni>
        <as:watermark>-65</as:watermark>
        <as:period>5</as:period>
        <as:count>12</as:count>
      </as:adaptive-period>
      <as:adaptive-period>
        <as:condition-expression>ssid < -65</as:condition-expressioni>
        <as:watermark>-65</as:watermark>
        <as:period>60</as:period>
        <as:count>12</as:count>
      </as:adaptive-period>
    </as:adaptive-subscriptions>
  </establish-subscription>
</netconf:rpc>
```

2. Upon the server switches to from the update interval 5 seconds to the new update interval 60 seconds, sent to the receivers to inform the receivers that the update interval value is switched to the new value.

```
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2016-11-21T13:51:00Z</eventTime>
  <adaptive-update xmlns="http://example.com/ietf-adaptive-subscription">
    <id>0</id>
    <period>60</period>
    <yp:datastore
      xmlns:ds="urn:ietf:params:xml:ns:yang:ietf-datastores">
      ds:running
    </yp:datastore>
    <yp:datastore-xpath-filter
      xmlns:ex="https://example.com/sample-data/1.0">
      /ex:example-wifi-mac
    </yp:datastore-xpath-filter>
  </adaptive-update>
</notification>
```

Bulk Subscription Recap

- Motivation:
 - The subscription protocol operation doesn't provide specific criteria to classify subscriptions and therefore lacks the capability to explicitly indicate which specific subscription associated with the notification should be bundled together
 - subscription A and B are bundled based on their relationship with a set of YANG data models
 - while subscription C and D are bundled based on "transport" and "encoding" parameters
- Goal:
 - defines a YANG data model and associated mechanism that classify subscription based on various different filtering criteria
 - allow subscriber applications to bulk subscribe/unsubscribe to publisher's targeted object source based on bundle size and bundle latency.
 - allow the publishers to report multiple notification in a single bundling message defined in [I-D.ietf-netconf-notification-messages].

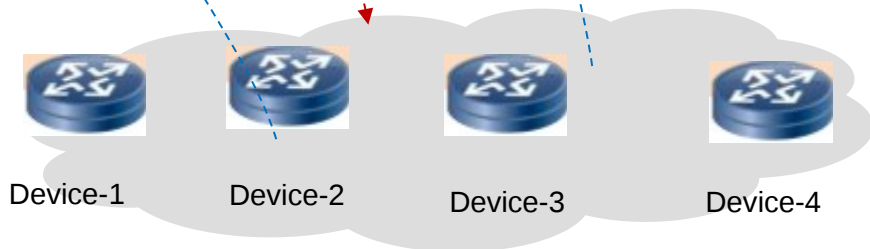
Bulk Subscription Solution Overview



a. Telemetry data Export capability Notification (Message Bundle Support)

b. Bulk Subscription to YANG Notification

c. Send bundled Message with bulk subscription extension to receiver



```

module: ietf-bulk-subscription
  +--rw bundle-groups
    +--rw bundle-group* [group-id]
      +--rw group-id      string
      +--rw subscription-id*   leafref
      +--rw yang-module*      yang:yang-identifier

  augment /sn:subscriptions/sn:subscription:
    +--rw max-bundle-size      uint32
    +--rw max-bundle-latency   uint32
    +--rw compression-algorithm string

  +---x bundle-subscription
    +---input
      +---w group-id?          -> /bundle-groups/bundle-group/group-id
      +--rw max-bundle-size      uint32
      +---w max-bundle-latency   uint32
      +---w compression-algorithm string
      +---w subscription-id*     subscription-id
      +---w masked-subscription-id* subscription-id

module: ietf-bulk-notification
  augment-structure /nm:message/nm:message-header:
    +--rw group-id?      string
    +--rw compression-algorithm string
  
```

- Augment ietf-subscribed-notifications module with bulk subscription attributes
- Define bundle subscription RPC to specify which subscriptions can be bundled and which not based on several subscription criteria such as encoding, transport
- Additional subscription criteria:
 - Max-bundle-latency: The maximum latency before a specific YANG Notifications generated must egress a publisher. This attribute enhances QoS feature and provide additional subscription bundle classification criteria.
 - Compression-algorithm: The technology with which an originator compress byte stream contents. This attribute enhances QoS feature and provide additional subscription bundle classification criteria.
- Subscription-id: indicates what subscription must be bundled together.
- Masked-subscription-id: indicates what subscription must not be bundled together.

Next Steps

- Key value of adaptive subscription:
 - Address sampling performance bottleneck on the device when facing Massive Data Collection and Processing
 - Greatly reduce the amount of data to be exported
- Key value of bulk subscription:
 - Improves data collection efficiency and performance.
 - Work together with message bundling defined in [I-D. ietf-netconf-notification-messages]
- Request adoption call on two drafts?
- Address any comments received in the meeting.