Bulk Subscription to YANG Event Notification

draft-wang-netconf-bulk-subscribed-notifications-00

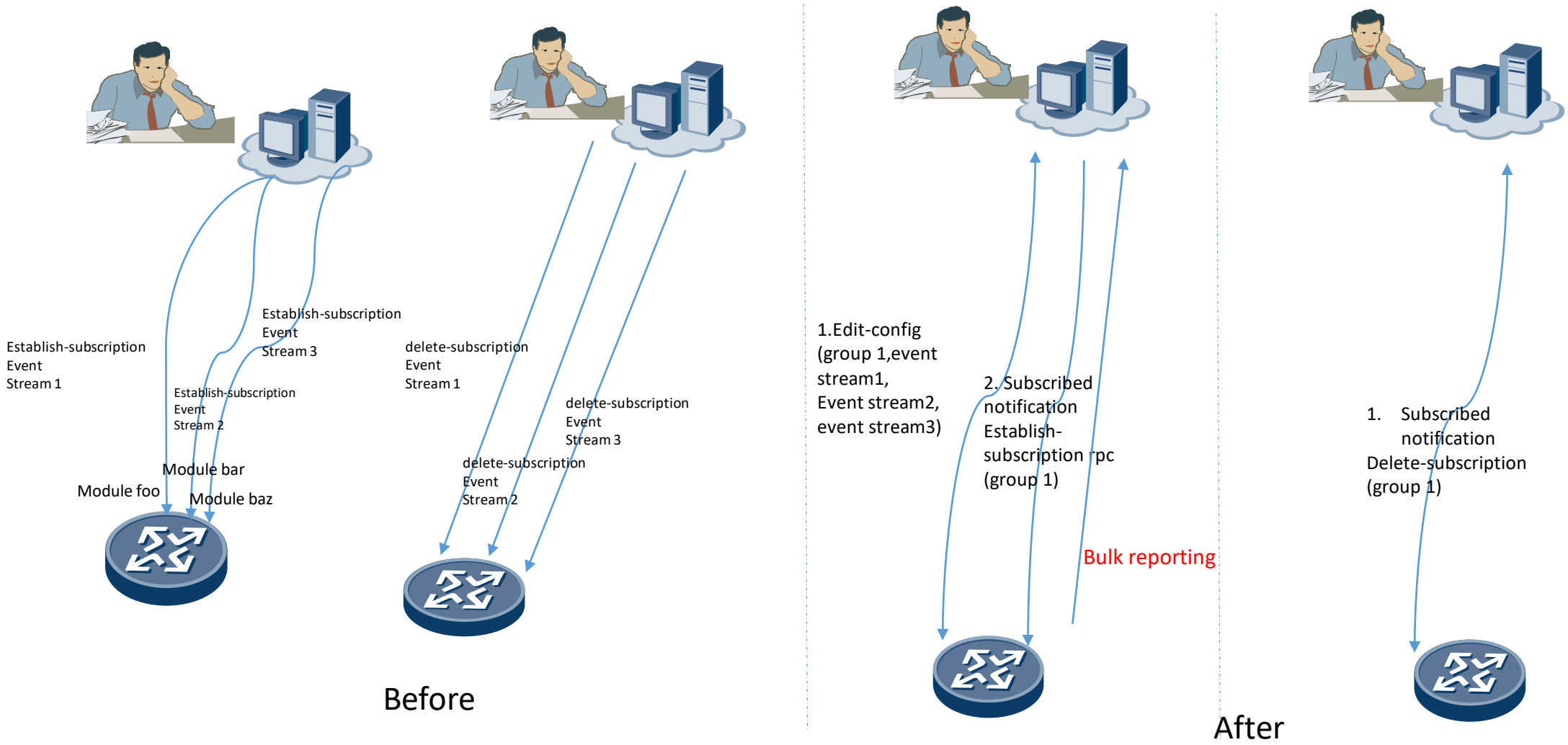
Michael Wang (wangzitao@huawei.com)

Qin Wu (bill.wu@huawei.com)

Why this draft?

- Now,
  - The Subscribed Notification Draft has been published as RFC8639.
    - It allows users to subscribe to the event streams that they are interested in.
  - Notification Message Headers and Bundles [I-D. ietf- netconf-notification-messages] uses subscription-id for identifying the targeted subscription.
  - However In some other cases, the users may want to subscribe to multiple event streams for one service
- But,
  - The existing mechanism lacks the capabilities
    - to identify a set of event streams that have a common characteristic, &
    - to perform protocol operation on a set of event stream via a single transaction...

Use Case: Bulk subscription lifecycle management



- Establishing a Dynamic Subscription (event stream could be pointing to the data object from each source, or data object for specific period from each source)
- Delete a Dynamic Subscription
- Bulk reporting (shown in the right figure)

What does this draft do?

I want to subscribe:
- Foo;
- Bar;
- Baz

- Introduce a **group identifier to provide** a more optimal mechanism for protocol operation
  - which would otherwise require multiple atomic transactions on a per event stream basis;

- Defines a YANG data model and associated mechanism that allows
  - subscriber to bulk subscribe to publishers' event streams based on their requirements.
  - publishers to report multiple event streams or subscriptions into a single notification message based on group identifier affiliation.

Before:

```
<rpc message-id="102">
  <establish-subscription>
    <stream-xpath-filter xmlns:ex="http://example.com/events">
      /foo/
    </stream-xpath-filter>
    ....
  </establish-subscription>
</rpc>
<rpc-reply message-id="102">
 <id>22</id>
</rpc-reply>

<rpc message-id="103">
  <establish-subscription>
    <stream-xpath-filter xmlns:ex="http://example.com/events">
      /bar/
    </stream-xpath-filter>
    ....
  </establish-subscription>
</rpc>
<rpc-reply message-id="103">
 <id>23</id>
</rpc-reply>

<rpc message-id="104">
  <establish-subscription>
    <stream-xpath-filter xmlns:ex="http://example.com/events">
      /baz/
    </stream-xpath-filter>
    ....
  </establish-subscription>
</rpc>
<rpc-reply message-id="104">
 <id>24</id>
</rpc-reply>
```

After:

```
<groups>
 <group>
  <group-id>10</group-id>
  <stream>/foo/</stream>
  <stream>/bar/</stream>
  <stream>/baz/</stream>
 </group>
</groups>

<rpc message-id="102">
  <establish-subscription>
   <group-id>10</group-id>
    ....
  </establish-subscription>
</rpc>

<rpc-reply message-id="102">
 <id>23</id>
</rpc-reply>
```

Solution Overview

```
module: ietf-bulk-subscription
  +--rw groups
     +--rw group* [group-id]
        +--rw group-id    string
        +--rw stream*     stream-target

  augment /sn:subscriptions/sn:subscription/sn:target:
    +--:(stream-group)
       +--rw group-id?   -> /groups/group/group-id
  augment /sn:establish-subscription/sn:input/sn:target:
    +--:(stream-group)
       +-- group-id?   -> /groups/group/group-id
```

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

Next Steps

- Key value:
  - The group identifier proposed in the solution provides a more optimal mechanism for protocol operation
  - The mechanism we proposed can reduce round-trips of requests and responses and save bandwidth on the interface between NMS and network device.
  - Provide multi-source aggregation
    - e.g., get ingress packet loss in each member link and aggregated in the trunk interface.
    - E.g., aggregate data for a specific period time for each source and then aggregate data from each source.
- Do we agree this is a problem that need to be solved and document as RFC?
- Solicit more comments
  - Your comments and suggestions are welcome!