IETF 95 ROLL

Routing over Low-Power And Lossy Networks

Chairs:
Peter van der Stok
Ines Robles

Secretary:
Michael Richardson

Note Well

Any submission to the IETF intended by the Contributor for publication as all or part of an IETF Internet-Draft or RFC and any statement made within the context of an IETF activity is considered an "IETF Contribution". Such statements include oral statements in IETF sessions, as well as written and electronic communications made at any time or place, which are addressed to:

- The IETF plenary session
- The IESG, or any member thereof on behalf of the IESG
- Any IETF mailing list, including the IETF list itself, any working group or design team list, or any other list functioning under IETF auspices
- Any IETF working group or portion thereof
- Any Birds of a Feather (BOF) session
- The IAB or any member thereof on behalf of the IAB
- The RFC Editor or the Internet-Drafts function

All IETF Contributions are subject to the rules of RFC 5378 and RFC 3979 (updated by RFC 4879).

Statements made outside of an IETF session, mailing list or other function, that are clearly not intended to be input to an IETF activity, group or function, are not IETF Contributions in the context of this notice. Please consult RFC 5378 and RFC 3979 for details.

A participant in any IETF activity is deemed to accept all IETF rules of process, as documented in Best Current Practices RFCs and IESG Statements.

A participant in any IETF activity acknowledges that written, audio and video records of meetings may be made and may be available to the public.

Source: https://www.ietf.org/about/note-well.html

Thank you very much to Michael Richardson for his excellent Job as co-chair :-)

Welcome Peter as now co-chair :-)

Meeting Materials

- Remote Participation
  - Jabber Room: roll@jabber.ietf.org
  - Meetecho: http://www.meetecho.com/ietf95/roll
- Etherpad:
  - http://tools.ietf.org/wg/roll/minutes
- Audio Streaming:
- Minutes taker:
- Jabber Scribe:
- **Please sign blue sheets :-)**

Agenda

- State of work items, ROLL I-D, Related I-D and Open Issues (5 min.)
- draft-ietf-roll-useofrplinfo (5 min)
- draft-ietf-roll-routing-dispatch (15 min)
- draft-jadhav-roll-no-path-dao-ps (15) - remotely
- draft-vanderstok-roll-mpl-yang (5 min)
- ROLL Charter discussion (10 min)
- Open Floor (5 min)

Milestones

Milestone	Schedule
Submit draft about when to use RFC 6553, RFC 6554, and IPv6-in-IPv6 encapsulation to the IESG. **[draft-ietf-roll-useofrplinfo-03]**	May 2016
Submit draft about how to compress RFC 6553, RFC 6554, and IP headers in the 6LoWPAN adaptation layer context to the IESG. **[draft-ietf-roll-routing-dispatch-00]**	April 2016
Evaluate WG progress, recharter or close	April 2016

State of Active Internet-Drafts

draft-ietf-roll-admin-local-policy-03	New! RFC 7732
draft-ietf-roll-applicability-ami-11	New version being worked on (Thank you Nancy!)
draft-ietf-roll-applicability-home-building-12	New! RFC 7733
draft-ietf-roll-applicability-template-07	Stable - not to be published
draft-ietf-roll-trickle-mcast-12	New! RFC 7731
draft-ietf-roll-mpl-parameter-configuration-08	New! RFC 7774
draft-ietf-roll-useofrplinfo-03	**New revision needed, discussion needed**
draft-ietf-6lo-routing-dispatch-05	**Moved from 6lo to ROLL.**

Related Internet-Drafts

draft-thubert-roll-dao-projection-02	Root initiated routing state in RPL	Future Discussion?
draft-jadhav-roll-no-path-dao-ps	No-Path DAO Problem Statement	Slides Today
draft-vanderstok-roll-mpl-yang-00	A YANG model for Multicast Protocol for Low power and lossy Networks (MPL)	Slides Today
draft-wang-roll-adaptive-data-aggregation	Design of Adaptive Data Aggregation Schemes	Future Discussion?
draft-zhong-roll-dis-modifications-00	DIS Modifications	Future Discussion?

Open Tickets

Ticket	Summary
#169	Work Item Proposals
#170	Use of ESC Dispatch value in new IETF header compression

When to use RFC 6553, 6554 and IPv6-in-IPv6

draft-ietf-roll-useofrpi

Michael Richardson
Pascal Thubert
Ines Robles

Cases

- Flow from RPL-aware-leaf to root
- Flow from root to RPL-aware-leaf
- Flow from non-RPL-aware-leaf to root
- Flow from root to non-RPL-aware-leaf
- Flow from RPL-aware-leaf to Internet
- Flow from Internet to RPL-aware-leaf
- Flow from non-RPL-aware-leaf to Internet
- Flow from Internet to non-RPL-aware-leaf
- Flow from RPL-aware-leaf to RPL-aware-leaf
- Flow from RPL-aware-leaf to non-RPL-aware-leaf
- Flow from non-RPL-aware-leaf to RPL-aware-leaf
- Flow from non-RPL-aware-leaf to non-RPL-aware-leaf

STORING & NON-STORING

Storing,Non-Storing} X {RPL-aware-leaf,non-RPL-aware,root, Internet} X {RPL-aware-leaf,non-RPL-aware,root,Internet}

Rules for the Proposed Scenarios

-This document assumes a rule that a **Header cannot be inserted or removed on the fly inside an IPv6 packet that is being routed**.

- This means that an **intermediate router that needs to add a header must encapsulate the packet in an outer IP header where the new header can be placed.**

- This also means that a Header can only be removed by an intermediate router if

- it is placed in an encapsulating IPv6 Header,
- and that the IPv6 header is *addressed* to that intermediate router!

The whole encapsulating header must be removed - a replacement may be added though.

Headers needed in Storing mode: RPI, RH3, IP-in-IP encapsulation

Use Case	RPI	RH3	IPv6-in-IPv6	IPIP dst
RAF to root	Yes	No	No	----
Root to RAF	Yes	No	No	----
root to ~RAF	Yes	No	Yes	hop
~RAF to root	Yes	No	Yes	root
RAF to Internet	Yes	No	Yes	root
Internet to RAF	Yes	No	Yes	RAF
~RAF to Internet	Yes	No	Yes	root
Internet to ~RAF	Yes	No	Yes	hop
RAF to RAF	Yes	No	No	--
RAF to ~RAF	Yes	No	Yes	hop
~RAF to RAF	Yes	No	Yes	RAF
~RAF to ~RAF	Yes	No	Yes	hop

RAF: RPL-aware-Leaf

~RAF: non-RPL-aware-Leaf

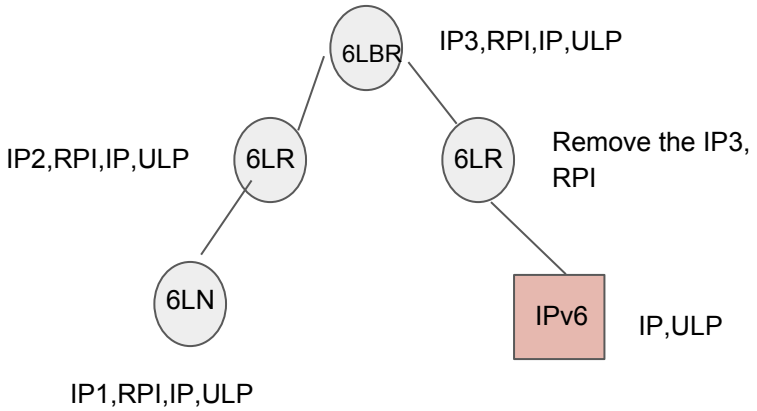
Headers needed in Non-Storing mode: RPI, RH3, IP-in-IP encapsulation

Use Case	RPI	RH3	IPv6-in-IPv6	IPIP dst
RAF to root	Yes	No	No	---
Root to RAF	Yes	Yes	No	---
root to ~RAF	No	Yes	Yes	6LR
~RAF to root	Yes	No	Yes	root
RAF to Internet	Yes	No	Yes	root
Internet to RAF	opt	Yes	Yes	RAF
~RAF to Internet	Yes	No	Yes	root
Internet to ~RAF	opt	Yes	Yes	6LR
RAF to RAF	Yes	Yes	Yes	root/RAF
RAF to ~RAF	Yes	Yes	Yes	root/6LR
~RAF to RAF	Yes	Yes	Yes	root/RAF
~RAF to ~RAF	Yes	Yes	Yes	root/6LR

RAF: RPL-aware-Leaf

~RAF: non-RPL-aware-Leaf

E.G. Flow from RPL-aware-leaf to non-RPL-aware-leaf (Storing)



Somehow, the sender has to know that the receiver is not RPL aware

In the completely general storing case, which includes not-RPL aware leaf nodes, it is not possible for a sending node to know if the destination is RPL aware, and therefore it must always use hop-by-hop IPIP encapsulation, and it can never omit the IPIP encapsulation.

6LoRH Compression cases

The [I-D.ietf-6lo-routing-dispatch] proposes a compression method for RPI, RH3 and IPv6-in-IPv6.

The uses cases mentioned in this draft MUST use 6LoRH. Examples of the use of 6LoRH are found in Apendix A of [I-D.ietf-6lo-routing-dispatch].

We need more comments on the ML Please :-)

Thank you! :)

A Routing Header Dispatch for 6LoWPAN

draft-thubert-roll-routing-dispatch

Pascal Thubert, Carsten Bormann, Robert Cragie, Laurent Toutain

IETF 95

Buenos Aires, April 2016

What's new Since IETF 94?

- Detailed the coalescence process
- RH3-6LoRH renamed to SRH-6LoRH
- Validation at ETSI plugtest
- New draft as a result
  - draft-thubert-6lo-inner-compression
  - Updates RFC 6282
  - RFC 6282 stateless => LL prefix FE80::
  - Proposal to get prefix from outer header
  - Also from 6LoRH even if no IP in IP

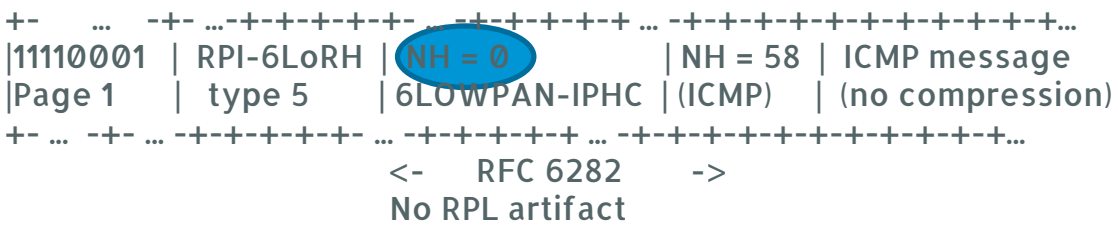
Rolling docs

- Separated draft-ietf-6lo-paging-dispatch
- Paging Dispatch remains at 6lo, now in last call
- Routing Dispatch moved to ROLL, now in last call

Issues / questions

- Better compression for various length
  - Current: only powers of 2
- RPI-6LoRH
  - Generic Name in RFC6550
  - Expands into RPL opt in HbH
  - Should it be named more specifically ?
- RH3-6LoRH
  - Specific Name inherited from RFC6554
  - But called SRH in that spec, not specific
  - Compressed format valid for other RH types
  - Should it be named less specifically ?

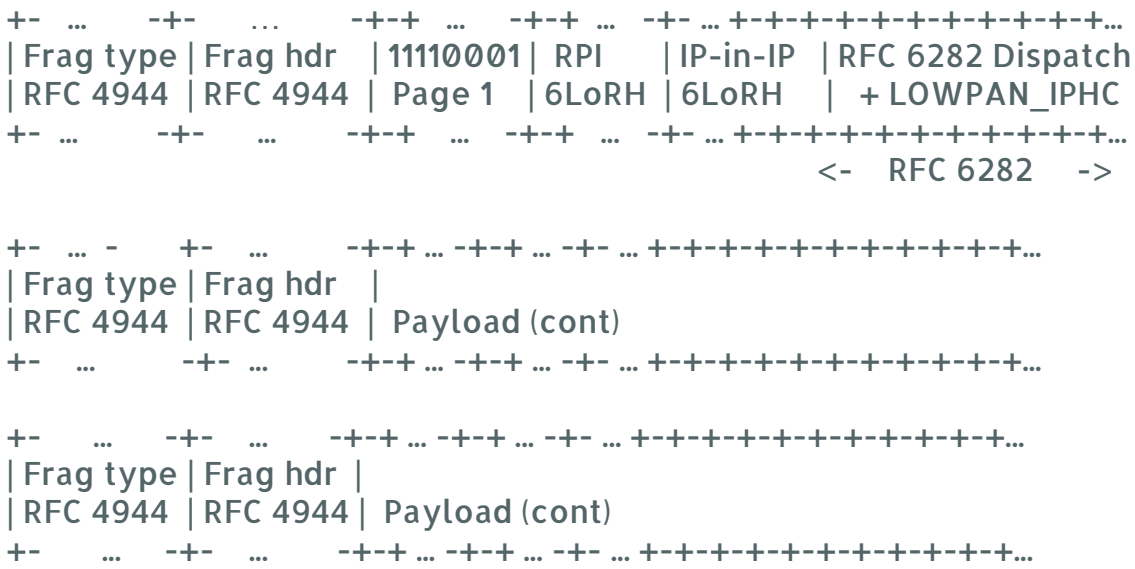
e.g. 6LoRH – RPI only, ICMP



<=>

With inner-compression:
LOWPAN_IPHC stateless based on root prefix for source and destination

e.g. Fragmented 6LoRH – IP-in-IP + RPI



<=>

With inner-compression:
LOWPAN_IPHC stateless based on outer packet source and destination

Final order

The original proposal MAC RH3-6LoRH* RPI-6LoRH IP-in-IP-LoRH IPHC blah Works better

Reason 1: We modify the RH3-6LoRH on the way, popping the first address as we go. It is easier to do if it is the first header of the compressed packet so we always play with the very beginning of the packet

Reason 2: So that IP header always TERMINATES the 6LoRH encapsulation,

When there is no IP in IP , this is already true for instance MAC RPI-6LoRH IPHC

One needs to differentiate a case that in UNCOMPRESSED form is

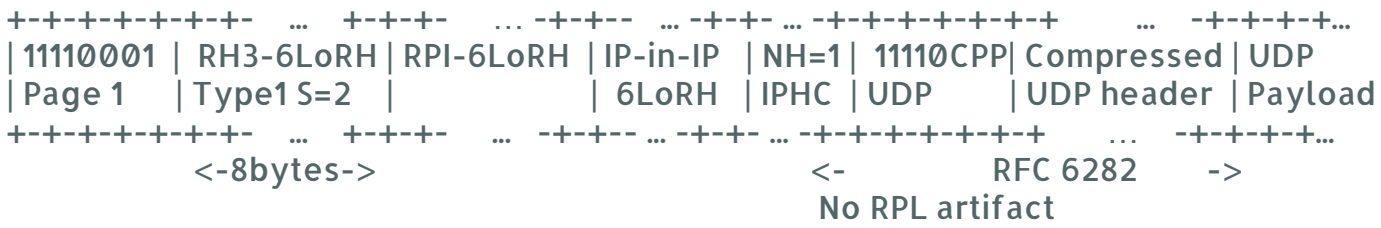
IP-in-IP RPI RH3 IP blah vs. **IP-in-IP IP RPI RH3 blah**

With a format like **MAC IP-in-IP-LoRH RH3-6LoRH* RPI-6LoRH IPHC blah** You cannot tell : (

With this format we have a clear separation for IP in IP in IP all the way

The separation of which header is in which encaps is clearly delineation with the IP header that terminates the encapsulated 6LoRH-headers.

UDP packet forwarded by the root



One may note that the RPI is provided. This is because the address of the root that is the source of the IP-in-IP header is elided and inferred from the InstanceID in the RPI. Once found from a local context, that address is used as Compression Reference to expand addresses in the RH3-6LoRH.

<=>

With inner-compression:
LOWPAN_IPHC stateless based on outer packet source from header source (itself from root) Last Hop (Dest) from last RH3-6LoRH entry

Calls

Last call started 23 March 2016

6LoRH also in last call at ROLL

Michael Richardson Shepherd

No-Path DAO of RPL

Problem Statement

https://tools.ietf.org/html/draft-jadhav-roll-no-path-dao-ps-00

Rahul, Rabi, Zhen@ Huawei

Hui Deng @China Mobile

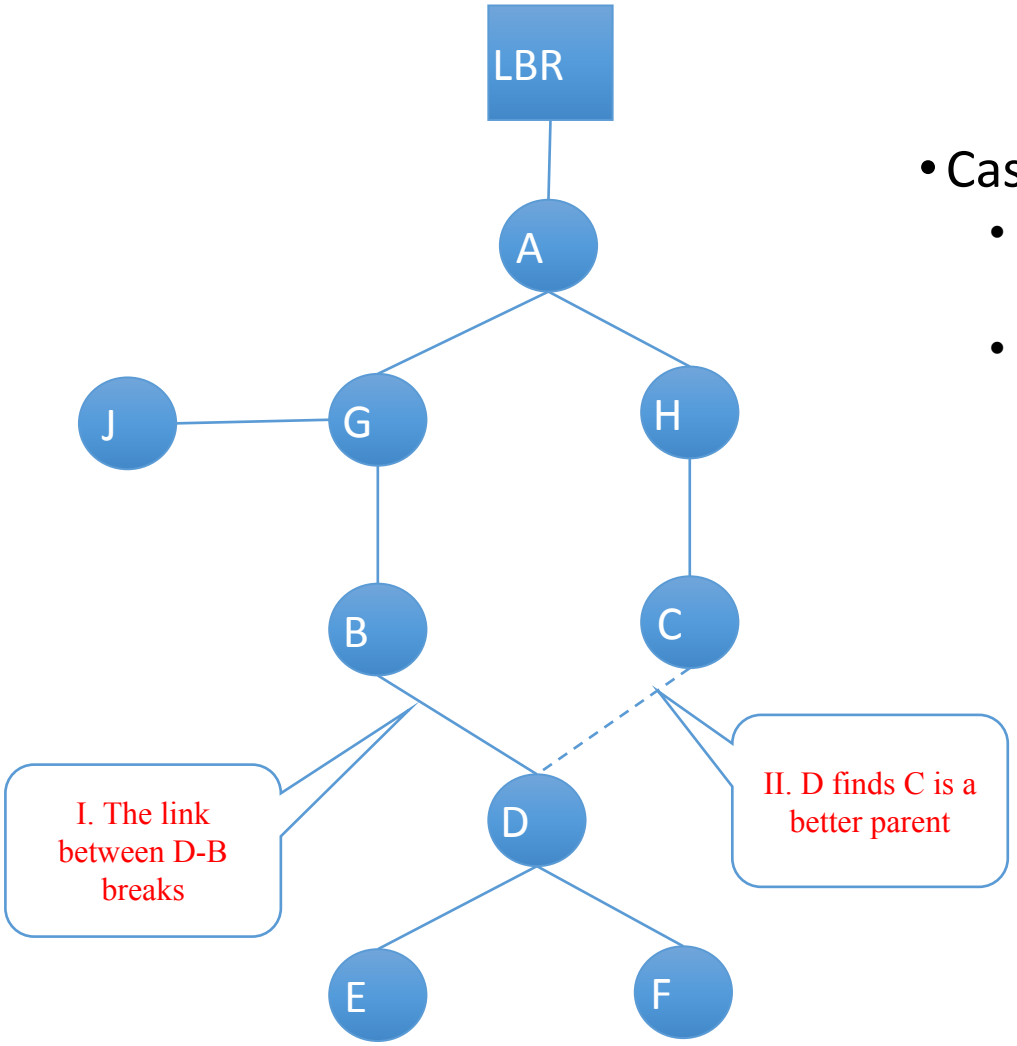
IETF95

No-Path DAO is important to handle network dynamics

- **NPDAO Recap**
  - NPDAO = DAO (lifetime:=0)
  - Used for route invalidation
    - Release resources (for e.g. routing entries) along the previous path
  - Traverses upwards along the path from previous best parent towards the sink

- **Why NPDAO is important?**
  - Routing entries are the biggest memory-hogging component (especially in bigger storing-mode RPL networks)
    - In case of contention, its better to know which entries are non-active.
  - When a node switches parent, the sub-tree rooted at that node switches. Thus a high possibility of invalid route entries.
  - Impacts P2P traffic

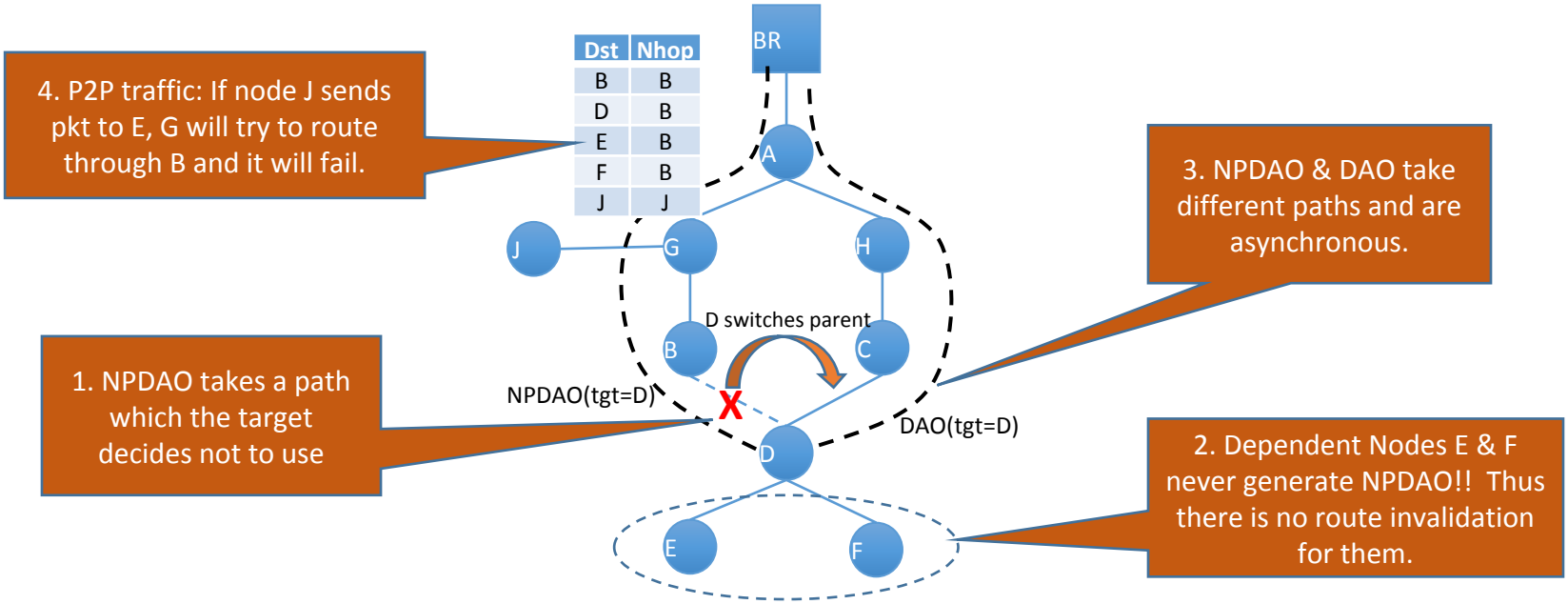
Sample Topology & NPDAO scenarios



- Cases in which NPDAO is generated
  - Intermittent or permanent link failure
    - I. The link between D-B breaks
  - Node finds a better metric parent and decides to switch
    - II. D finds C is a better parent

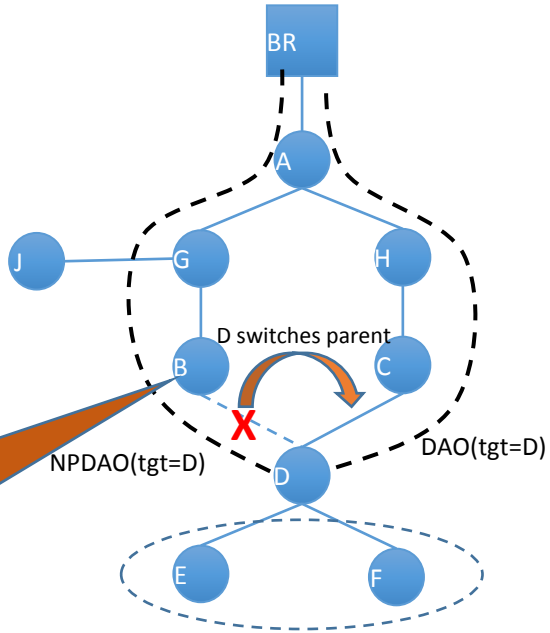
Problems with NPDAO

- NPDAO messaging depends on previous link which the node decided to no longer use
- No route invalidation for dependent nodes
- Possibility of Route downtime caused by asynchronous operation of NPDAO and DAO.
  - If NPDAO reaches before DAO, then the route will be unavailable till the time DAO reaches the all common parent nodes (A & BR in the example below).
- Impact on P2P traffic because of NPDAO inefficiency



Possible existing solutions and corresponding problems…

- It may be possible for the parent to detect child unavailability*
  - Problem is parent cannot act unilaterally based on this info
  - On error detection, RFC6550 section 11.2.2.3 mentions parent can send "a packet" to clear the RPL states*… The provisions are vaguely stated…
    - The primary problem is that DAO has state information in terms of DAOSequence and PathSequence. An NPDAO needs to adhere to this state info. Thus it becomes difficult for any other node to generate DAO on behalf of target.



Even if B detects link failure to D, it cannot invalidate route along the path. At best it can only clear local states.

Section 11.2.2.3 states:
"With DAO inconsistency loop recovery, a packet can be used to recursively explore and clean up the obsolete DAO states along a sub-DODAG."

[*] Thanks to Cenk for pointing this out

Handling intermittent link failures*

- DAO has an optional ACK mechanism
- A node generating an NPDAO would know whether the NPDAO was received successfully by the previous parent
- At a later point, the node could retry NPDAO on realizing that it had regained the link

[*] Thanks to Cenk & Simon for pointing this out

Requirements for NPDAO improvements

- Should be tolerant to link failures to previous parent
- Should be possible to invalidate routes for dependent nodes as well
- Avoid route downtime because of NPDAO, DAO operation

Summary of Mailing list discussion up to now

- Q [**Cenk**]: Widen the scope of draft to include DAO-DAOACK scenarios?
- A:Yes, this scenarios definitely needs to be incorporated… [details on slide 6]

- Q [**Cenk**]: Can an implementation respond in case the parent node learns that the child node is no longer available? Can it try to clear the routing state along the old path? RFC 6550 section-11.2.2.3 talks about clearing RPL states.
- A: The statement in the RFC is not concisely defined and it is non-trivial to clear states as defined in the section-11.2.2.3. [Details on slide 5]

- Q [**Simon**]: In case of transient failures, DAOACK failure knowledge could be used by the child node and could retry NPDAO at later point on link resumption?
- A: Yes, this solution caters to the specific transient failure scenario. We will add this point to the draft. [Details on slide 6]

- Q [**Cenk**]: 6Lo Neighbor Discovery can detect next-hop child node unavailability and clear RPL states
- A: 6Lo ND will help in detecting next-hop unavailability but on detection the parent node cannot clear the RPL states along the route [for reasons mentioned on slide 5]

Credits:
Cenk Gündogan cnkgndgn@gmail.com
Simon Duquennoy simonduq@sics.se

Next Step

- Shall we work on this problem within ROLL WG?
- WG Adoption?

Thank you

ROLL working group

YANG MPL model
draft-vanderstok-roll-mpl-yang-00

P. van der Stok, P. Thubert

Domain spans MC addresses
MC addresses are assigned to interfaces

```
+--rw mpl-domain
   +--rw domains*  [domainID]
      +--rw domainID                uint16
      +--rw MClist*                 yang:ipv6-address
   +--rw addresses* [MCaddress]
      +--rw MCaddress               yang:ipv6-adddress
      +--rw interfaces*             string
```

* : list
[key] : key attribute(s) of list

Operational parameters
Per domain assign MPL parameters

```
+--rw mpl-op
   +--rw SE_LIFETIME                              uint16
   +--rw PROACTIVE_FORWARDING                     boolean
   +--rw SEED_SET_ENTRY_LIFETIME                  uint64
   +--rw mpl-parameter* [domainID]
      +--rw domainID                              uint16
      +--rw DATA_MESSAGE_IMIN                     uint16
      +--rw DATA_MESSAGE_IMAX                     uint16
      +--rw DATA_MESSAGE_K                        uint16
      +--rw DATA_MESSAGE_TIMER_EXPIRATIONS        uint16
      +--rw CONTROL_MESSAGE_IMIN                  uint16
      +--rw CONTROL_MESSAGE_IMAX                  uint16
      +--rw CONTROL_MESSAGE_K                     uint16
      +--rw CONTROL_MESSAGE_TIMER_EXPIRATIONS     uint16
      +--rw MC_adddress*                          yang:ipv6-adddress
```

Operational statistics
Per domain and seed progress in packets

```
+--ro mpl-seeds* [seedID, domainID]
   +--ro seedID                          uint64
   +--ro domainID                        uint16
   +--ro local                           boolean
   +--ro life-time                       uint64
   +--ro min-seqno                       uint8
   +--ro data-number                     uint8
   +--ro control-number                  uint8
   +--ro nr-of-timers                    uint8
   +--ro seed_timers* [seqno]
      +--ro seqno                        uint8
      +--ro I                            uint8
      +--ro c                            uint8
      +--ro e                            uint8
      +--ro t                            uint8
```

Operational statistics
Per domain and seed, MPL counters

```
+--ro mpl-statistics* [seedID, domainID]
          +--ro seedID                          uint64
          +--ro domainID                        uint16
          +--ro c-too-high                      uint64
          +--ro nr-forwarded                    uint64
          +--ro nr-not-forwarded                uint64
          +--ro nr-of-messages-received         uint64
          +--ro nr-of-copies-received           uint64
          +--ro nr-of-messages-forwarded        uint64
          +--ro nr-of-copies-forwarded          uint64
          +--ro nr-of-refused                   uint64
        +--ro nr-of-notreceived                 uint64
          +--ro nr-of-missing                   uint64
          +--ro nr-of-inconsistent-data         uint64
          +--ro nr-of-consistent-data           uint64
          +--ro nr-of-inconsistent-control      uint64
          +--ro nr-of-consistent-control        uint64
```

ROLL working group

ROLL charter discussion

The Working Group is focused on routing issues for LLN.

Current items

- When to use RFC6553, RFC6554, and IPv6-in-IPv6 encapsulation.
- How to compress RFC6553, RFC6554, and IP headers in the 6LoWPAN adaptation layer context

The Working Group is focused on routing issues for LLN.

Proposed Items for re-charter?

- Maintenance of RPL and protocols developed by ROLL
  - e.g. No-Path DAO Problem Statement (draft-jadhav-roll-no-path-dao-ps)
  - DIS Modifications (draft-zhong-roll-dis-modifications)
  - Work on mixed storing and non-storing (draft-pthubert-roll-dao-projection)
- How to reduce paths for RPL in non-storing mode
- Automatic selection of MPL forwarders to reduce message replication
- Data models for RPL and MPL management
  - E.g. A YANG model for Multicast Protocol for Low power and lossy Networks (MPL) (draft-vanderstok-roll-mpl-yang)
- Source-Routed Multicast for RPL (draft-bergmann-bier-ccast).

Open Floor

Additional Slides

The Working Group is focused on routing issues for LLN.

Proposed Items for re-charter? (Part II)

Ticket 169:

- A MOP based on BIER and/or Bloom filters, which is a way to compress BIER with a chance of false positive.
- The capability to kill a route from the root, e.g. if a DAO state is installed and should be cleaned up, e.g. if the root realizes that the node is no more there or there is a duplication or what-else.
- The capability to trigger a DAO from the root if a target is expected to be present in the network but there is no DAO state (maybe to save resources)
- The capability to request the triggering of a new iteration of a DODAG from a RPL node or a controlling element.