

# Root-ACK

- draft-jadhav-roll-storing-rootack-01

# Motivation

- End to end path establishment indication
  - Node can initiate app traffic on this indication
  - Section 4 of RPL-Observations draft details the problem stmt
- For RUL-scenario to send NA in response to e2e path establishment

Non-Storing MOP

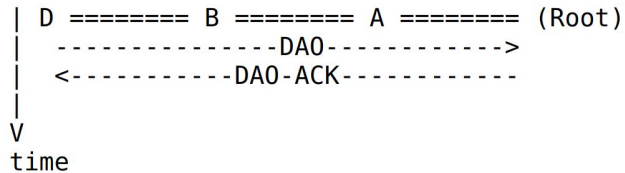


Figure 2: NS-MOP DAO/DAO-ACK handling

Storing MOP

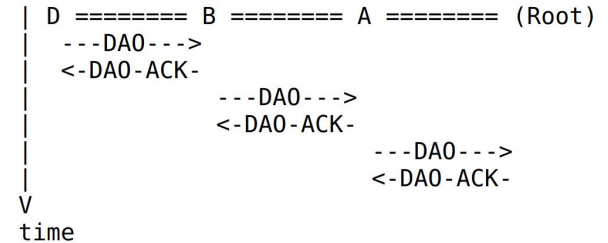


Figure 3: Storing MOP DAO/DAO-ACK handling

# Basic Operation

- RootACK sent directly from the root to the Target
- K-flag in TIO to indicate root to send RootACK
- PathSeq is used to tally RootACK to DAO

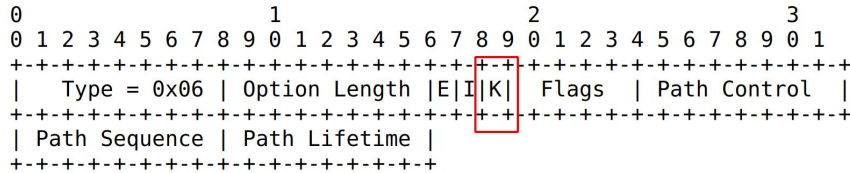
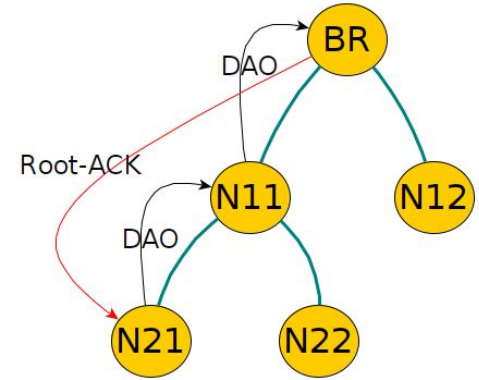
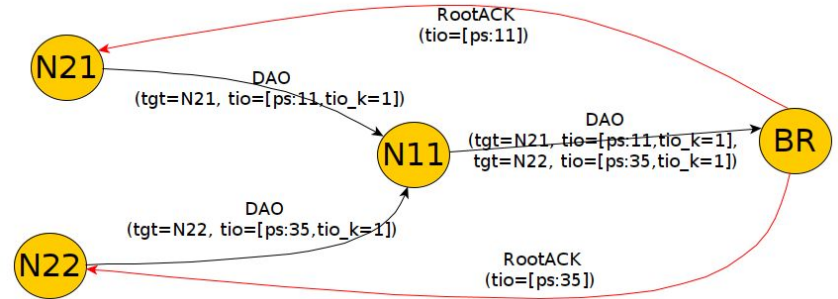


Figure 4: Updated Transit Information Option (New K flag added)

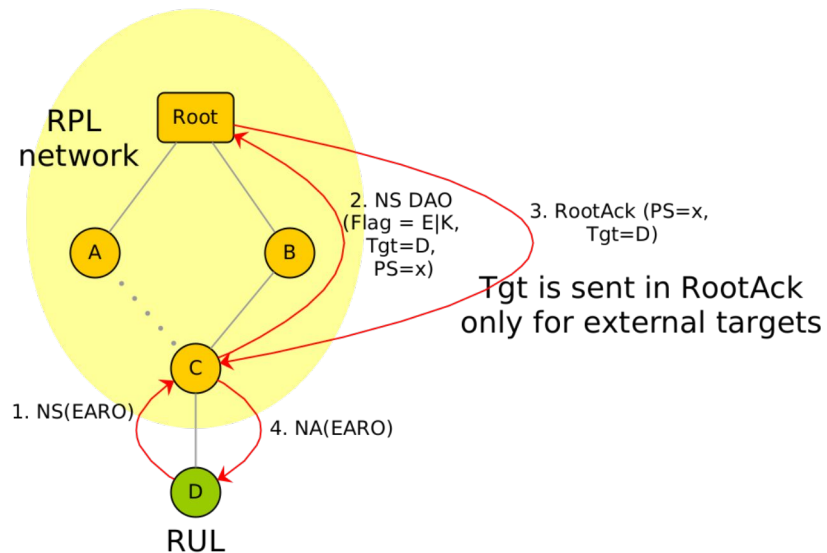


# K flag in TIO

- K flag is set by the target in the TIO
  - Used by the root to send the RootACK
  - Target may set the K flag only once after startup
  - RootAck may be sent asynchronously by the root
    - Useful for CAP query
- Intermediate 6LRs K-flag handling
  - DAO is regenerated on 6LRs on behalf of target node
  - K flag has to be stored in context to the target. Similar to E-flag.
  - When the intermediate nodes see the K flag disabled from the target the K flag could be reset

# RULs with RootACK

- Send NA to RUL only when e2e path is estd
  - Send NA in response to RootACK
- For RUL targets, the 6LR sends DAO directly to the root even in storing MOP
  - As specified in unaware-leaves



# Updates in the last version

- Calling RootACK consistently in the document
- Implications of DelayDAO
- Explicit section for RULs

## Next Steps:

- Reviews
- Adoption?