IoT Workshop
RPL Tutorial

April 2011

JP Vasseur (jpv@cisco.com)

Cisco Distinguished Engineer

Co-Chair of the IETF ROLL Working Group

Chair of the Technology Advisory Board – IPSO alliance, IEEE P1901.2 IP sub-group

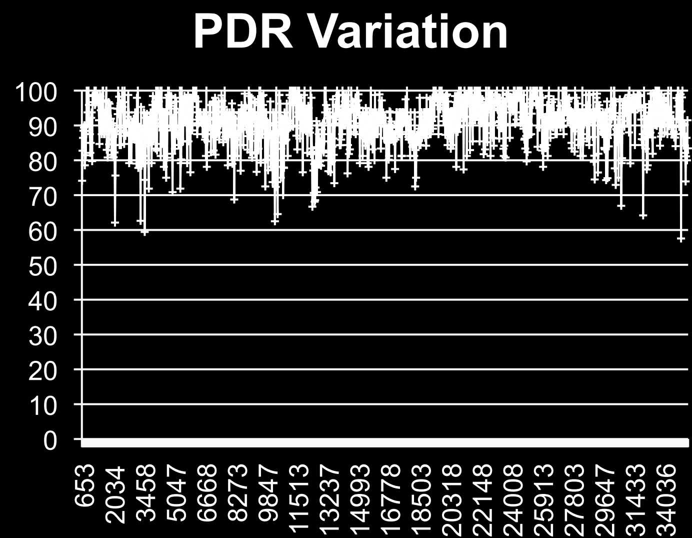
Where should Routing Take Place ?

- Historically, a number of interesting research initiatives on routing in WSN,
- Main focus on algorithms … a bit less on architecture
- Most work assuming the use of MAC addresses – L2 “routing” (mesh-under)
- Support of multiple PHY/MAC is a MUST: IEEE 802.15.4, LP Wifi, PLC (number of flavors), …
- Layered architecture supporting multiple PHY/MAC, there aren’t that many options … IP !

See the position paper on the “mesh under versus route over” debate – IETF ID to be published soon

Routing Technical challenges in LLNs

- Energy consumption is a major issue (for battery powered sensors/controllers),
- Limited processing power
- Very dynamic topologies:
  - Link failure (LP RF)
  - Node failures (triggered or non triggered)
  - Node mobility (in some environments),



- Data processing usually required on the node itself,
- Sometimes deployed in harsh environments (e.g. Industrial),
- Potentially deployed at very large scale,
- Must be self-managed (auto-discovery, self-organizing networks)

“Classic” IP Networks are different

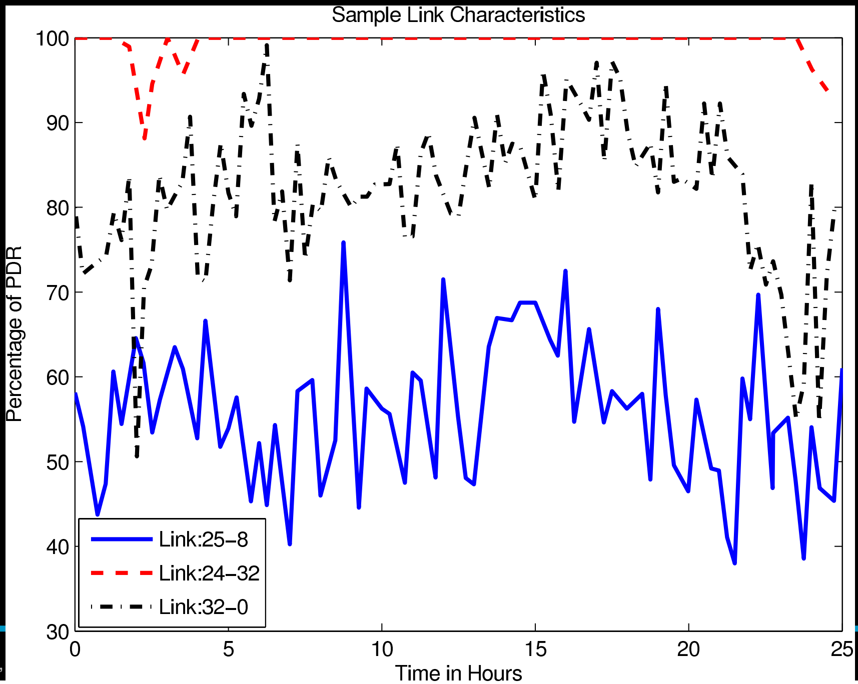
- Routing protocols used in Service Providers’ network are link state
- Scalability is a must but clearly not the same order of magnitude (most ISIS network are L2 flat)
- Convergence time in these networks is key: ~ 10s of ms but link/node characteristics are quite different
  - Low BER
  - Immediate triggering (Link layer trigger or Fast KA (BFD)
  - Use of pre-configured backup path with FRR (IP/MPLS)
  - Use of dampening in case of rare link flaps
- Mix of protection and restoration approach
- No need for node metrics/constraints

Lossy links are not just LP wireless …

- PLC also quite challenging
- Impedance variations, noise floor, K factor, …
- Both LP wireless and PLC:
  - Very hard to model link behavior (even with G/G/K Markov Chain, no M and no D …)
  - Only valid model is real-data link profile from deployed networks

How loosy is lossy ?

- Note just an increased BER
- Strong instabilities … that should be locally handled
- Fast global convergence via restoration in LLN would ineluctably lead to routing oscillation
- Under-react should be the rule …



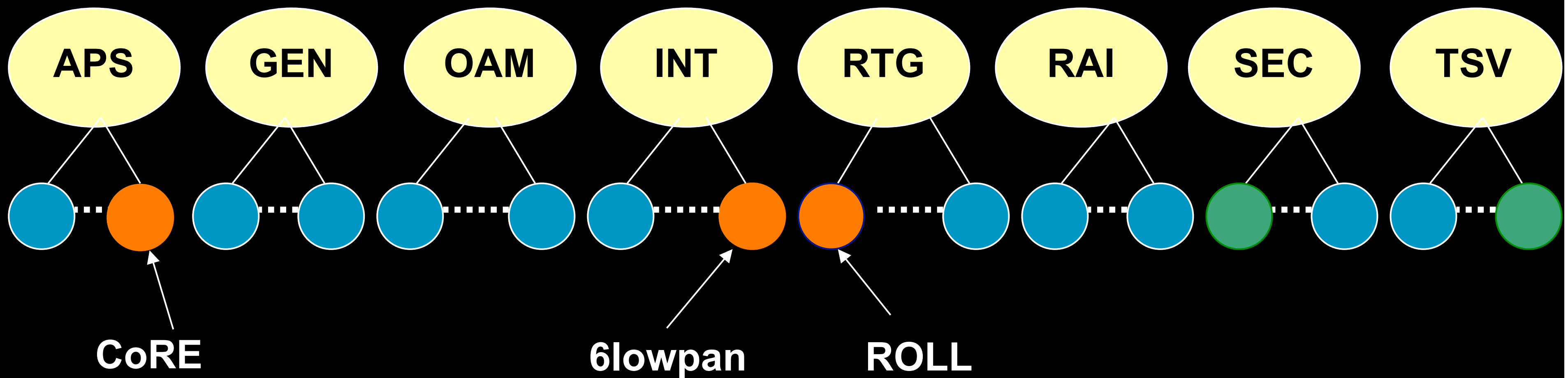
IETF – Routing Protocols

Long history in developing routing protocols at the IETF:

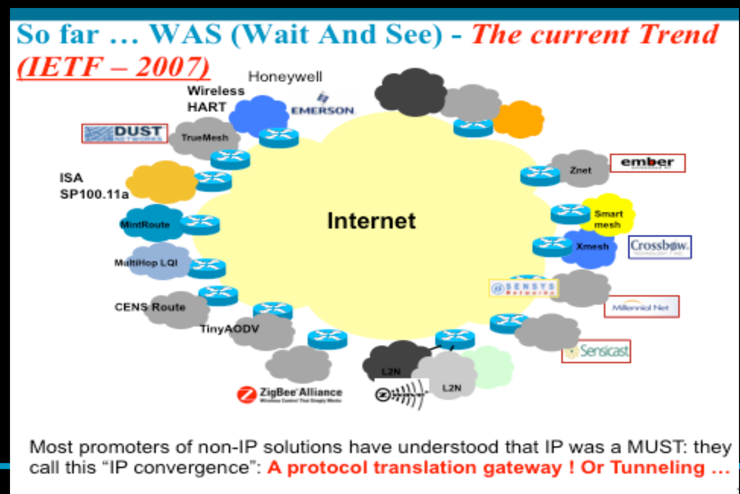
- RIP,
- OSPF,
- IS-IS,
- BGP
- MANET: AODV, OLSR, ...

The Internet Engineering Task Force

- New Routing WG (ROLL) formed for LLN in 2008



Reuse whenever possible



Routing Over Low power and Lossy Link (ROLL) WG

- Working Group Formed in Jan 2008 and re-chartered once

  http://www.ietf.org/html.charters/roll-charter.html

  Co-chairs: JP Vasseur (Cisco), David Culler (Arch Rock)

- Mission: define Routing Solutions for LLN (Low power and Lossy Networks)
- Very active work with a good variety of participants
- Rechartered to specify the routing protocol for smart objects networks (after protocol survey)
- DT formed (and now dissolved)
- Several proposals: ***one*** of then adopted as WG document: RPL

IETF WG ROLL status as of today

- Work Items
  - RPL is designed to support different LLN application requirements
    - RFC 5548 - Routing requirements for Urban LLNs
    - RFC 5673 - Routing requirements for Industrial LLNs
    - RFC 5826 - Routing requirements for Home Automation LLNs
    - RFC 5867 - Routing requirements for Building Automation LLNs
  - Routing metrics for LLN: approved
  - Produce a security Framework
  - Protocol work: Specified in https://tools.ietf.org/html/draft-ietf-roll-rpl-19
  - Applicability statement of ROLL routing protocols
- Timeline was key (in particular for SG).

RPL Tutorial

Specific Routing Requirements

- Deliberate choice of 4 main application areas
- Support of unicast/anycast/multicast
- Adaptive routing with support of different metrics (latency, reliability, …)
- Support of constrained-based routing (energy, CPU, memory)
- Support of P2MP, MP2P and P2P with asymmetrical ECMP
- Scalability
- Discovery of nodes attributes (aggregator)
- 0-config (Warning not to add too many options !)
- Performance: indicative (lesson learned from the Internet)
- Security

Approach

- Examine current ROLL application requirement drafts
  - Distill a set of common requirements across application domains
  - Establish a minimalist set of criteria
- Examine current IETF routing protocols
  - In RFCs or I-Ds that are on a working group's agenda
  - Evaluate these protocols in terms of ROLL criteria

Slide from IETF-72

Five Criteria

- Table scalability: how does the routing table size scale?
- Loss response: how expensive is it when links come and go?
- Control cost: how does the control overhead scale?
- Link cost: can the protocol consider link properties?
- Node cost: can the protocol consider node properties?

Slide from IETF-72

Summary

Name	Table Size	Loss Response	Control Cost	Link Cost	Node Cost
OSPF	fail	fail	fail	pass	fail
OLSRv2	fail	fail	fail	pass	pass
TBRPF	fail	pass	fail	pass	?
RIP	fail	fail	fail	?	fail
AODV	pass	?	pass	fail	fail
DSDV	fail	fail	fail	?	fail
DYMO[-low]	pass	fail	pass	fail	fail
DSR	fail	?	pass	fail	?

Slide from IETF-72

IETF ROLL WG Consensus

- Several routing protocols:
  - Proactive: RPL (initial work by the Design Team)
  - Reactive: DADR, …
- Strong WG consensus to adopt RPL as the routing protocol for LLN

RPL: the IP Routing Protocol for Smart Objects

Application Layer
(HTTP, SMTP, FTP, SNMP, IMAP, DNS, …)

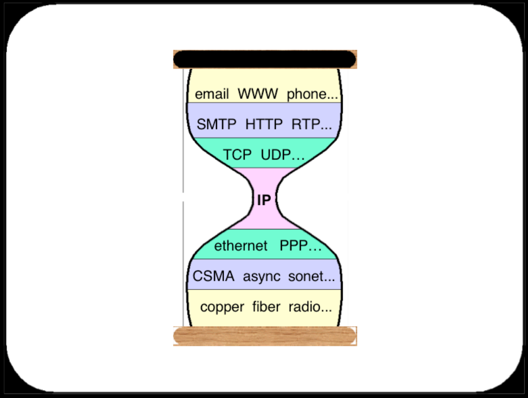
Transport Layer
(TCP, UDP, SCTP, RTP, …)

Internet Layer IP
(Routing, Multicast, QoS, …)

Link Layer

Physical Layer

In compliance with the layered architecture of IP, RPL does not rely on any particular features of a specific link layer technology. RPL is designed to be able to operate over a variety of different link layers, including ones that are constrained, potentially lossy, or typically utilized in conjunction with highly constrained host or router devices, such as but not limited to, low power wireless or PLC (Power Line Communication) technologies.



Protocol Design Choices

- Difficult tension between {flexibility, wide set of requirements, constrained devices}
- Option 1: take the union … Not a good choice …
  - If at all possible not always a good choice to overload the protocol with features not used by the application
  - An aspect that has been neglected by several protocol “Designers”
- Option 2: take the intersection and make the design modular ☺
  - Typically a subset of the RPL specification can be implement in light in the network requirement
  - Allows for minimal footprint implementations

RPL builds Directed Acyclic Graphs

- Tree would have been simpler but need for redundancy
- RPL supports the concept of DAG instances (a colored DAG), concept similar to MTR
- Allows a node to join multiple colored DAG with different Objective Functions
- And within an instance, there might be multiple DODAG (Destination Oriented DAG)
- A node may belong to more than one RPL instance
- Packets are tagged to follow a specific instance (defined at the application layer): no loops between instances

RPL Message Types

- RPL Control message are ICMPv6 messages
- RPL message={Base, Options}

  DIS: DODAG Information Solicitation

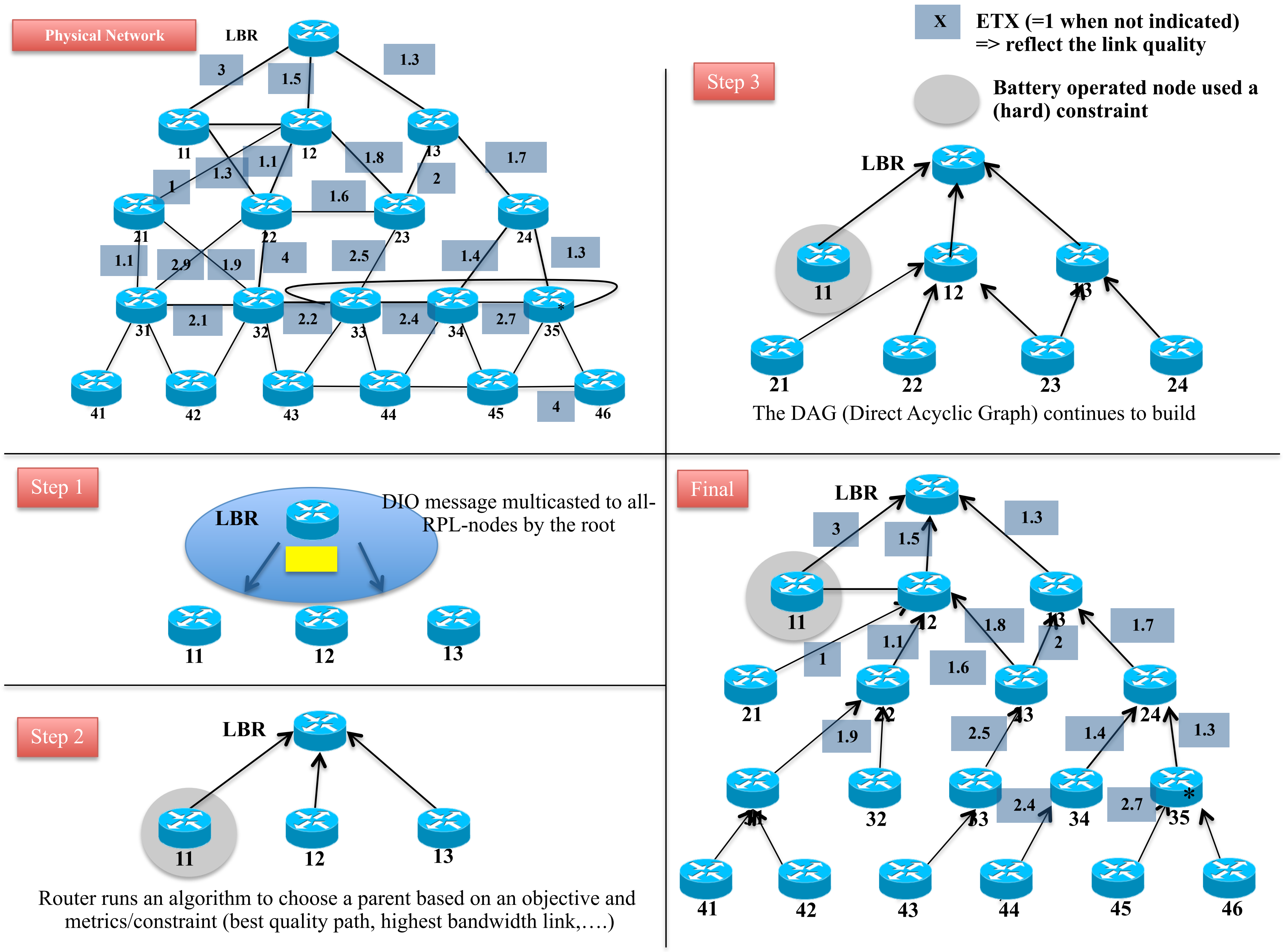
  DIO: DODAG Information Object

  DAO: Destination Advertisement Object

  DAO-ACK: Destination Advertisement Object Acknowledgement

  + The 4 secured versions
- Link-local scope: source is link-local unicast and destination=link-local unicast or all-RPL-nodes(FF02::1) (for all RPL messages except DAO/ DAO-ACK in non storing mode, DIO replies to DIS)

Building a DAG – Upward routing


Physical Network
LBR
X
ETX (=1 when not indicated)
=> reflect the link quality
Battery operated node used a (hard) constraint
Step 3
LBR
The DAG (Direct Acyclic Graph) continues to build
Step 1
LBR
DIO message multicasted to all-RPL-nodes by the root
Step 2
LBR
Router runs an algorithm to choose a parent based on an objective and metrics/constraint (best quality path, highest bandwidth link,….)
Final
LBR

The RPL DIO Message

G: Grounded

RPLInstanceID				Version	Rank	
G	0	MOP	Prf	DTSN	Flags	Reserved
DODAGID						
Options						

MOP:

0: no Downward route

1: Non Storing Mode

2: Storing without Multicast

3: Storing with Multicast

DTSN: set by the node issuing the DIO used to maintain DAO

Option

Type	Length	Data …

RPL Option DODAG Configuration Option

- DODAG Configuration (in DIO): unchanged by intermediate nodes, sent occasionally (always upon receiving DIS)

Path Control Size: #bits of Path Control Field

DagMaxRankIncrease may be used by Local Repair

Parameters controlled by root

ype=4	Length=14	Flag	A	PCS	DIOIntDbl
DIOIntMin	DIORund	MaxRankIncrease			
MinHopRankIncrease		OCP			
Reserved	Def Lifetime	Lifetime Unit			

DIOIntDbl: DIOInterdoubling

DIOIntervalMin: Imin

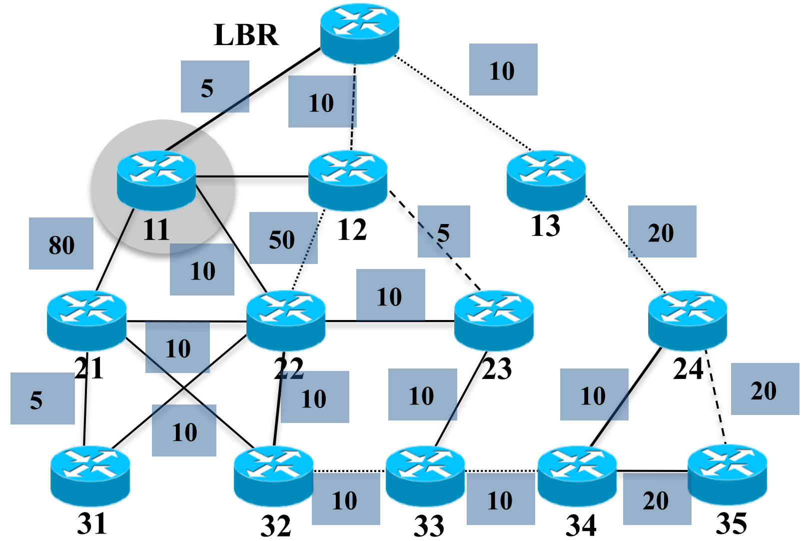
DIORund: K

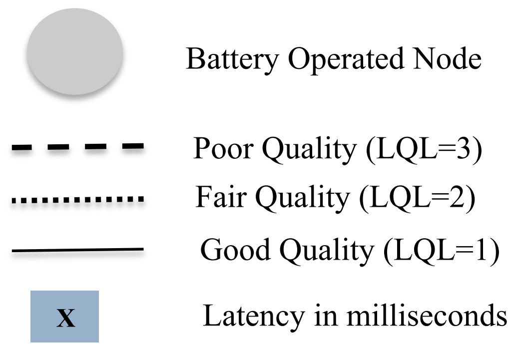
Default Lifetime for all RPL routes

Example of a RPL option

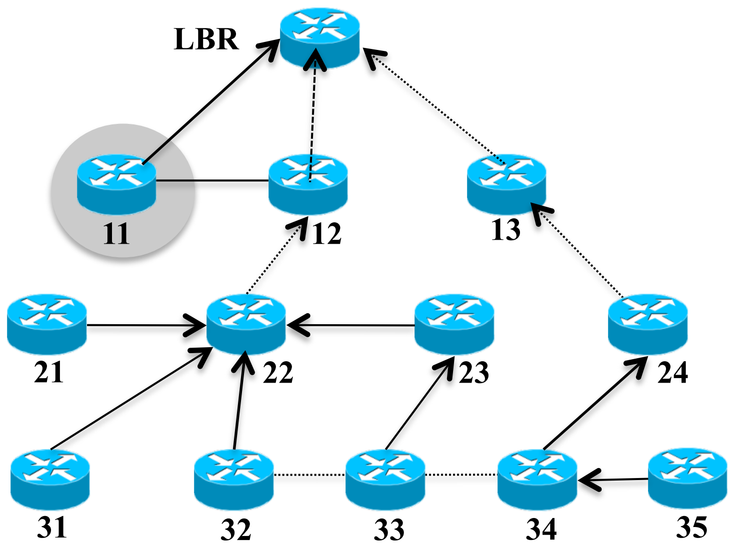
Concept of Multiple RPL Instances (a la MTR)

Physical topology

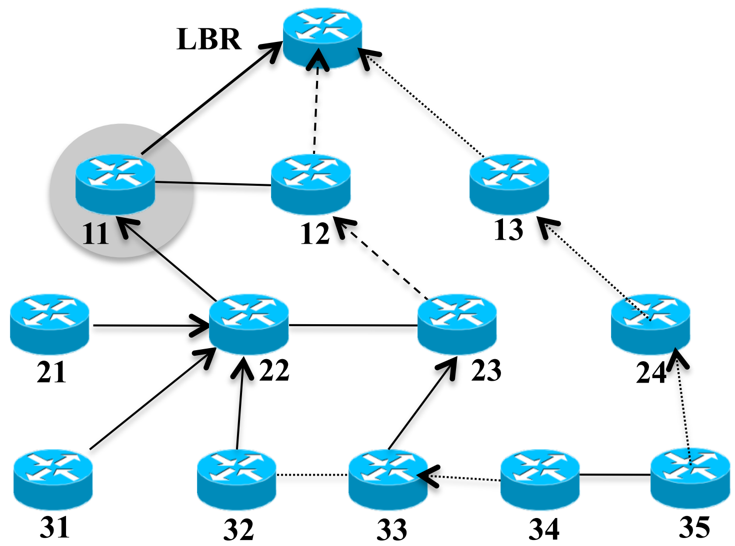




DAG Instance 1: high quality – no battery operated nodes
DAG Instance 2: low latency



RPL instance 1



RPL instance 2

Routing Metrics used by RPL

Use of adaptive routing metrics …

- Today's IGP use ***static link*** metrics

  Administrative cost or polynomial cost

- Using dynamic metric is not a new idea (experimented in ARPANET-II based on average queue lenght)

  Hard to control … routing oscillations

  Issue with too frequent control traffic in LLN

Routing Metrics in LLNs

Specified in draft-ietf-roll-routing-metrics

Node Metrics	Link Metrics
Node State and Attributes Object Purpose is to reflects node workload (CPU, Memory…) "O" flag signals overload of resource "A" flag signal node can act as traffic aggregator	**Throughput Object** Currently available throughput (Bytes per second) Throughput range supported
Node Energy Object "T" flag: Node type: 0 = Mains, 1 = Battery, 2 = Scavenger "I" bit: Use node type as a constraint (include/exclude) "E" flag: Estimated energy remaining	**Latency** Constraint - max latency allowable on path Metric - additive metric updated along path
Hop Count Object Constraint - max number of hops that can be traversed Metric - total number of hops traversed	**Link Reliability** Link Quality Level Reliability (LQL) 0=Unknown, 1=High, 2=Medium, 3=Low Expected Transmission Count (ETX) (Average number of TX to deliver a packet)
	Link Colour Metric or constraint, arbitrary admin value

Object can be used as metric and/or constraint - metric can be additive/max/..

The RPL Objective Function

- An objective function defines how nodes perform parent selection and how to compute rank based on metrics
- Defined by the OCP
- Combined with metrics/constraints

  "Use the LQL as a global recorded metric and favor paths with the minimum number of low and fair quality links, use the link color as a link constraint to avoid non encrypted links".

  "Find the best path in term of latency (link latency is used as a global aggregated metric), while avoiding poor quality links and battery operated nodes".

- See OF0 and MRHOF

Mode of Operation

- Parent selection governed by OF, decoupled from metrics and constraints
- Node leaving a DODAG should remember the DODAG parameters for some period of time to avoid rejoining a node in a former sub-DAG, thus avoiding loops.
- DODAGVersion governed by DAGroot according to implementation specific events
- A node with DODAG parent set =={} may set G bit (with lower Pref)
- DODAG selection is also implementation-specific

Mode of Operation (Cont')

- Movement within a DAG

    Node may jump to a parent with a lower rank

    Within a DODAG a node cannot advertise a rank L>L +DAGMaxRankIncrease (RANK=INFINITY is an exception)

    Node can select any parent advertising a higher DODAGVersion

    Node can at any time join a different DODAG within the same RPL instance with no rank restriction (except if the node used to belong to this DODAG Version)

    If a node needs to move down it MAY poison its sub-DAG

- Poisoning

    Means sending DIO with Rank=INFINITY

    Node cannot select a parent advertising a Rank=Infinity

    Still a node may detach without poisoning by setting the G Flag

Mode of Operation (Cont)

- A node should prefer to stay in its DODAG via an alternate parent if any should the preferred parent have left its DODAG
- DIO transmission is governed by Trickle Timer
- Reception of DIO from less DAGRank causing no change to DODAG parent set, preferred parents, Rank, reception of unicast DIS, => consistent
- Trickle timer reset upon inconsistency detection:

  Packet forwarding error (Rank-error, Forwarding-error, …)

  Reception of DIS with all predicates==true

  New DODAGVersion (new DODAGVersionNumber, new RPL Instance)
- See Trickle Algorithm – next slide

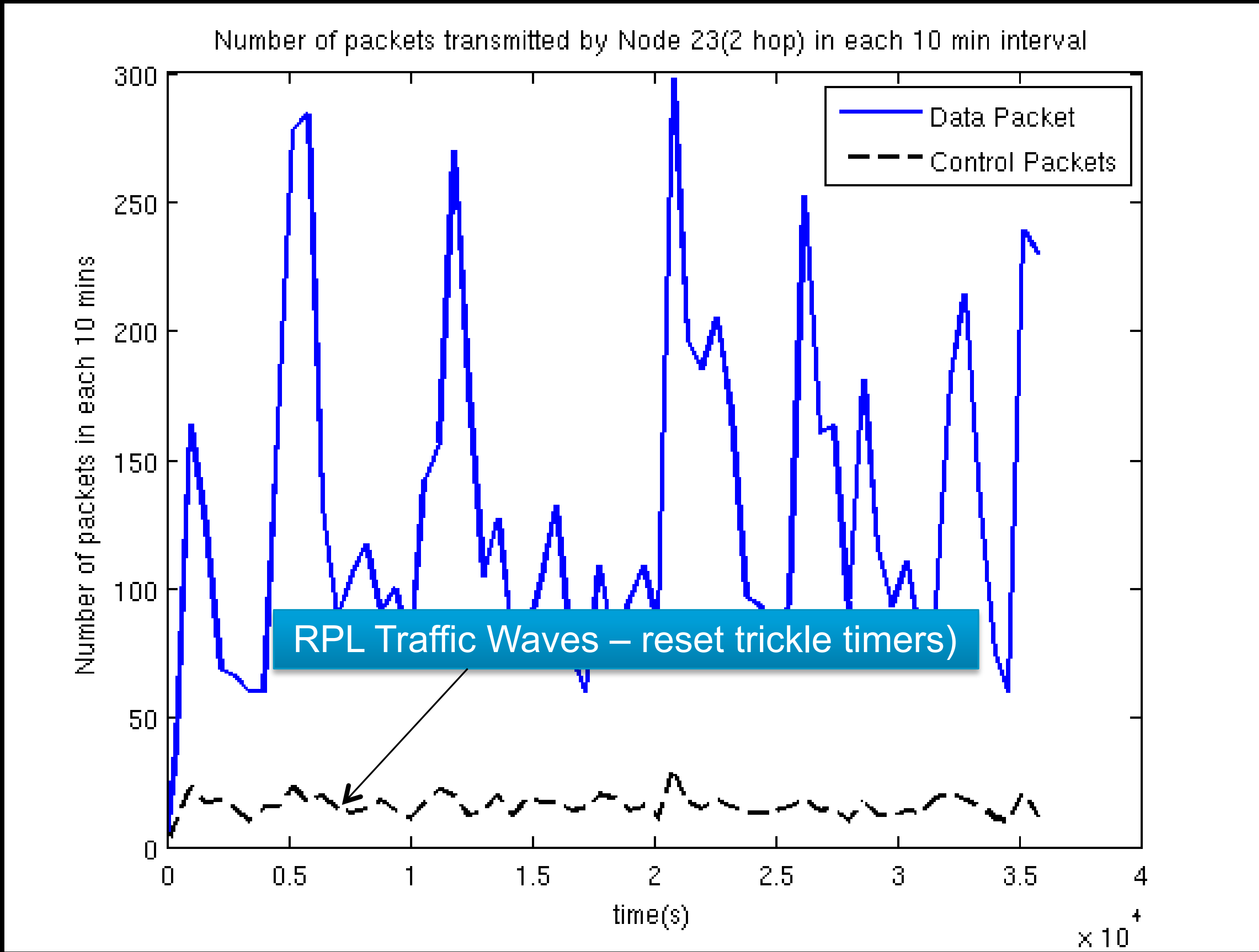
The use of Trickle Timers

- The basic idea is to suppress redundant messages (key when resources such as energy and bandwidth are scarce)
- Here is the algorithm:

T: Timer value. T is in the range [I, I/2]
C: Redundancy counter
K: Redundancy constant
$\mathbf{I_{min}}$: Smallest value of I
$\mathbf{I_{doubling}}$: The number of times is doubled before maintaining a constant multicast rate.
$\mathbf{I_{max}}$: Largest value of $\mathbf{I_{max}} = \mathbf{I_{min}} * 2^{\mathbf{Idoubling}}$.
When T fires, if C>K, then send DIO, then upon expiration of I, compute new(I) and T.

Detection of inconsistency => Trickle timer reset
Nodes may increment C if they receive consistent messages

Trickle at works …



RPL DIS message and Option

- Base Format:

Flags	Reserved	Option …

- Allows for predicate to solicit replies from subset of nodes

V: Version predicate

Receiver DOAGVersion=Version?

I: InstanceID predicate

Receiver RPLInstanceID=RPLInstanceID ?

Type=7	Option Length=19	RPLInstanceID	V	I	D	Flags
DODAGID						
Version num						

- Used to solicit a DIO from a RPL node in the vicinity
- Ability to add filtering to the request to limit the number of replies (use of predicates)

D: DODAGID predicate

Receiver DODAGID=DODAGID?

Node reset trickle timer when all predicates are true.

Operation as a leaf node

- As with many other protocol, refers to the ability to participate to the routing domain, without forwarding RPL traffic.
- Governed by policy or upon receiving unsupported/ unrecognized OF/Metric/Constraint
- Send DIO in very specific circumstances (transition – node was part of another DODAG receiving packet for old topology) with RANK=Inifinite
- May unicast transmit DAO or multicast DAO for 1-hop optimization

Downward routing

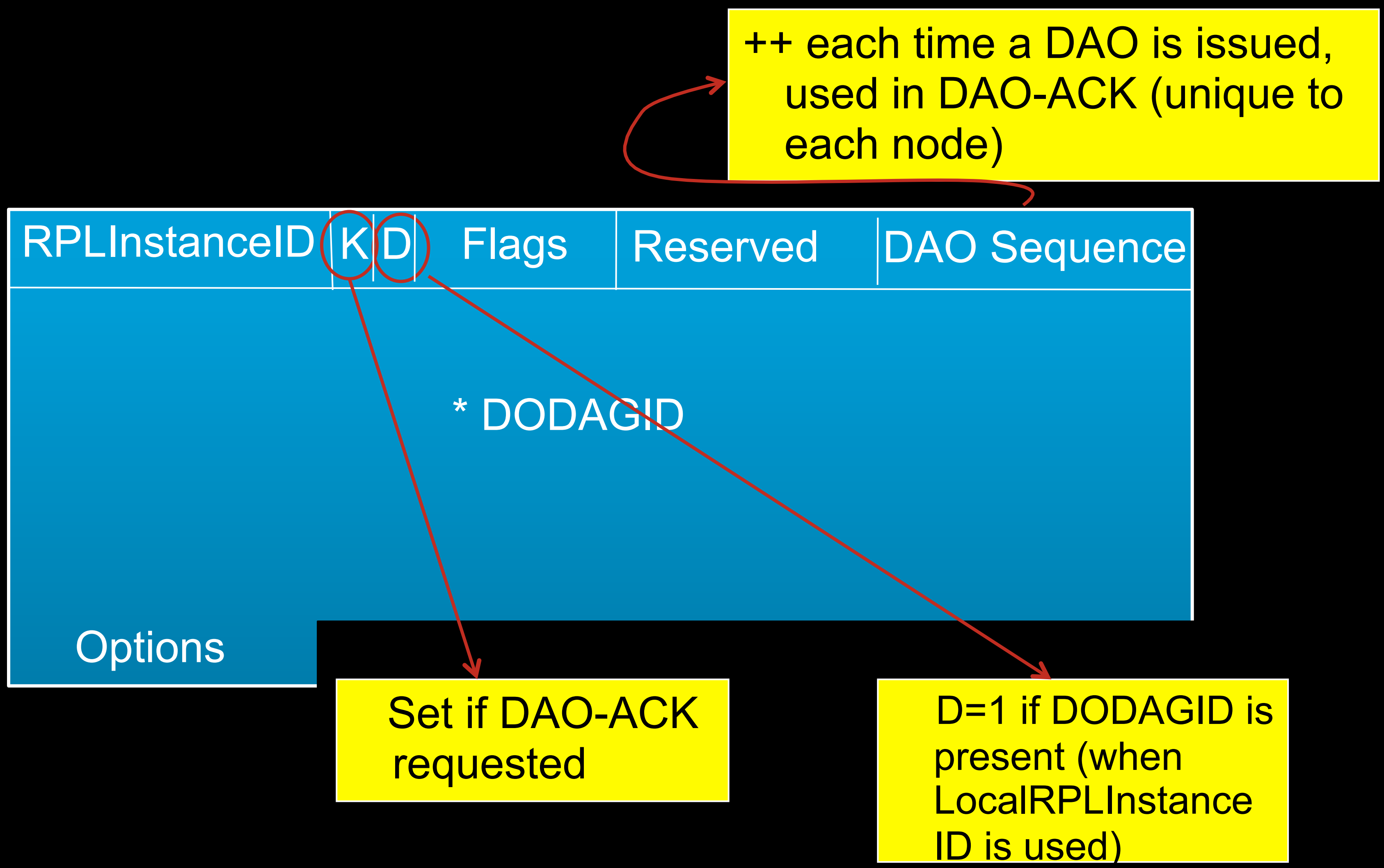
Storing Versus Non-Storing Mode

- RPL supports two modes (mix not allowed):
  - Storing mode: nodes do store/maintain routing tables
  - Non-storing mode: nodes use default routing upward and source routing downward
- Upward routing similar
- Differences in downward routing:
  - Storing mode: packets travel up to a common ancestor
  - Non storing mode: packets travel up to the DAG root, then source routed
- Impact on forwarding:
  - Use of source routing in non-storing
  - No use of RPL option header (no risk of loops) in non-storing

Populating the Routing Tables

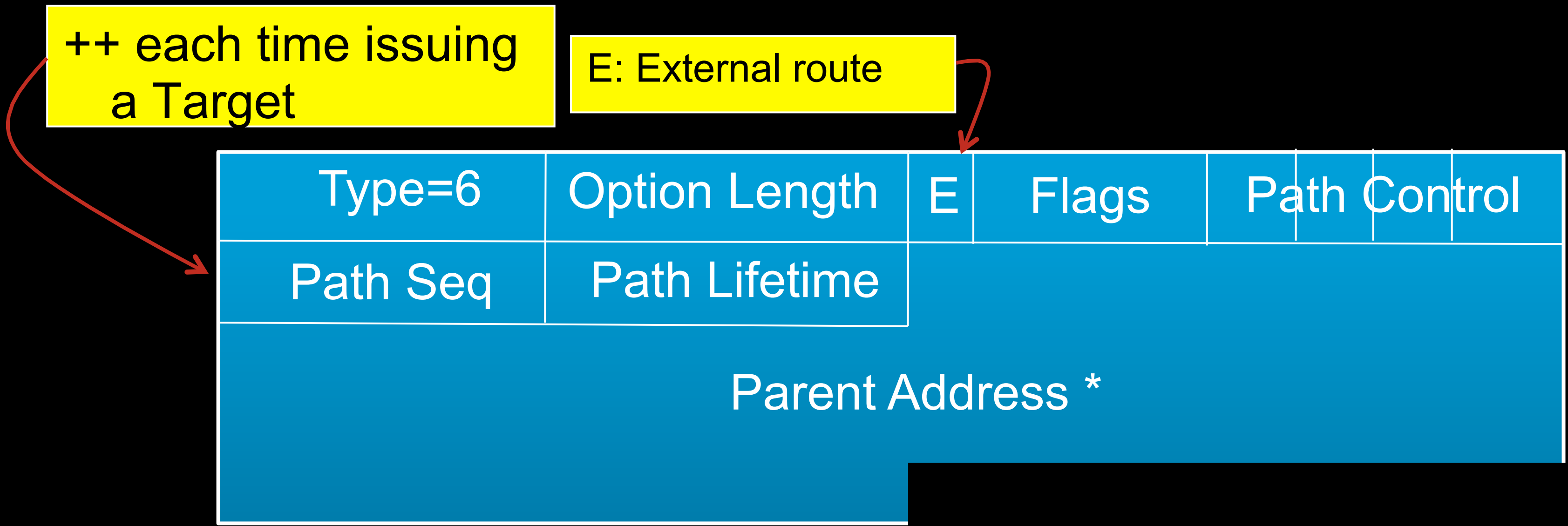
- DAG provides UP connectivity
- Requires DOWN connectivity (routes toward the leaves)
- RPL specifies DAO messages used to advertize prefixes to parents (storing) or DAGRoot (non-storing)
- Nodes capable of storing the prefixes populate their routing tables
- Packets are routed up to a common ancestor for P2P routing with an optimization for 1-hop reachable nodes

The RPL DAO Message



RPL Option: Target

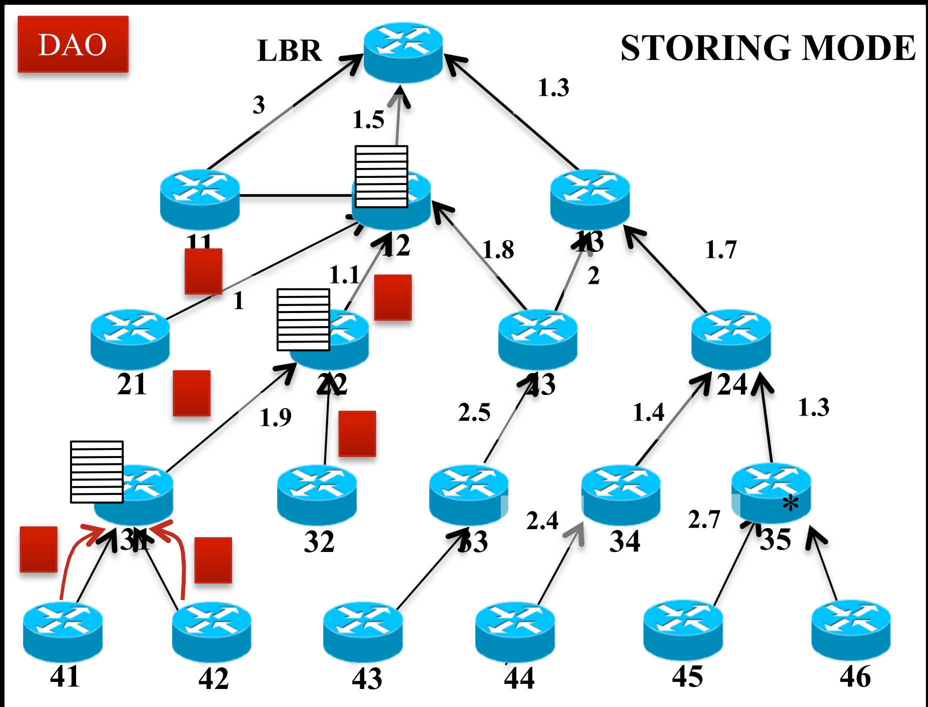
- RPL Target (may be in DAO): indicates reachability.
- Transit Information (may be in DAO): may contain Parent address for an ancestor used with source routing (could be one Transit info per DAO parent for non-storing). A Path Control field used for non storing to influence the reverse path.



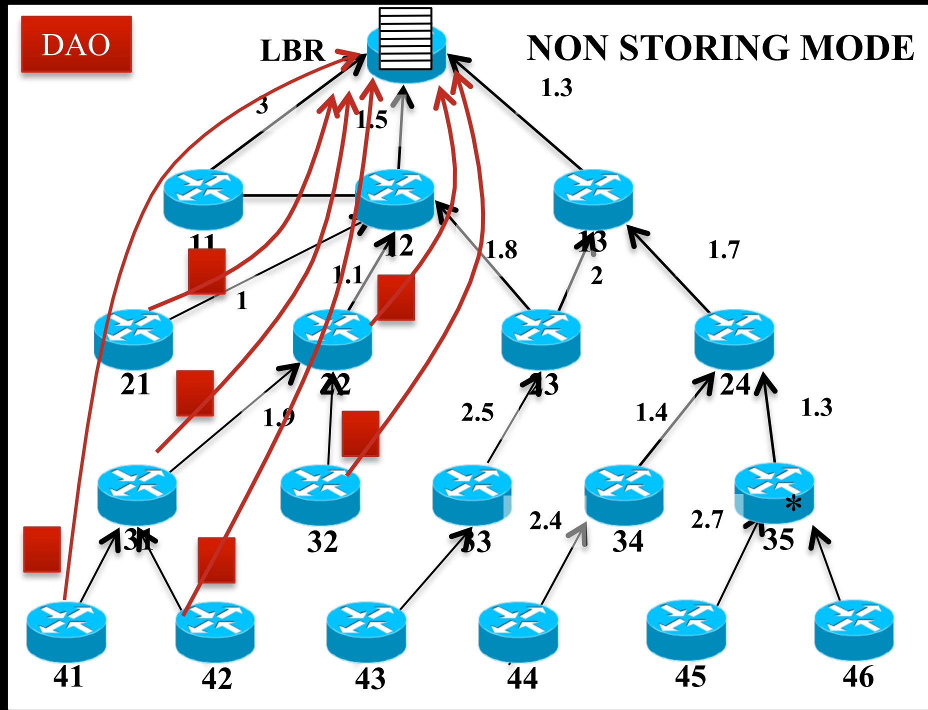
Path Control: Number of 1-bit fields specified in PCS of DODAG config (DIO).

Populating the routing tables using DAO message

- Two modes of operations: storing mode and non storing modes



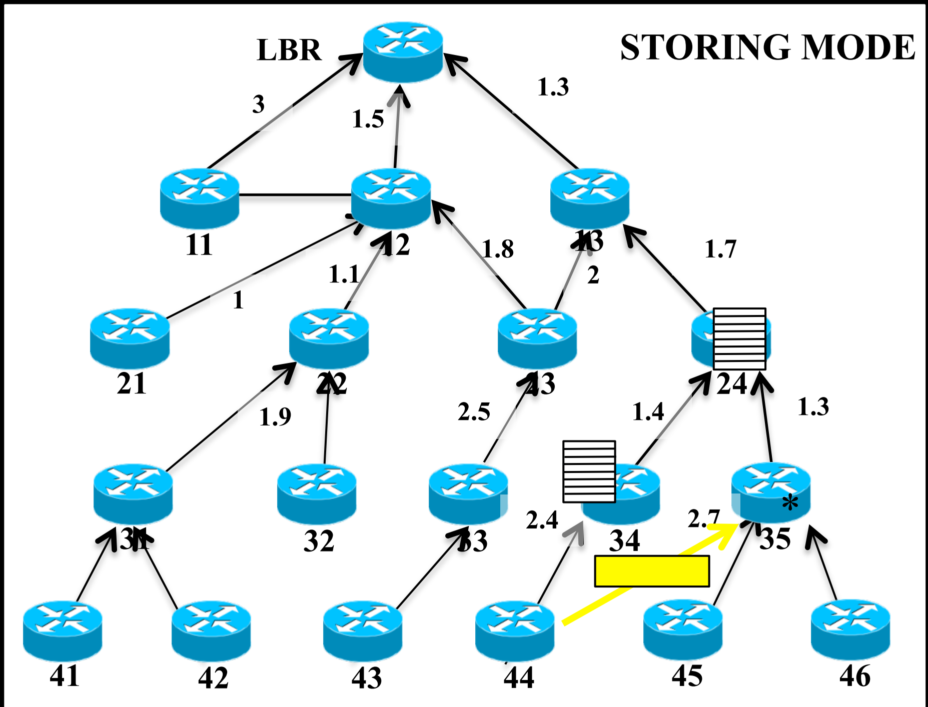
Unicast to DAO parents



Unicast to DODAG Root (not processed by intermediate nodes)

Triggering Multicast DAO for 1-hop routing

- Node may multicast DAO using link local all-RPL-node
- Used only to advertise information about the node itself (prefixes directly connected or owned by the node itself)
- MUST NOT be used to relay information using unicast DAO
- Usually preferred than routes learned through unicast DAO



Loops in DODAG: detection and repairs

Potential Loops

- Loops in DV are hardly avoidable (due to control message loss and sibling routing)
- Tension between Loops avoidance and loop detection
- RPL supports both
  - E.g. Rules about the rank: do not attach to a node deeper in the DAG
  - E.g. Set flags in the packet header to detect loops that may occur (datapath validation)
- When loop could occur ?
  - DIO/DAO message loss is the most common example …
- RPL makes use of on-demand loop detection with data packets (e.g. rank of sender and direction are used for loop detection)

Loop Detection

- Idea: add flags to data packets to detect and breaks loops
- Receiving a packet with inconsistent flag according to the rank is a loop indication (Set to INFINITY when moving to new DODAGVersion)
- DAG inconsistency loop detection:
    - O=1 from a node with higher Rank
    - O=0 from a node with a lower rank
- Once an inconsistency is detected, the R bit is set. Upon receiving a packet with R=1, the packet MUST be dropped and trickle reset.

<table>
<tr><td colspan="5"></td><td>Option Type</td><td>Opt Data Len</td></tr>
<tr><td>O</td><td>R</td><td>F</td><td>0 0 0 0 0</td><td>RPLInstanceID</td><td colspan="2">Sender Rank</td></tr>
<tr><td colspan="7">Sub-TLVs</td></tr>
</table>

O: Down bit; R: Rank-error bit; F: Forwarding Error bit => Expected to change en-route (packet MUST be discarded if RPL option header not understood)

RPL Option header (IPv6 Hop-by-hop header) immediately following the IPv6 header

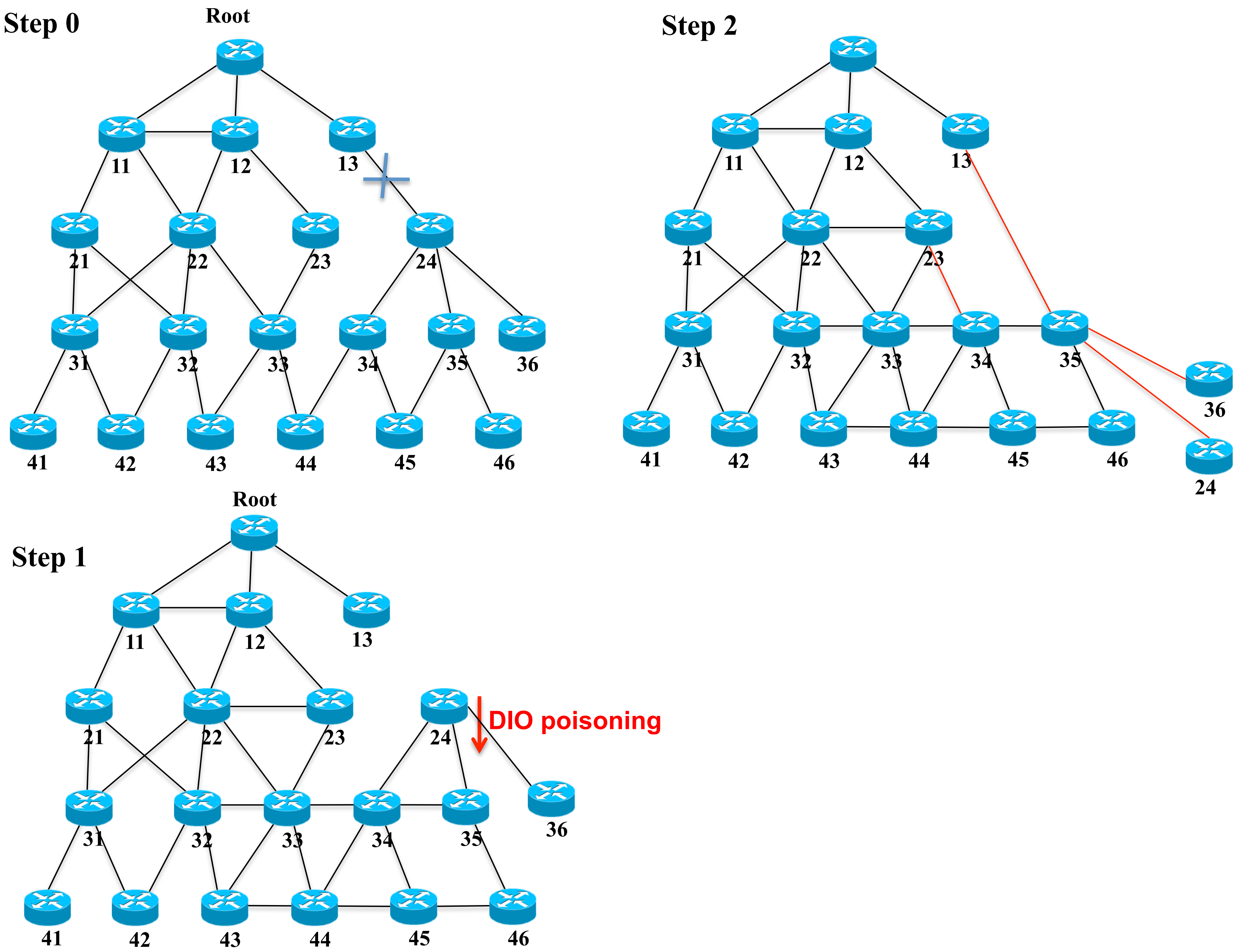
DAO inconsistency detection and recovery

- Only for storing mode (use of ICMP “Error in source routing Header” in non-storing mode)
- Upon reception of a packet with no-route, sent it back with F=1 to clean-up state.

DODAG Maintenance and Repairs

Global versus Local Repair

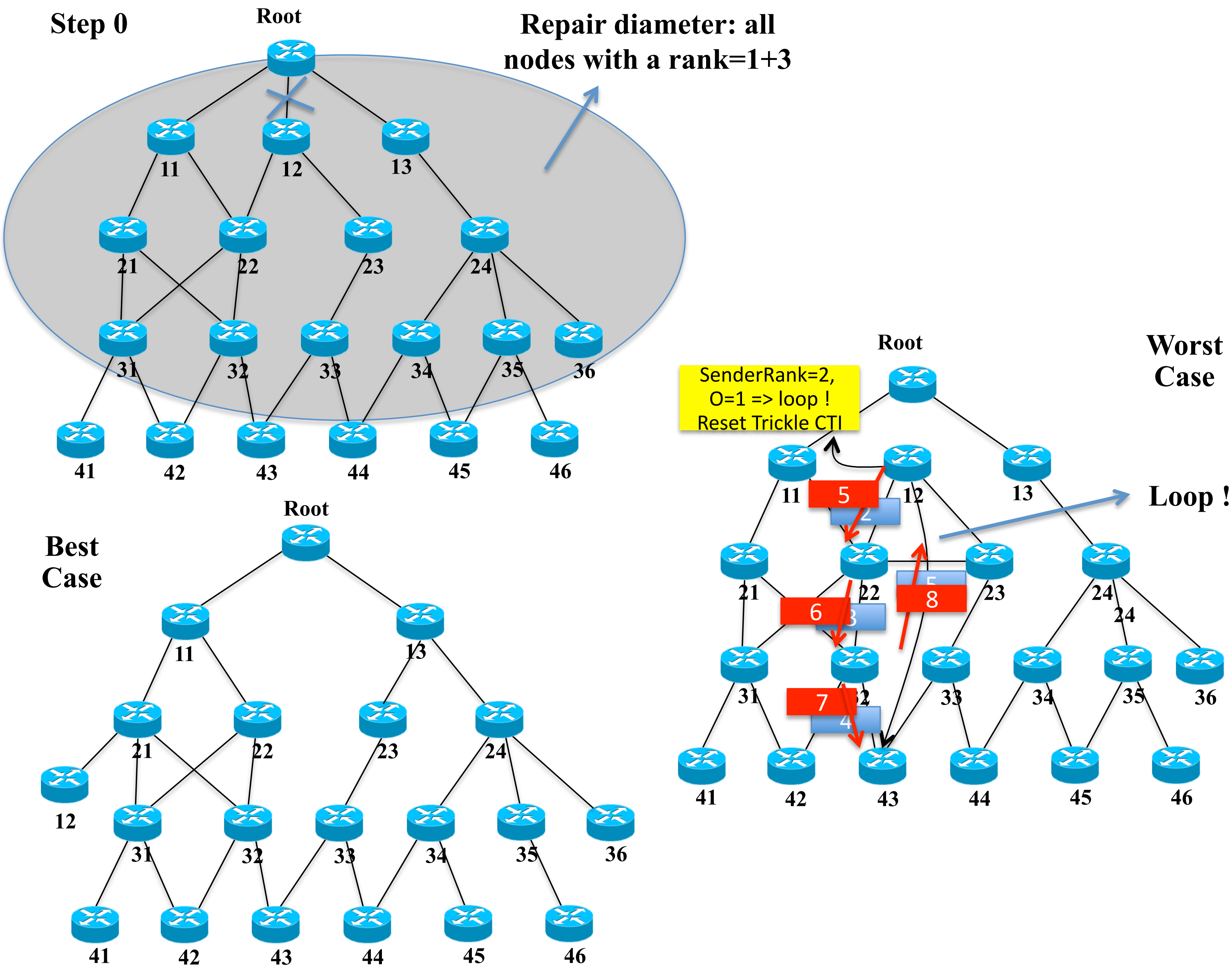
- Global repair: rebuilt the DAG … requires a new DAG Sequence number generated by the root
  - Triggered by the root
  - Potentially signaled to the root (under investigation)
- Local Repair: find a “quick” local repair path
  - Only requiring local changes !
  - May not be optimal according to the OF and overall DAG shape, which is fine
- Complementary approaches


Step 0
Root
11
12
13
21
22
23
24
31
32
33
34
35
36
41
42
43
44
45
46
Step 2
11
12
13
21
22
23
31
32
33
34
35
36
41
42
43
44
45
46
24
Step 1
Root
11
12
13
21
22
23
24
DIO poisoning
31
32
33
34
35
36
41
42
43
44
45
46

Remember the DAGMaxRankIncrease ?

- Before moving down, remember the rank r
- Do not attach to any node with rank>r+H
- This help reduce the size of infinity ☺

Let L be the lowest rank within a DODAG Version that a given node has advertised. Within the same DODAG Version, that node MUST NOT advertise an effective rank higher than L + DAGMaxRankIncrease. INFINITE_RANK is an exception to this rule: a node MAY advertise an INFINITE_RANK within a DODAG version without restriction. If a node's Rank were to be higher than allowed by L + DAGMaxRankIncrease, when it advertises Rank it MUST advertise its Rank as INFINITE_RANK.


Step 0
Root
Repair diameter: all nodes with a rank=1+3
11 12 13
21 22 23 24
31 32 33 34 35 36
41 42 43 44 45 46
Best Case
Root
11 13
12
21 22 23 24
31 32 33 34 35 36
41 42 43 44 45 46
Worst Case
Root
SenderRank=2, O=1 => loop ! Reset Trickle CTI
Loop !
11 12 13
21 22 23 24
31 32 33 34 35 36
41 42 43 44 45 46
5
2
6
3
8
5
7
4

Forwarding in LLN

Forwarding

Two related drafts (Passed Working Last Call):

- **draft-ietf-6man-rpl-option**
  - RPL information in data-plane with IPv6 Hop-by-Hop Option
- **draft-ietf-6man-rpl-routing-header**
  - Source routes for non-storing mode with IPv6 Routing Header

Some Simulation Results

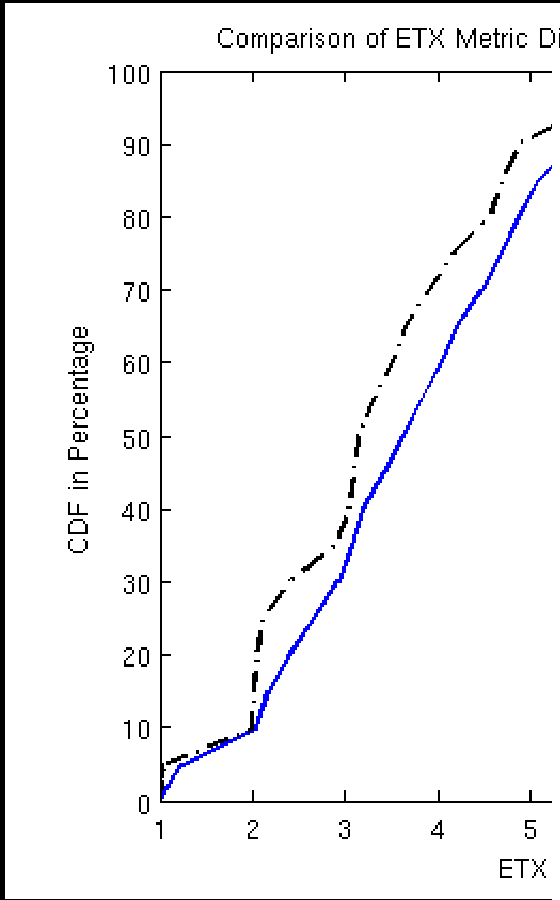
Simulation Results

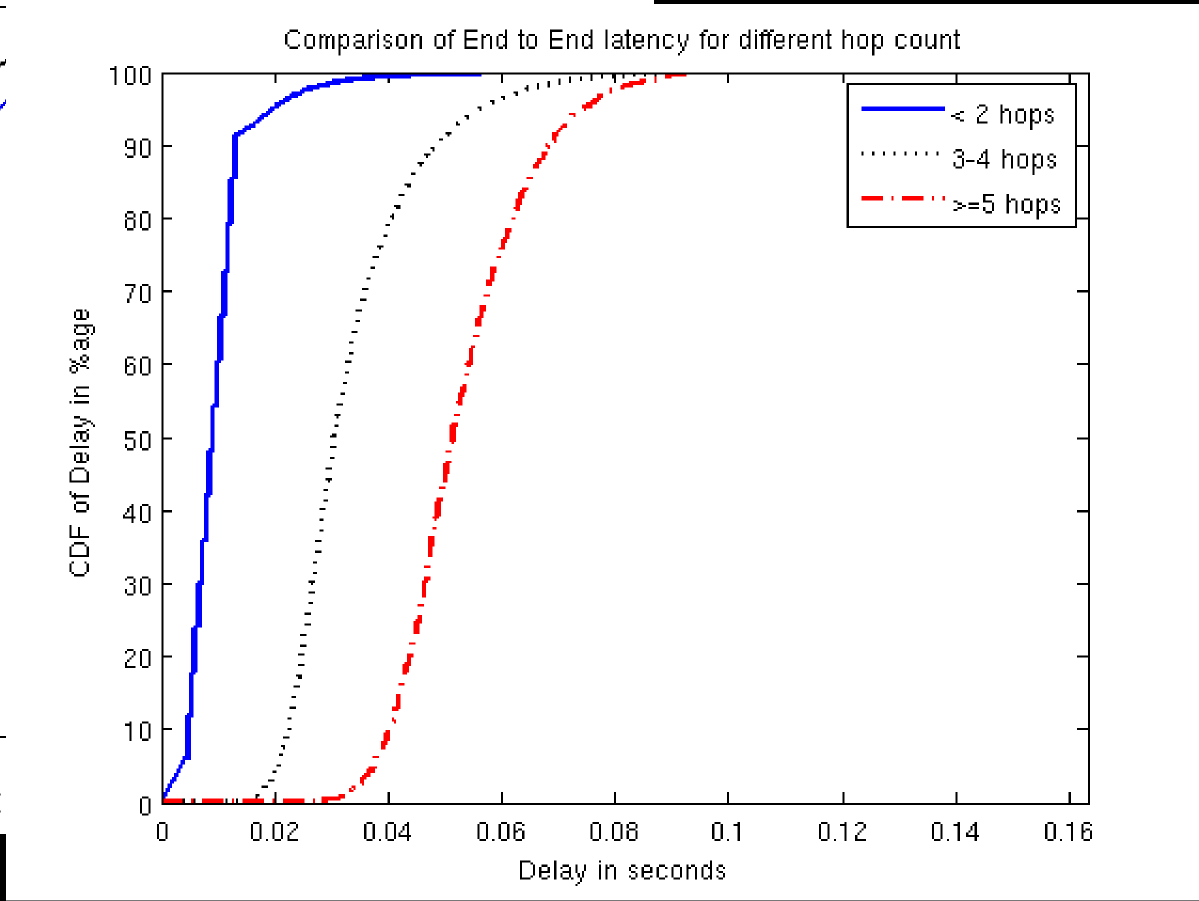
- Based on Omnet++ (Discrete Event simulator) / Castalia (Wireless) – Radio: TelosB CC2420 with 15.4 links.
- No formal Markov chain modeling: capture of thousands on link traces capturing topologies and PDR/RSSI, both in-door and outdoor. Additional Markov Chain for link failures.
- Trickle used with I_min=1s and I_doubling (Max=18.2 hours)
- Most traffic simulated MP2P(80%), CBR, Metric=ETX (other simulation for building automation)

Simulations can be quite useful to find issues, not to demonstrate that a protocol actually works…

Simulation Results

Path Cost stretch using ETX as metric (specifically for building: 60% 1-hop, 20% 2-hop, 20% uniformly distributed).

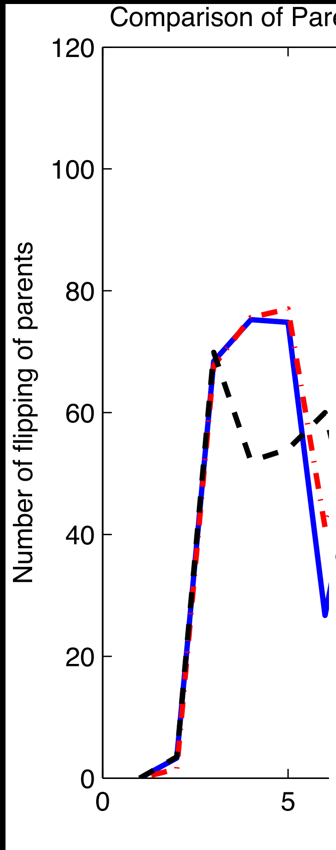


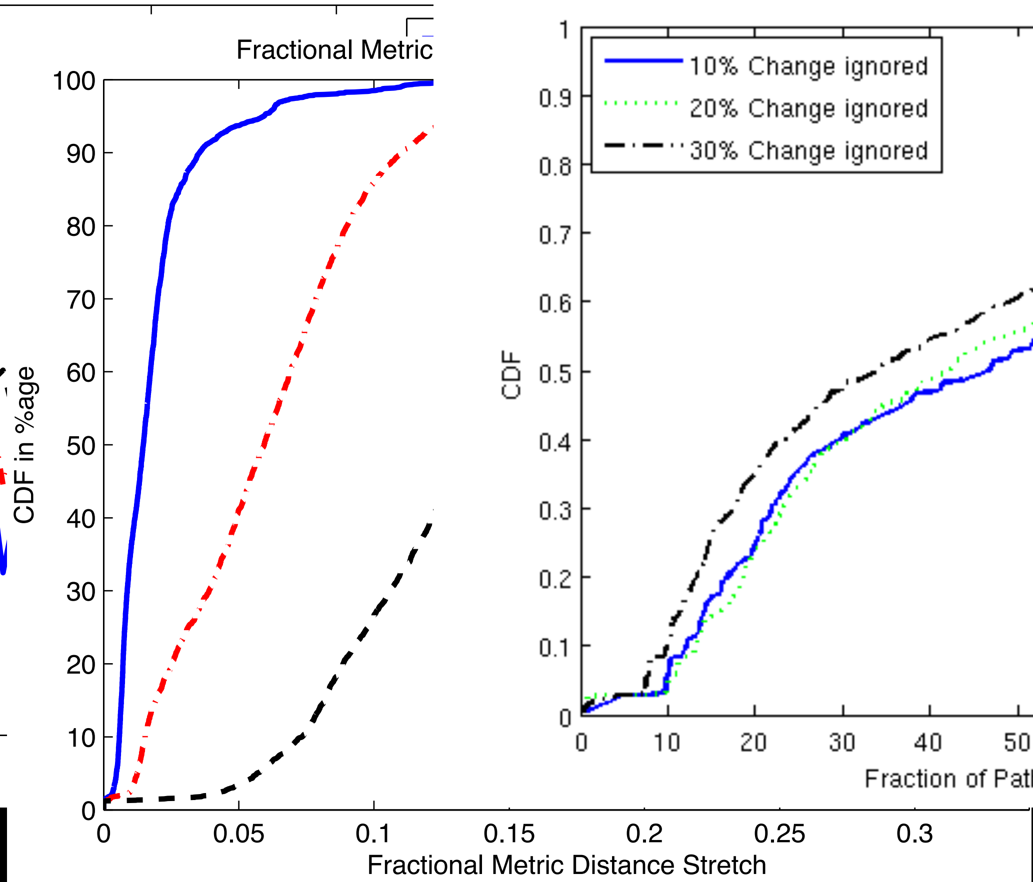


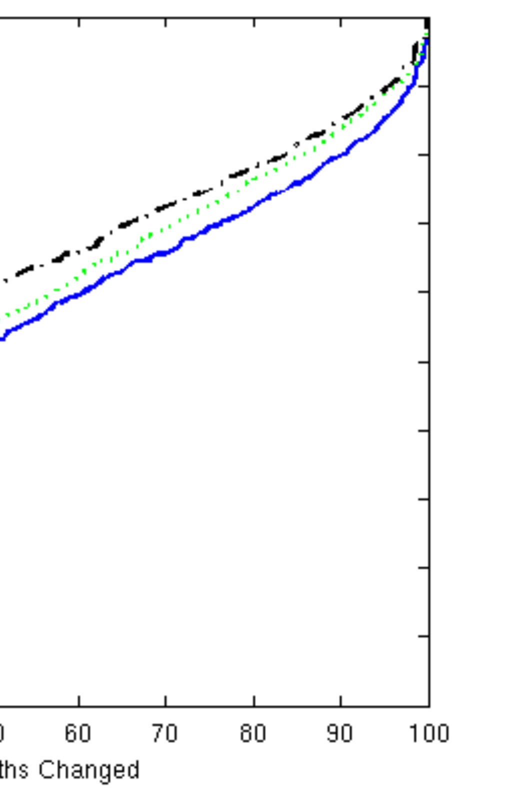
Simulation Results – Routing Stability

Number of parents flipping

Fraction of path change



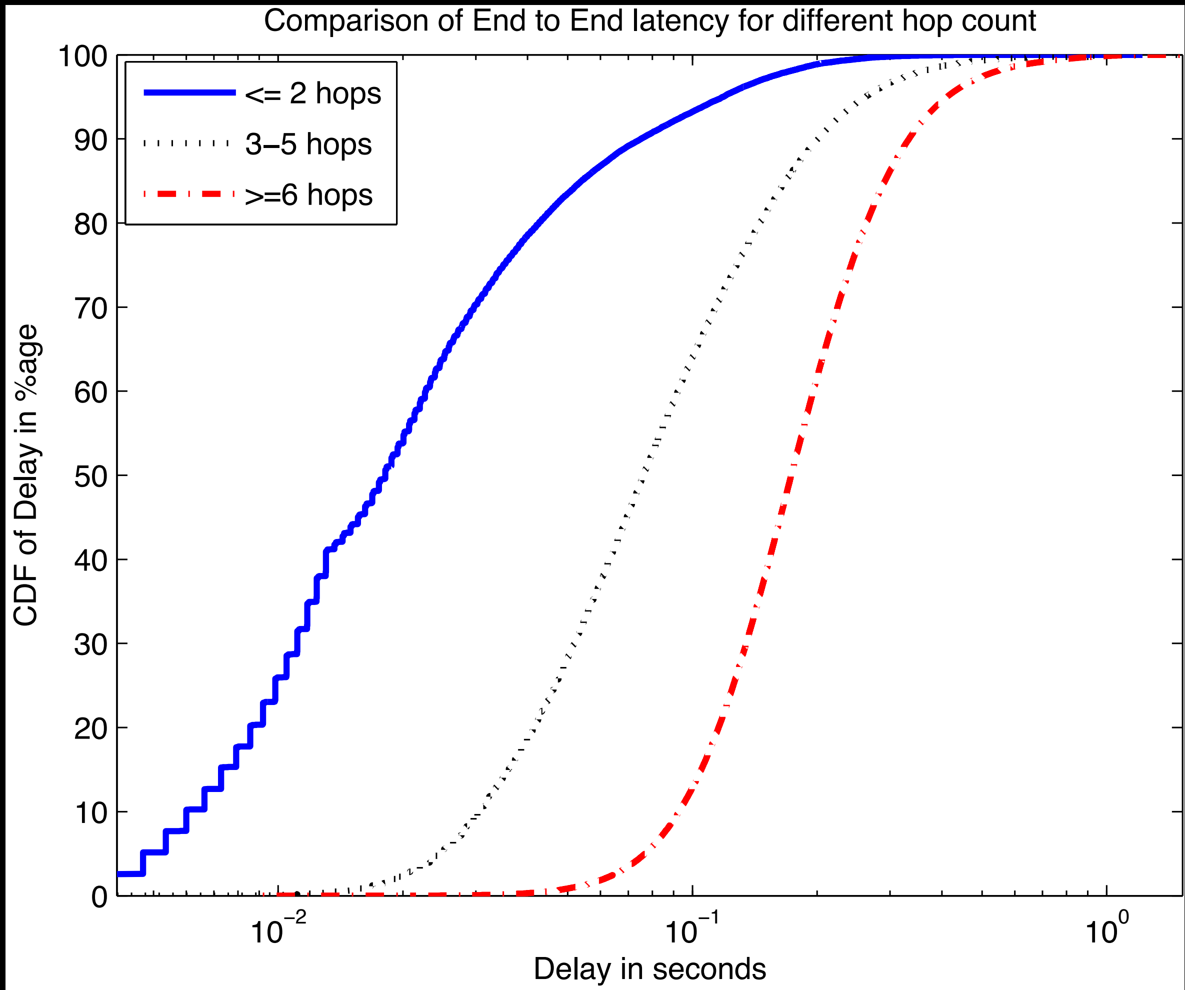




Path Cost Stretch

Simulation Results – Routing Stability

End-to-End Latency



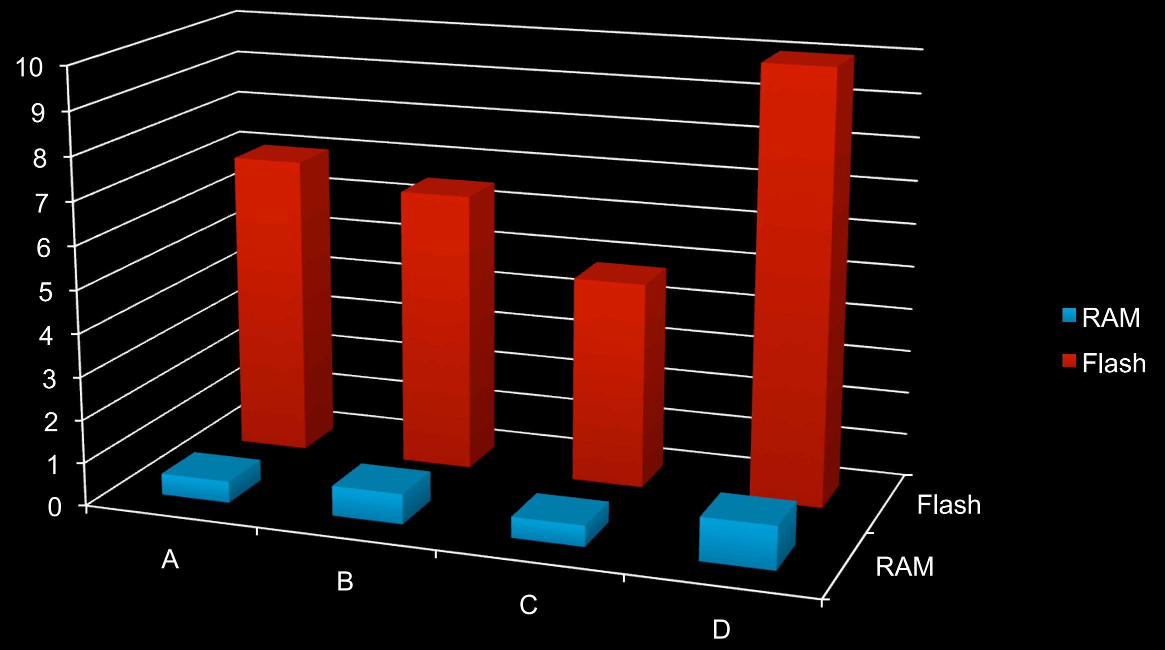
Is RPL Ready for "prime time" ?

- Lots of ideas coming from deployed networks
- More than 15 implementations *including several commercial products this year*
- Stable specification (One last IESG DISCUSS, Routing metrics, trickle approved)
- Two interoperability test events took place (earlier revision):
    - IPSO (Storing mode)
    - Zigbee/IP (Non-Storing modes)

– **Adopted by Industry alliances: Zigbee/IP, Wavenis, IEEE P1901.2, …**

Is RPL Lightweight ?

- RPL has been designed to be MODULAR
- RAM and Flash usage figures of four independent implementations



Potential Future items

- Applicability statements
- ROLL re-chartering discussion
- Potential candidates
  - Lightweight security
  - Label Switching (Label distribution, Forwarding !)
  - “Routing Admission control”
  - Path Computation Element

Thank you for your attention


cisco