6LowPAN
ACKs

☐ Slide/Figures Sources

- IPSO Alliance Webinar “6LowPAN for IP Smart Objects”
- 6LoWPAN: The Wireless Embedded Internet, Shelby & Bormann, ISBN: 978-0-470-74799-5, (c) 2009 John Wiley & Sons Ltd
What is 6LoWPAN?

- An Adaptation Layer to fit IPv6 over Low-Power wireless Area Networks
- Defined by IETF standards
  - RFC 4919, 4944
  - draft-ietf-6lowpan-hc and -nd
  - draft-ietf-roll-rpl

TCP/IP Protocol Stack

- HTTP
- RTP
  - TCP
  - UDP
  - ICMP

6LoWPAN Protocol Stack

- Application
  - UDP
  - ICMP

- Transport
- Network
- IPv6 with LoWPAN
- IEEE 802.15.4 MAC
- IEEE 802.15.4 PHY

- Physical
- Data Link
- Ethernet MAC
- Ethernet PHY
Benefits of 6LoWPAN Technology

- Low-power RF + IPv6 = The Wireless Embedded Internet
- 6LoWPAN makes this possible
- The benefits of 6LoWPAN include:
  - Open, long-lived, reliable standards
  - Easy learning-curve
  - Transparent Internet integration
  - Network maintainability
  - Global scalability
  - Enables a standard socket API
  - Minimal use of code and memory
  - Direct end-to-end Internet integration
  - Multiple topology options
Evolution of Wireless Sensor Networks

Price

Cabling

Proprietary radio + network

Z-Wave, prop. ISM etc.

ZigBee and WHART

6lowpan Internet

Scalability

Vendor lock-in

Complex middleware

Open development and portability

Increased Productivity

1980s

2000

2006

2008 ->
Architecture
LoWPANs are stub networks

Simple LoWPAN
- Single Edge Router

Extended LoWPAN
- Multiple Edge Routers with common backbone link

Ad-hoc LoWPAN
- No route outside the LoWPAN

Internet Integration issues
- Maximum transmission unit
- Application protocols
- IPv4 interconnectivity
- Firewalls and NATs
- Security

IPv6-LoWPAN Router Stack

<table>
<thead>
<tr>
<th>IPv6</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet MAC</td>
<td>LoWPAN Adaptation</td>
</tr>
<tr>
<td>IEEE 802.15.4 MAC</td>
<td></td>
</tr>
<tr>
<td>Ethernet PHY</td>
<td>IEEE 802.15.4 PHY</td>
</tr>
</tbody>
</table>
Adaptation Features

- Efficient header compression
  - IPv6 base and extension headers, UDP header

- Fragmentation
  - 1280 byte IPv6 MTU -> 127 byte 802.15.4 frames
Additional Features

- Support for e.g. 64-bit and 16-bit 802.15.4 addressing
- Useful with low-power link layers such as IEEE 802.15.4, narrowband ISM and power-line communications
- Network autoconfiguration using neighbor discovery
- Unicast, multicast and broadcast support
  - Multicast is compressed and mapped to broadcast
- Support for IP routing (e.g. IETF RPL)
- Support for use of link-layer mesh (e.g. 802.15.5)
Internet Protocol v6

- IPv6 (RFC 2460) = the next generation Internet Protocol
  - Complete redesign of IP addressing
  - Hierarchical 128-bit address with decoupled host identifier
  - Stateless auto-configuration
  - Simple routing and address management

- Majority of traffic not yet IPv6 but...
  - Most PC operating systems already have IPv6
  - Governments are starting to require IPv6
  - Most routers already have IPv6 support
  - So the IPv6 transition is coming
    - 1400% annual growth in IPv6 traffic (2009)
IPv4 vs. IPv6 Addressing

An IPv4 address (dotted-decimal notation)

172.16.254.1

10101100.00010000.11111110.00000001

One byte = Eight bits

Thirty-two bits (4 * 8), or 4 bytes

An IPv6 address (in hexadecimal)

2001:0DB8:AC10:FE01:0000:0000:0000:0000

Zeroes can be omitted

2001:0DB8:AC10:FE01::


Image source: Indeterminant (Wikipeida)

GFDL

IPv4: $7 \times 10^{-6}$ [addresses/m²]

IPv6: $666 \times 10^{21}$ [addresses/m²]
IPv4 vs. IPv6 Header

Image source: Bino1000, Mkim (Wikipeida)  GFDL
6LoWPAN is an adaptation header format
- Enables the use of IPv6 over low-power wireless links
- IPv6 header compression
- UDP header compression

Format initially defined in RFC4944

Updated by draft-ietf-6lowpan-hc (work in progress)
6LowPAN Header Compression at a Glance

- Stateless compression
- Flow-independent compression
- Simple tricks on IPv6/UDP header

- Common values for header fields => compact forms
- Version is always 6
- Traffic Class and Flow Label are zero
- Payload Length always derived from L2 header
- Source and Destination Addresses can be elided (link-local) and/or compressed depending on the “context” of the transmission

<table>
<thead>
<tr>
<th>Ver</th>
<th>Traffic Class</th>
<th>Flow Label</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Payload Length</td>
<td>Next Header</td>
</tr>
<tr>
<td></td>
<td>Source Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Destination Address</td>
<td></td>
</tr>
</tbody>
</table>
IPv6 Addressing

- 128-bit IPv6 address = 64-bit prefix + 64-bit Interface ID (IID)
- The 64-bit prefix is hierarchical
  - Identifies the network you are on and where it is globally
- The 64-bit IID identifies the network interface
  - Must be unique for that network
  - Typically is formed statelessly from the interface MAC address
    - Called Stateless Address Autoconfiguration (RFC2462)
- There are different kinds of IPv6 addresses
  - Loopback (0::1) and Unspecified (0::0)
  - Unicast with global (e.g. 2001::) or link-local (FE80::) scope
  - Multicast addresses (starts with FF::)
  - Anycast addresses (special-purpose unicast address)
IPv6 addresses are compressed in 6LoWPAN

Prefix

- Addresses within 6LoWPAN typically contain common prefix
- Nodes typically communicate with one or few central devices
- Establish state (contexts) for such prefixes – only state maintenance
- Support for up to 16 contexts 6LoWPAN compresses IPv6 addresses by

Interface ID

- Typically derived from L2 addr during autoconfiguration
- Elide when Interface Identifier can be derived from L2 header
Addressing Example
The Header Compression header

- **TF** (Traffic Class and Flow Label)
  - 0: Carried Inline (ECN+DSCP+Flow), 1: ECN+Flow, 2: ECN+DSCP, 3: All zero
- **NH** (Next Header compression)
  - 0: Carried Inline, 1: Next Header is compressed
- **HLIM** (Hop Limit = Inline, 1, 64, 255)
  - 0: Carried Inline, 1: 1, 2: 64, 3: 255
- **CID** (Context Identifier Extension)
  - 0: No 1-byte CID identifier, 1: 1-byte identifier follows
- **SAC/DAC** (Source/Destination Address Compression)
  - 0: Stateless, 1: Context-based
- **SAM/DAM** (Source/Destination Address Mode)
  - 0: 16 bytes inline, 1: 8 bytes inline, 2: 2 bytes inline, 3: elided
- **M** (Multicast Destination)
  - 0: Destination is not multicast, 1: Destination is multicast

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>1</th>
<th>TF</th>
<th>NH</th>
<th>HLIM</th>
<th>CID</th>
<th>SAC</th>
<th>SAM</th>
<th>M</th>
<th>DAC</th>
<th>DAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

In-line IPv6 Header Bits
Traffic Class and Flow Label Compression

TF = 00

IPHC ECN DSCP rsv Flow Label Inline IPv6 Bits

TF = 01

IPHC ECN rsv Flow Label Inline IPv6 Bits

TF = 10

IPHC ECN DSCP Inline IPv6 Bits

TF = 11

IPHC Inline IPv6 Bits

[DSCP = 0]

[Flow Label = 0]

[ECN=0, DSCP=0, Flow Label=0]
Next Header and Hop limit compression

☐ Next Header Field

NH = 0

IPHC  Inline IPv6 Bits  Next Header  Inline IPv6 Bits  Uncompressed Next Header

NH = 1

IPHC  Inline IPv6 Bits  Inline IPv6 Bits  NHC  Uncompressed Next Header

☐ Hop Limit Field

HLIM = 00

IPHC  Inline IPv6 Bits  Hop Limit  Inline IPv6 Bits

HLIM = 01

IPHC  Inline IPv6 Bits  [Hop Limit = 1]

HLIM = 10

IPHC  Inline IPv6 Bits  [Hop Limit = 64]

HLIM = 11

IPHC  Inline IPv6 Bits  [Hop Limit = 255]
Addresses Compression

- **Source/Destination Address Compression modes**
  - **SAM/DAM = 00**
    - IPv6 Address Bits [0,127]
  - **SAM/DAM = 01**
    - IPv6 Address Bits [64,127] [64-bit prefix elided]
  - **SAM/DAM = 10**
    - Bits [112,127] [112-bit prefix elided]

- **Stateless Mode (SAC/DAC=0)**
  - Prefix is link-local (fe80::/10)

- **Context-Based (SAC/DAC=1)**
  - Prefix taken from stored contexts (up to 16 contexts)
  - CID = 0, use ContextID = 0
  - CID = 1, include 4-bit ContextID for source & destination
# Multicast Addresses Compression

<table>
<thead>
<tr>
<th>DAM = 00</th>
<th>0xFF</th>
<th>Flags</th>
<th>Scope</th>
<th>Group Identifier Bits [0,111]</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAM = 01</td>
<td>Flags</td>
<td>Scope</td>
<td>Group Identifier Bits [64,111]</td>
<td><em>(Solicited Node Mcast)</em></td>
</tr>
<tr>
<td>DAM = 10</td>
<td>Flags</td>
<td>Scope</td>
<td>Group Identifier Bits [80,111]</td>
<td><em>(All DHCP Servers/Relays)</em></td>
</tr>
<tr>
<td>DAM = 11</td>
<td>Group ID Bits [104,112]</td>
<td><em>Flags = 0, Scope = 2</em></td>
<td><em>(Link-local All-Nodes, All-Routers)</em></td>
<td></td>
</tr>
</tbody>
</table>
UDP Header Compression

- Compressed IPv6 Hdr
- Compressed UDP Hdr

"Compressed IPv6"
"How IPv6 is compressed"

"Compressed UDP"
"How UDP is compressed"
Assume common values for header fields and define compact forms:
- Ports within 61616 to 61632 (4 bits)
- Length derived from IPv6 Length
- Checksum may be elided if other integrity checks are in use (e.g. Ipsec)

- C (Checksum): 0: Inline, 1: Elide
- P (Ports):
  - 0: Inline
  - 1: Elide first 8 bits of Dest Port
  - 2: Elide first 8 bits of Source Port
  - 3: Elide first 12 bits of Source and Dest Ports
Link Local

<table>
<thead>
<tr>
<th>Link Hdr</th>
<th>FCF</th>
<th>DSN</th>
<th>DSTPAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>DST</td>
<td>00-17-3B-00-AA-BB-CC-DD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRC</td>
<td>00-17-3B-00-11-22-33-44</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPv6 Hdr</th>
<th>Traffic Class = 0</th>
<th>Flow Label = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ver</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Payload Length</td>
<td></td>
<td>Next Header = UDP</td>
</tr>
<tr>
<td>Source Prefix</td>
<td>fe80::/64</td>
<td>Hop Limit = 1</td>
</tr>
<tr>
<td>Source IID</td>
<td>0217:3B00:AABB:CCDD</td>
<td></td>
</tr>
<tr>
<td>Dest Prefix</td>
<td>fe80::/64</td>
<td></td>
</tr>
<tr>
<td>Dest IID</td>
<td>0217:3B00:1122:3344</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UDP Hdr</th>
<th>Source Port</th>
<th>Destination Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

802.15.4

48-byte UDP/IPv6 Hdr → 7 bytes
### Global Unicast

<table>
<thead>
<tr>
<th>Link Hdr</th>
<th>Len = 50</th>
<th>FCF</th>
<th>DSN</th>
<th>DSTPAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>DST</td>
<td>00-17-3B-00-AA-BB-CC-DD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRC</td>
<td>00-17-3B-00-11-22-33-44</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPv6 Hdr</th>
<th>Ver = 6</th>
<th>Traffic Class = 0</th>
<th>Flow Label = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payload Length</td>
<td>Next Header = UDP</td>
<td>Hop Limit = 23</td>
<td></td>
</tr>
<tr>
<td>Source Prefix</td>
<td>2001:5a8:4:3721::/64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source IID</td>
<td>::1234</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dest Prefix</td>
<td>2001:5a8:4:3721::/64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dest IID</td>
<td>::ABCD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UDP Hdr</th>
<th>Source Port</th>
<th>Destination Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td></td>
<td>Checksum</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>802.15.4</th>
<th>1</th>
<th>2</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>1</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP-HC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hop Limit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source Addr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dest Addr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UDP-HC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

48-byte UDP/IPv6 Hdr → 12 bytes
Fragmentation

- IPv6 requires a minimum 2-PDU of 1280 bytes
- 802.15.4 has a maximum payload of 127 bytes
- IPv6 packets should be fragmented by 6LowPAN adaptation layer

<table>
<thead>
<tr>
<th>802.15.4</th>
<th>Disp</th>
<th>IPv6 Datagram</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.15.4</td>
<td>Frag</td>
<td>IPv6 Datagram (Frag 1)</td>
</tr>
<tr>
<td>802.15.4</td>
<td>Frag</td>
<td>IPv6 Datagram (Frag 2)</td>
</tr>
<tr>
<td>802.15.4</td>
<td>Frag</td>
<td>IPv6 Datagram (Frag N)</td>
</tr>
</tbody>
</table>
- **dgram_size**: size of the fragment in bytes
- **dgram_tag**: fragmentation ID (common to all fragments)
- **dgram_offset**: fragmentation offset (word of 8 bytes). Elided in the 1\(^{\text{st}}\) fragment
Fragmentation in Practice

- The performance of large IPv6 packets fragmented over low-power wireless mesh networks is poor!
  - Lost fragments cause whole packet to be retransmitted
  - Low-bandwidth and delay of the wireless channel
  - 6LoWPAN application protocols should avoid fragmentation
  - Fragmentation handled at the application layer (COAP)
  - Compression should be used on existing IP application protocols when used over 6LoWPAN if possible
6LoWPAN Setup & Operation

☐ Autoconfiguration is important in embedded networks

☐ In order for a 6LoWPAN network to start functioning:
  ■ Link-layer connectivity between nodes (commissioning)
  ■ Network layer address configuration, discovery of neighbors, registrations (bootstrapping)
  ■ Routing algorithm sets up paths (route initialization)
  ■ Continuous maintenance
IPv6 Neighbor Discovery

- IPv6 is the format - ND is the brains
  - “One-hop routing protocol” defined in RFC4861
- Defines the interface between neighbors
- Finding Neighbors
  - Neighbor Solicitation / Neighbor Acknowledgement
- Finding Routers
  - Router Solicitation / Router Advertisement
- Address resolution using NS/NA
- Detecting Duplicate Addresses using NS/NA
- Neighbor Unreachability Detection using NS/NA
- DHCPv6 may be used in conjunction with ND
IPv6 Neighbor Discovery

- Host
  - Router Solicitation
  - Router Advertisement
  - Neighbor Solicitation
  - Neighbor Advertisement

- Router
  - Neighbor Advertisement
  - Neighbor Solicitation

- DAD
  - Neighbor Advertisement
  - Neighbor Solicitation

- Address Resolution
  - Neighbor Advertisement
  - Neighbor Solicitation
6LoWPAN Neighbor Discovery

- Standard ND for IPv6 is not appropriate for 6LoWPAN:
  - Assumption of a single link for an IPv6 subnet prefix
  - Assumption that nodes are always on
  - Heavy use of multicast traffic (broadcast/flood in 6LoWPAN)
  - No efficient multihop support over e.g. 802.15.4

- 6LoWPAN Neighbor Discovery provides:
  - An appropriate link and subnet model for low-power wireless
  - Minimized node-initiated control traffic
  - Node Registration (NR) and Confirmation (NC)
  - Duplicate Address Detection (DAD) and recovery
  - Support for extended Edge Router infrastructures

- ND for 6LoWPAN has been specified in draft-ietf-6lowpan-nd (work in progress)
In normal IPv6 networks RAs are sent to a link based on the information (prefix etc.) configured for that router interface.

In ND for 6LoWPAN RAs are also used to automatically disseminate router information across multiple hops.
Node Registration

- 6LoWPAN-ND Optimizes only the **host-router** interface
  - RFC4861 = signaling between all neighbors (distributed)
- Nodes register with their neighboring routers
  - Exchange of NR/NC messages
  - Binding table of registered nodes kept by the router
- Node registration exchange enables
  - Host/router unreachability detection
  - Address resolution (a priori)
  - Duplicate address detection
- Registrations are soft bindings
  - Periodically refreshed with a new NR message
## NR/NC Format

<table>
<thead>
<tr>
<th>Type (NR)/(NC)</th>
<th>Code</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>TID</td>
<td>Status</td>
<td>P</td>
</tr>
<tr>
<td>Binding Lifetime</td>
<td>Advertising Interval</td>
<td></td>
</tr>
<tr>
<td>Owner Interface Identifier</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Owner Nonce</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Registration option(s)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Typical 6LoWPAN-ND Exchange
What should be the Optimum “Ideal” Routing Protocol for WSNs

- Shortest-path routes
- Avoids overlap
- Minimum energy consumption
- Needs global topology information
Algorithm classes
- Distance-vector
  *Links are associated with cost, used to find the shortest route. Each router along the path store local next-hop information about its route table.*
- Link-state
  *Each node acquires complete information about the network, typically by flooding. Each node calculated a shortest-path tree calculated to each destination.*

Types of Signaling
- Proactive
  *Routing information acquired before it is needed.*
- Reactive
  *Routing information discovered dynamically when needed.*

Route metrics are an important factor
Here we consider IP routing (at layer 3)

Routing in a LoWPAN
- Single-interface routing
- Flat address space (exact-match)
- Stub network (no transit routing)
Protocols for 6LoWPAN

- IP is agnostic to the routing protocol used
  - It forwards based on route table entries
- Thus 6LoWPAN is routing protocol agnostic
- Special consideration for routing over LoWPANs
  - Single interface routing, flat topology
  - Low-power and lossy wireless technologies
  - Specific data flows for embedded applications
- MANET protocols useful in some ad-hoc cases
  - e.g. AODV, DYMO
- New IETF working group formed
  - Routing over low-power and lossy networks (ROLL)
  - Developed specifically for embedded applications
IETF ROLL

- Routing Over Low power and Lossy networks (ROLL)
  - Working group at the IETF
- Standardizing a routing algorithm for embedded apps
- Application specific requirements
  - Home automation
  - Commercial building automation
  - Industrial automation
  - Urban environments
- Analyzed all existing protocols
- Solution must work over IPv6 and 6LoWPAN
- Protocol in-progress called RPL “Ripple”
  - Proactive distance-vector approach
Background

- Constrained devices! (few Kbytes of RAM, dozens of Kbytes of Flash, 8/16-bit microcontroller)
- Low-speed highly unstable lossy links (oscillation avoidance is a key)
- Potentially very large scale (10-100sK nodes)
- Unattended devices in harsh environments
RPL Phylosophy

General Principle: “Do Not Overact”
ROLL RPL “Ripple”
Requirements/Objectives

- Unicast/Anycast/Multicast
- Adaptive Routing
  - Different metrics
- Constraint-Based Routing
  - Parallel Paths
- Scalability
- Goal: RPL builds up a Destination-Oriented Directed Acyclic Graph (DODAG)
- What DODAG depends on the specific Objective Function (OF)
Metrics & Constraints

- Metric: scalar to capture the link/path performance (reliability, interference, throughput, etc.)
- Constraint: criteria to eliminate links from a DODAG
- Routing OF combines metric and constraints
  - Es: “find the path with the maximum reliability that does not traverse any non-encrypted link”
The Most Common Routing Metric

- **Node**
  - Residual Energy (node)
  - CPU, Storage, WorkLoad, Battery/Mains

- **Link**
  - Throughput (local/global metric)
  - Latency (local/global metric)
  - Reliability (local/global metric)
    - Expected Transmission Count (ETX)
    - Link Quality Level (LQL)

- **Hop Count**
The ETX

- “The average number of packet transmissions to successfully transmit a packet”

- Ex:

  - Being the packet error probability from A to B and from B to A $p$ and $q$, respectively, the ETX is: $1/[(1-p)(1-q)]$
  
  - $p$ and $q$ can be estimated through periodic beaconing
RPL Messages

- **DODAG Information Object (DIO)**
  - Link Local Messages to Advertise/Build Up DODAG
  - Contains all the information to exchange metrics/set constraints

- **Destination Advertisment Object (DAO)**
  - Used to propagate information on destination (prefixes)
  - Support to ptp and ptmp traffic

- **DODAG Information Solicitation (DIS)**
  - Used to solicit DIOs
DODAG Construction

Battery operated node
ETX (=1 when not indicated)
Step 1 and 2

Step 1: DIO message multicasted by the root

Step 2: LBR connections with numerical labels.
Step 3
Final DODAG

Resulting DAG computed by RPL

LBR

1.5

3

1.3

1.1

12

1.8

13

1.7

2

1.6

21

1.1

1.9

22

2.5

23

1.4

24

1.3

2.1

2.2

2.4

2.7

31

32

33

34

35

41

42

43

44

45

4
The very same network may support different applications

- **Es:** network with battery- and main-operated devices, high and low bandwidth links, and two applications (telemetry and alarming)
- One “time sensitive path” for alarms (low latency, high reliability, no constraints on energy)
- One “non-time-sensitive path” for telemetry energy optimized (do not traverse battery-operated nodes)
OF1: use the LQL as a global metric, minimize the number of low and fair quality links, avoid non-encrypted links.

OF2: find the best path in terms of latency while avoiding poor quality links and battery-operated nodes.
Taxonomy of Routing Protocols for Wireless Sensor Networks

- **DATA CENTRIC PROTOCOLS**
  - Flooding, Gossiping, SPIN, Directed Diffusion, SAR (Sequential Assignment Routing), Rumor Routing, Constrained Anisotropic Diffused Routing, COUGAR, ACQUIRE, ZigBee, RPL

- **HIERARCHICAL PROTOCOLS**
  - LEACH, PEGASIS, TEEN (Threshold Sensitive Energy Efficient Sensor Network Protocol)

- **LOCATION BASED (GEOGRAPHIC) PROTOCOLS**
  - MECN, SMECN (Small Minimum Energy Com Netw), GAF (Geographic Adaptive Fidelity), GEAR, Distributed Topology/Geographic,