



the sounds of smart environment



Streaming the Sound of Smart Cities: Experimentations on the SmartSantander test-bed

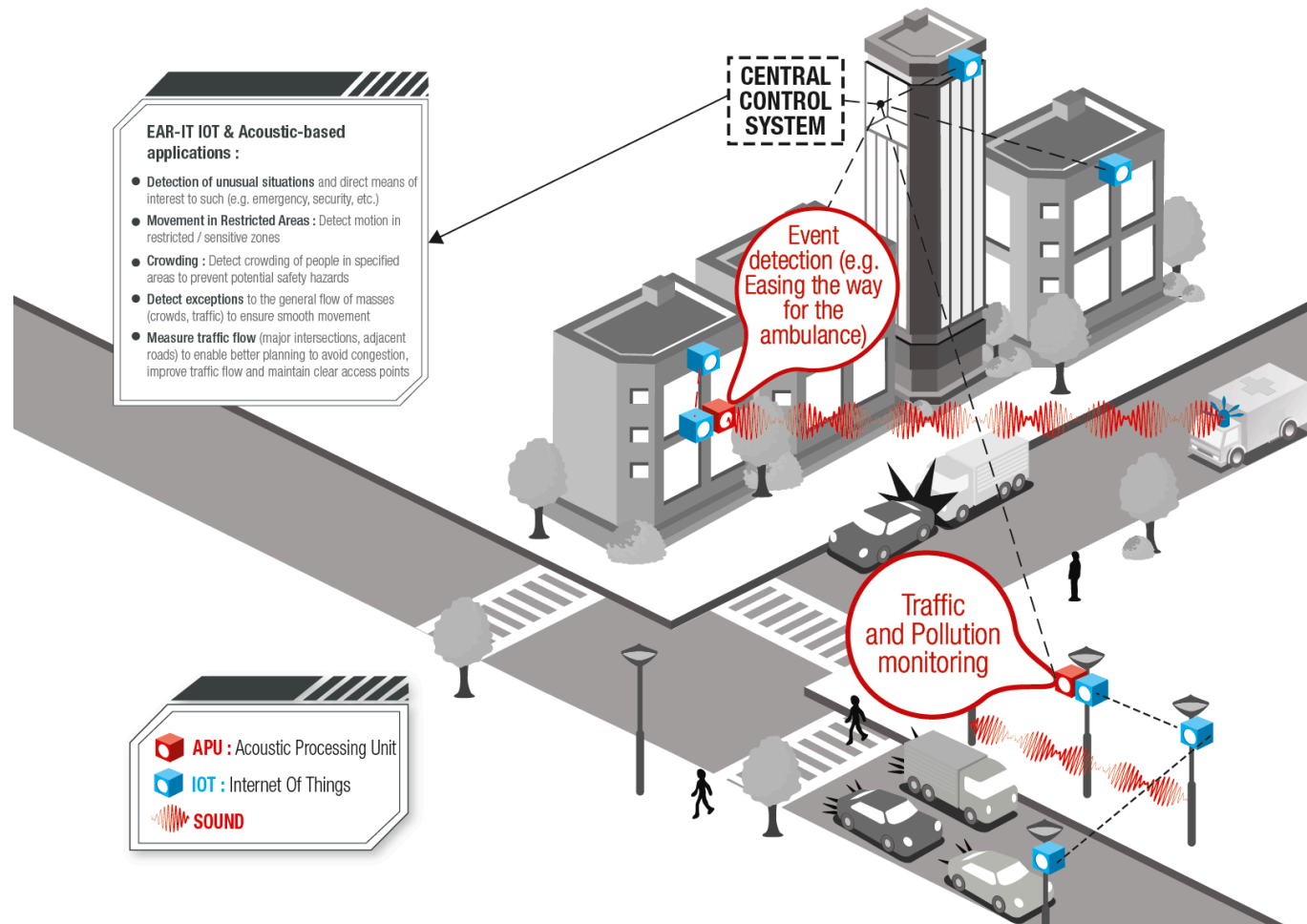
C. Pham (LIUPPA lab, University of Pau) and P. Cousin (EGM)

IEEE iThing 2013, Beijing, China, August 20-23, 2013



**MANDAT
INTERNATIONAL**







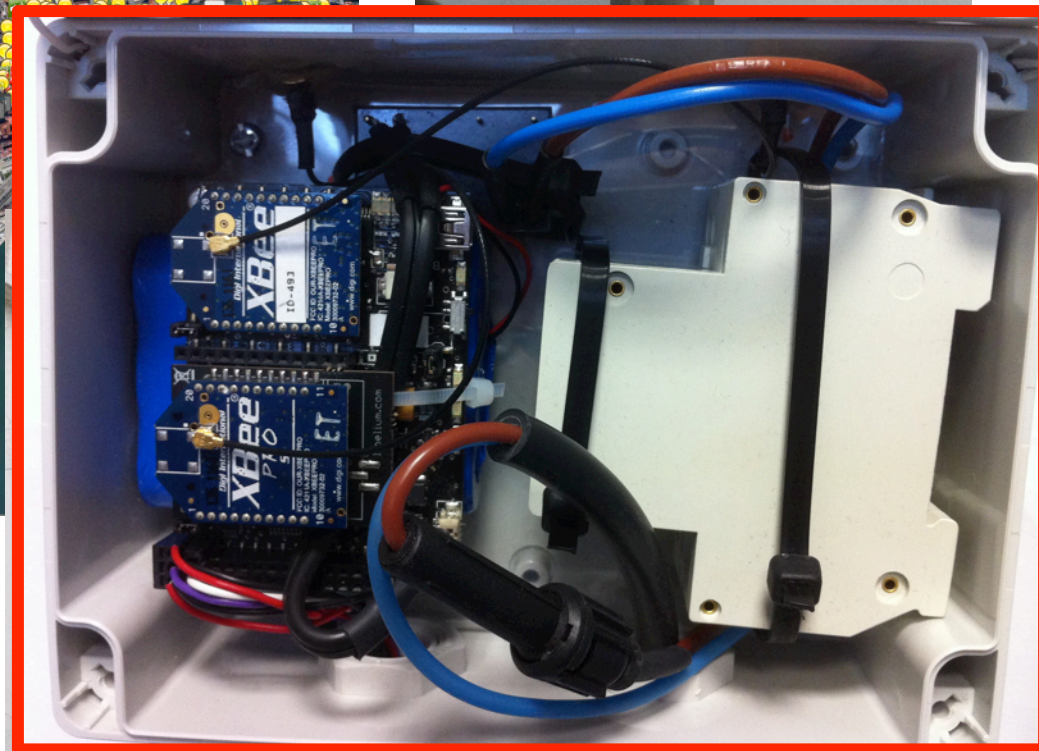
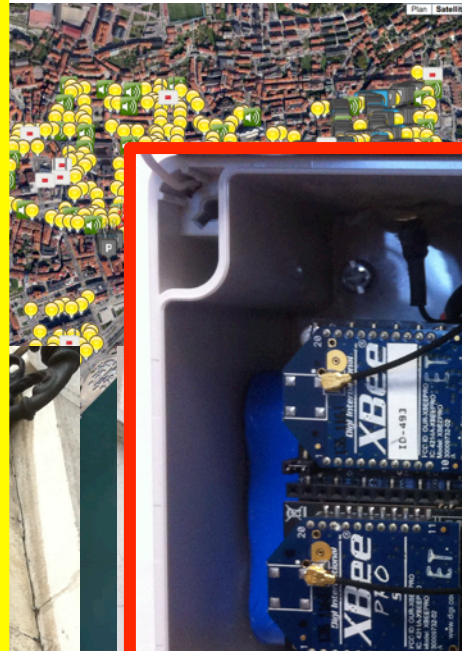






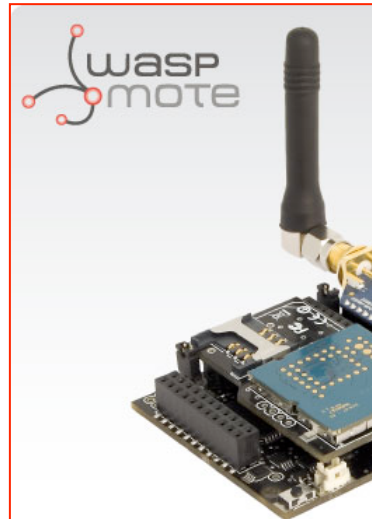


SmartSantander test-bed





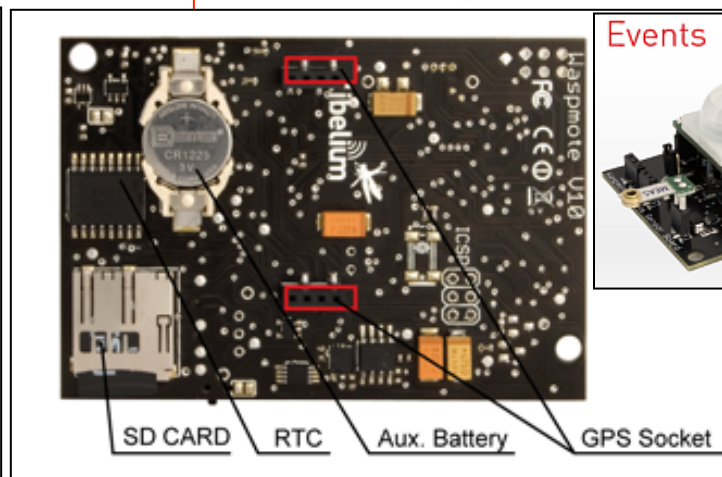
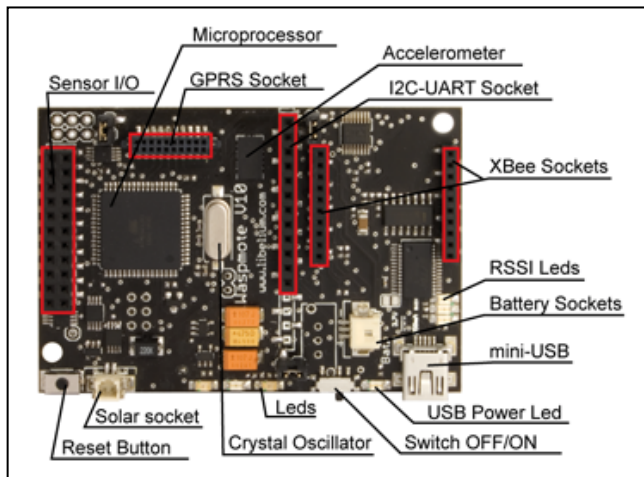
ATmega1281 microcontroller
8Mhz, 4K RAM & 2G SD card.
2.4GHz IEEE 802.15.4 XBee
Libelium API v031



Gases



- Carbon Monoxide – CO
- Carbon Dioxide – CO₂
- Oxygen – O₂
- Methane – CH₄
- Hydrogen – H₂
- Ammonia – NH₃
- Isobutane – C₄H₁₀
- Ethanol – CH₃CH₂OH
- Toluene – C₆H₅CH₃
- Hydrogen Sulfide – H₂S
- Nitrogen Dioxide – NO₂
- Temperature
- Humidity



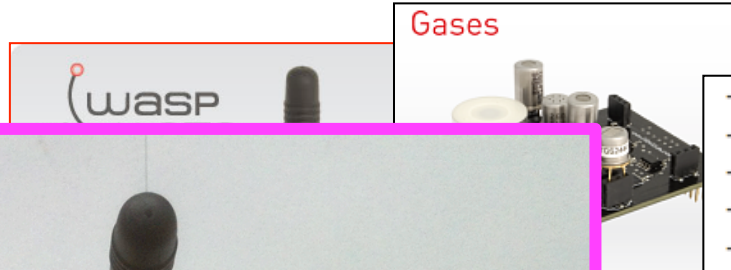
Events



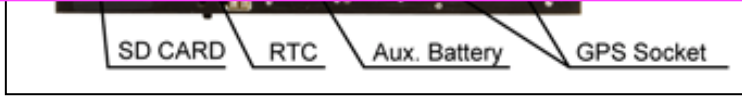
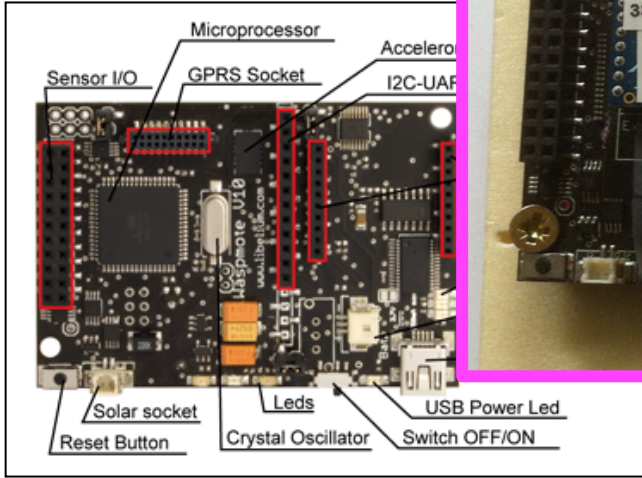
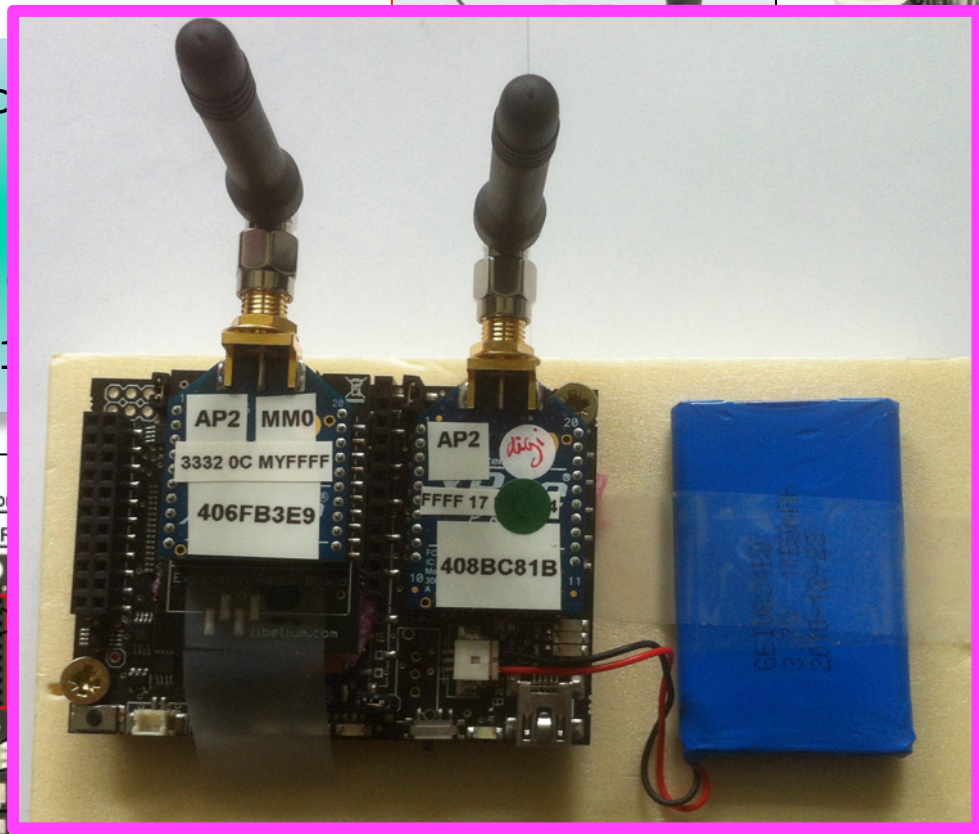
- Pressure/Weight
- Bend
- Vibration
- Impact
- Hall Effect
- Tilt
- Temperature (+/-)
- Liquid Presence
- Liquid Level
- Luminosity
- Presence (PIR)
- Stretch



ATmega1281 microcontroller
 8Mhz, 4K RAM & 128K ROM
 2.4GHz IEEE 802.15.4
 Libelium API v03.0



- Gases**
- Carbon Monoxide – CO
 - Carbon Dioxide – CO₂
 - Oxygen – O₂
 - Methane – CH₄
 - Hydrogen – H₂
 - Ammonia – NH₃
 - Isobutane – C₄H₁₀
 - Ethanol – CH₃CH₂OH
 - Toluene – C₆H₅CH₃
 - Hydrogen Sulfide – H₂S
 - Nitrogen Dioxide – NO₂
 - Temperature
 - Humidity



- Pressure/Weight
- Bend
- Vibration
- Impact
- Hall Effect
- Tilt
- Temperature (+/-)
- Liquid Presence
- Liquid Level
- Luminosity
- Presence (PIR)
- Stretch

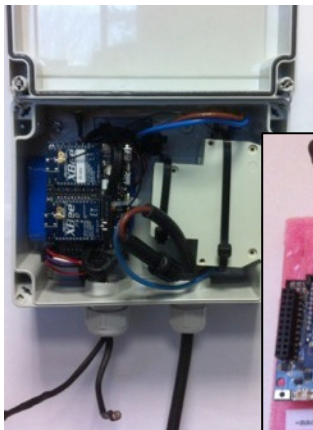


SmartSantander network qualification

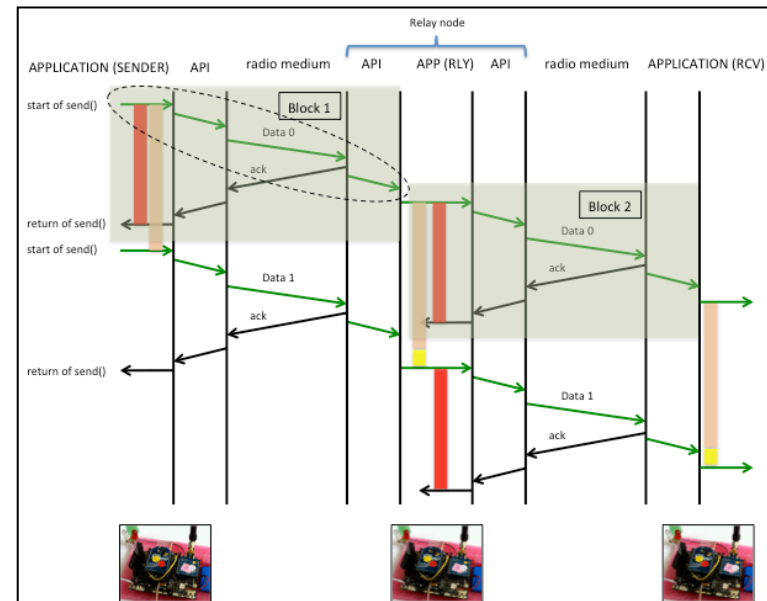
- What we have
 - Mesh configuration of IoT and gateways
 - IoT nodes rely on IEEE 802.15.4 radio
 - Radio modules are Xbee from Digi
- What we want to know
 - Upper bounds on sending and receiving throughput
 - Performances in a networked environment
 - Impact of API on performances
 - Where are the limitations?
 - To what extent audio traffic can be supported?

Phase 1: IoT node qualification

- Phase 1
 - Determine upper bounds on performances of a single IoT node
 - Determine upper bounds on performances of multi-hop transmissions



Traffic Generators
Sniffers
Advanced timing

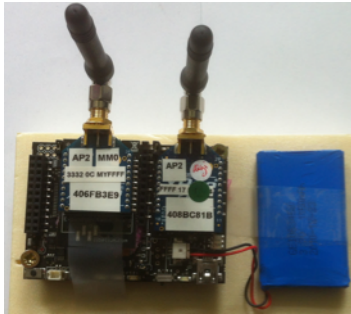


Phase 2: network qualification

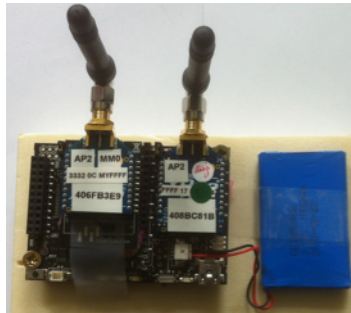
- Phase 2
 - Performances in a networked environment: node density, traffic loads



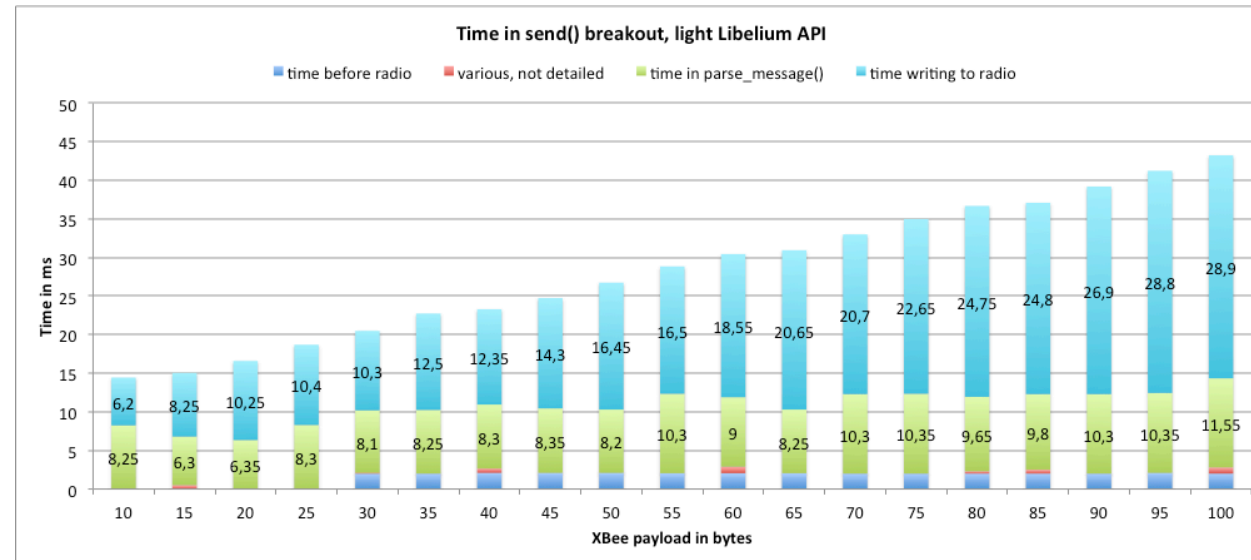
- Use representative locations in Santander for on-site test campaigns
- Deploy on IoT nodes traffic generators & sniffers
- Use mobile traffic generators & sniffers for dynamic traffic patterns
- Throughput, packet losses, latency,...

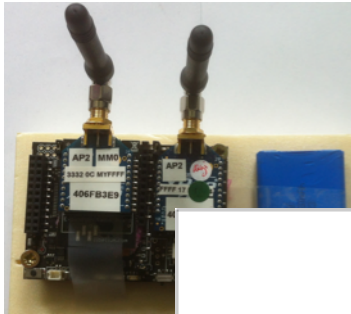


LIBELIUM WASPMOTE

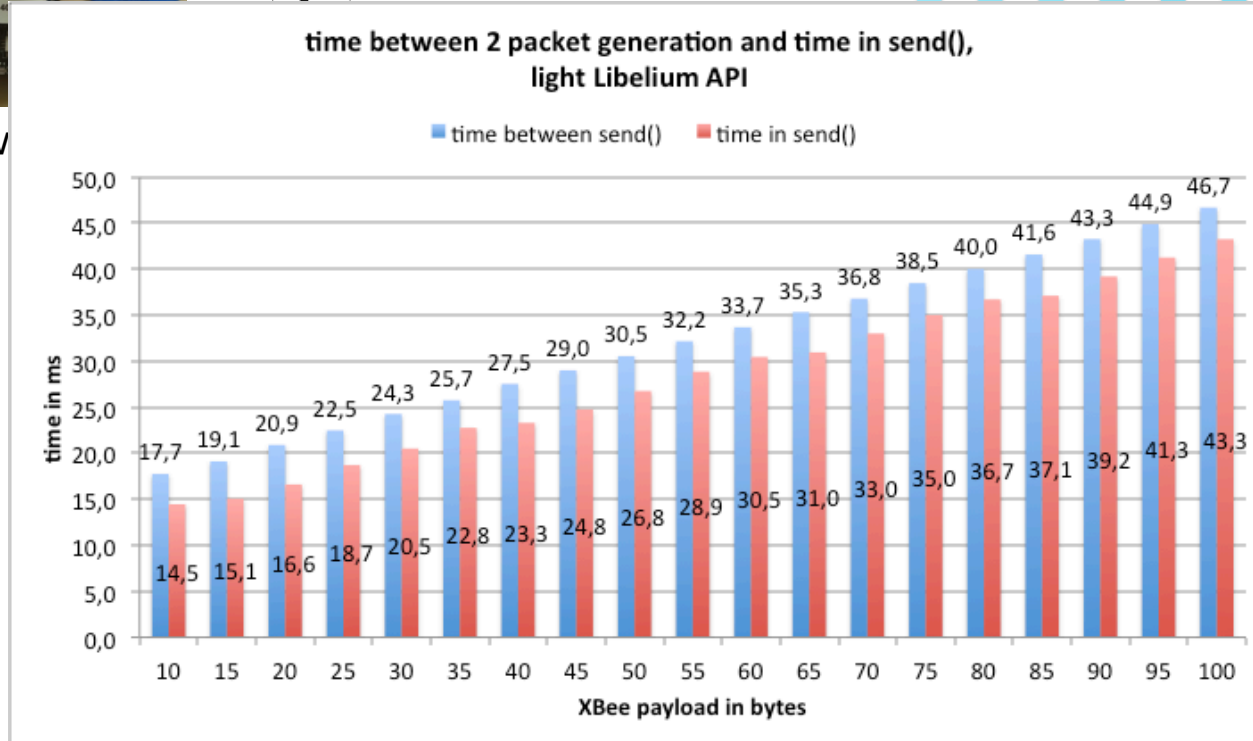
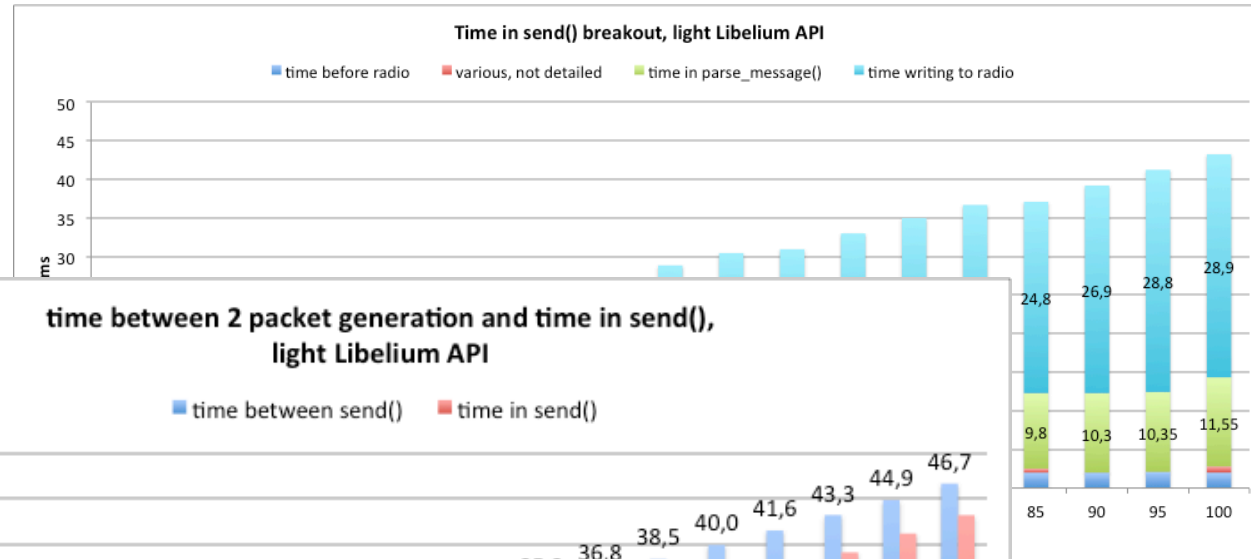


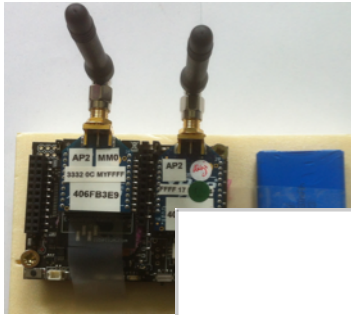
LIBELIUM WASPMOTE



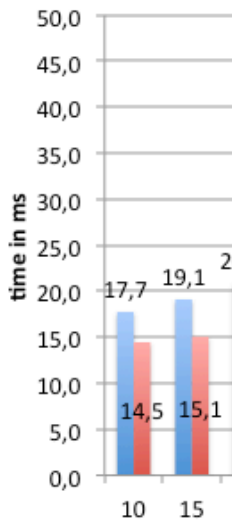
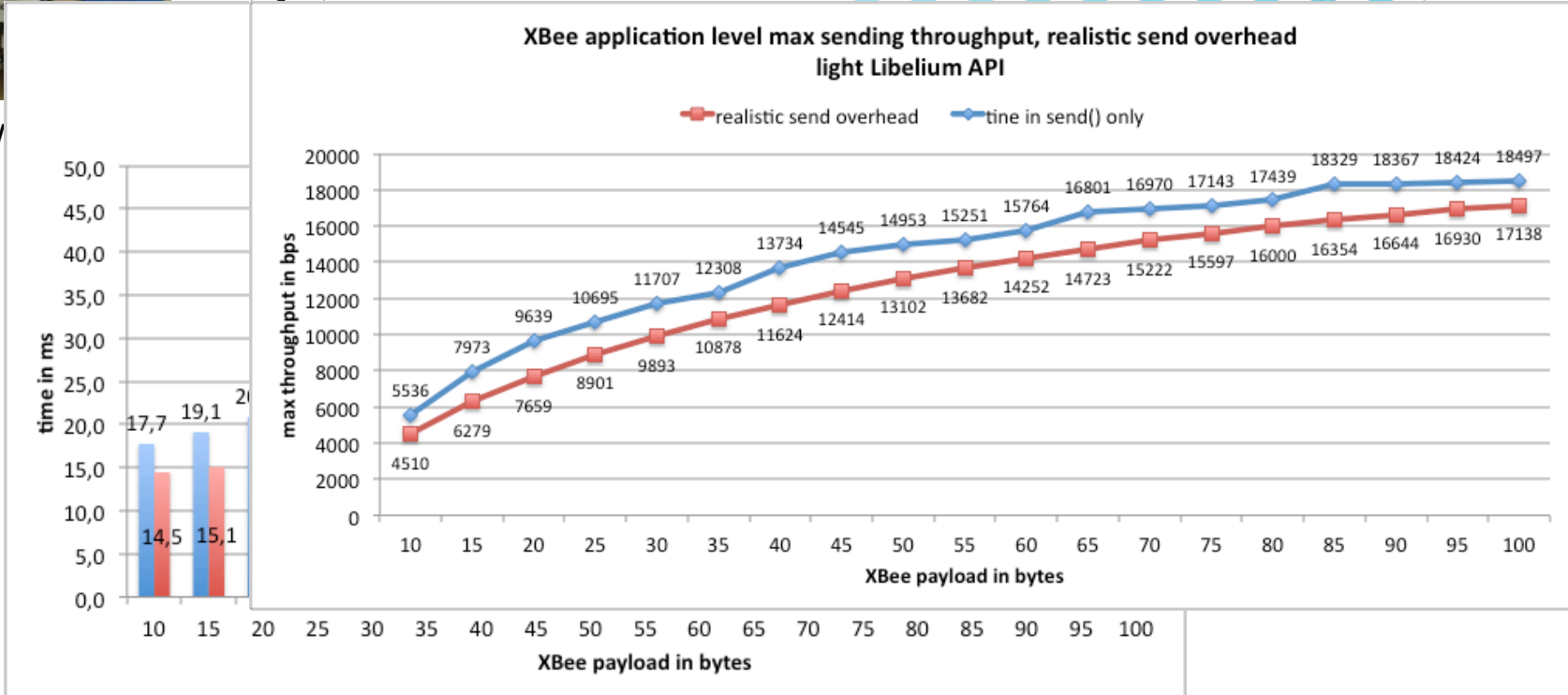
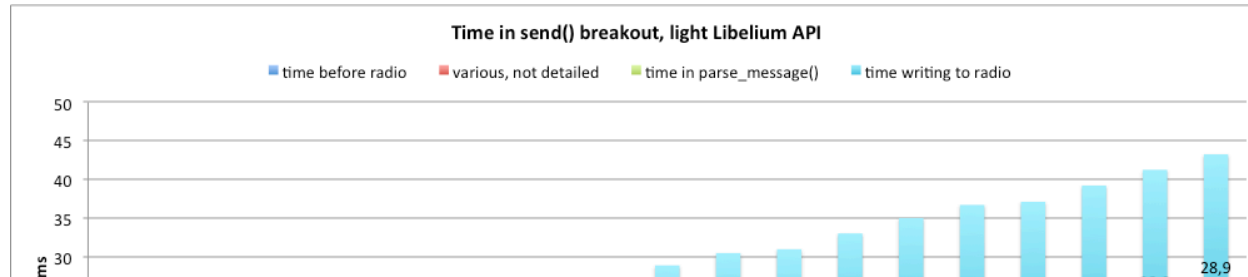


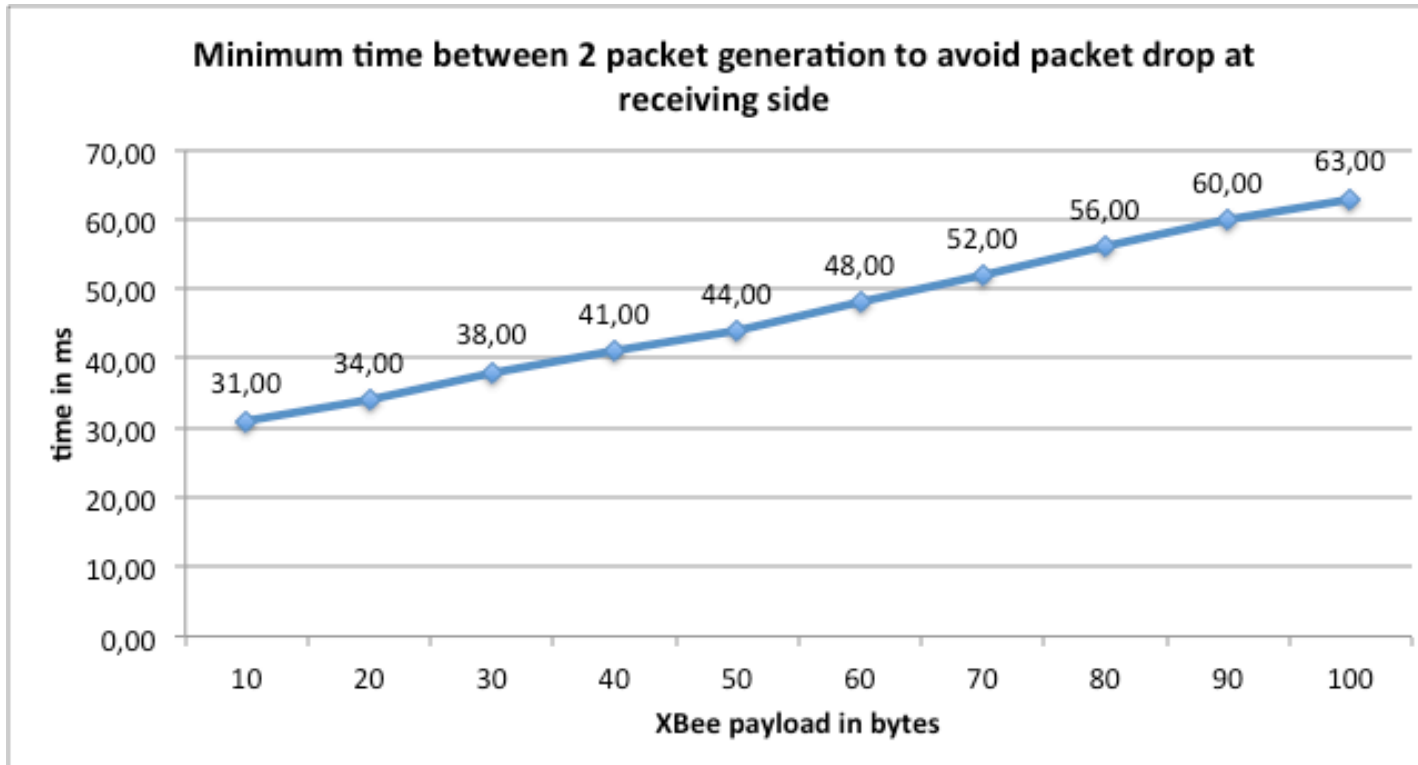
LIBELIUM V

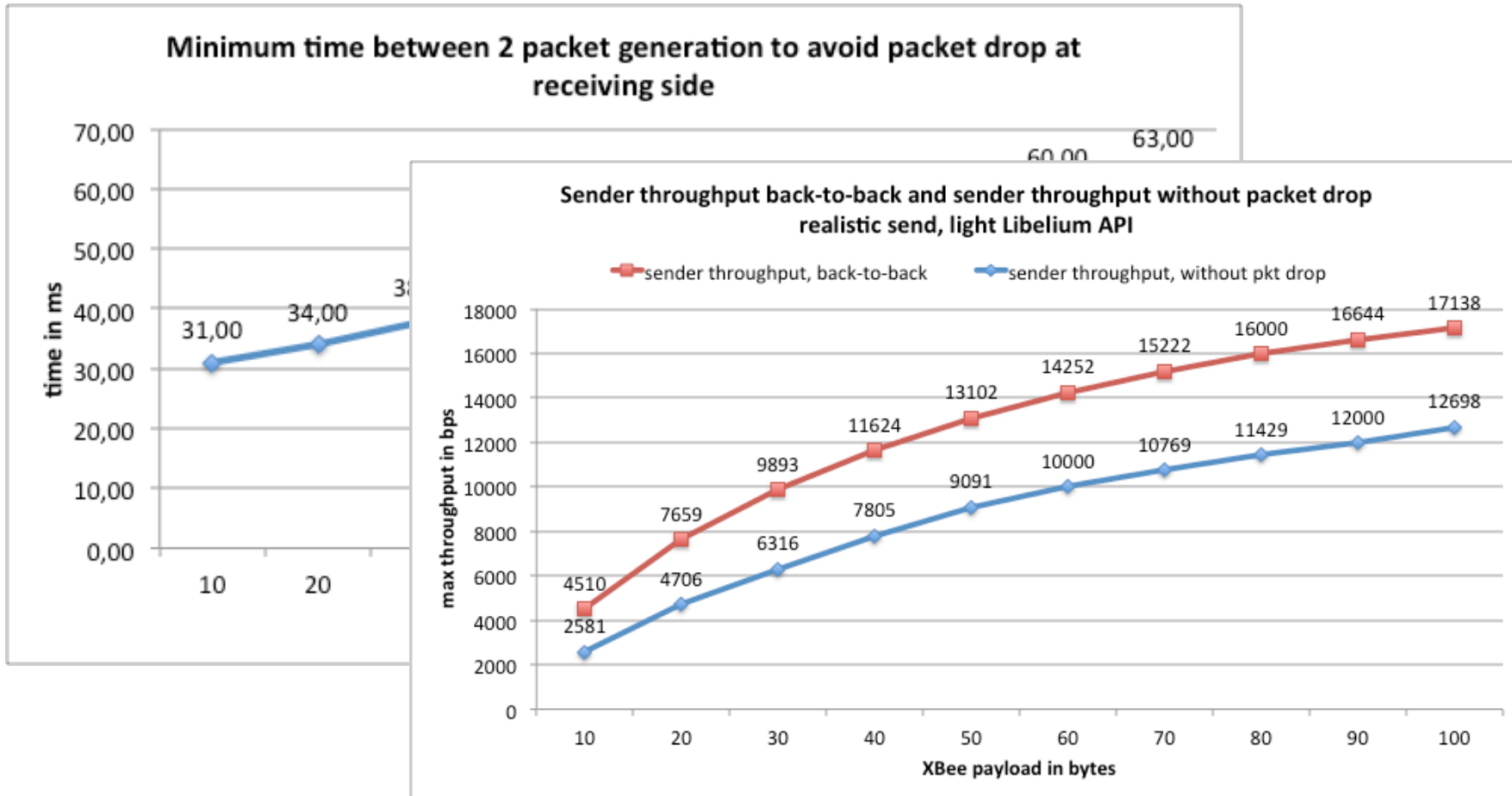




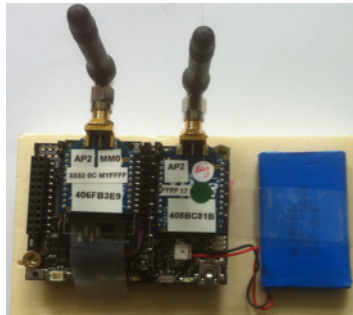
LIBELIUM V







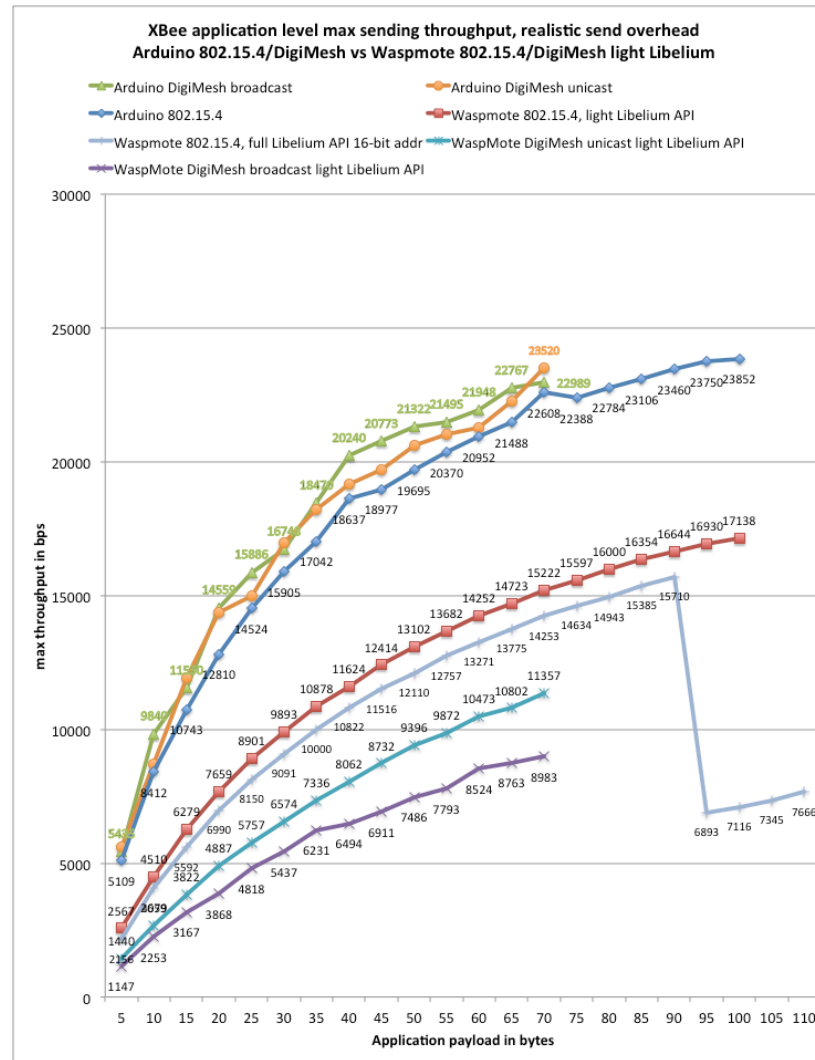
Technology comparison



LIBELIUM WASPMOTE



ARDUINO MEGA2560

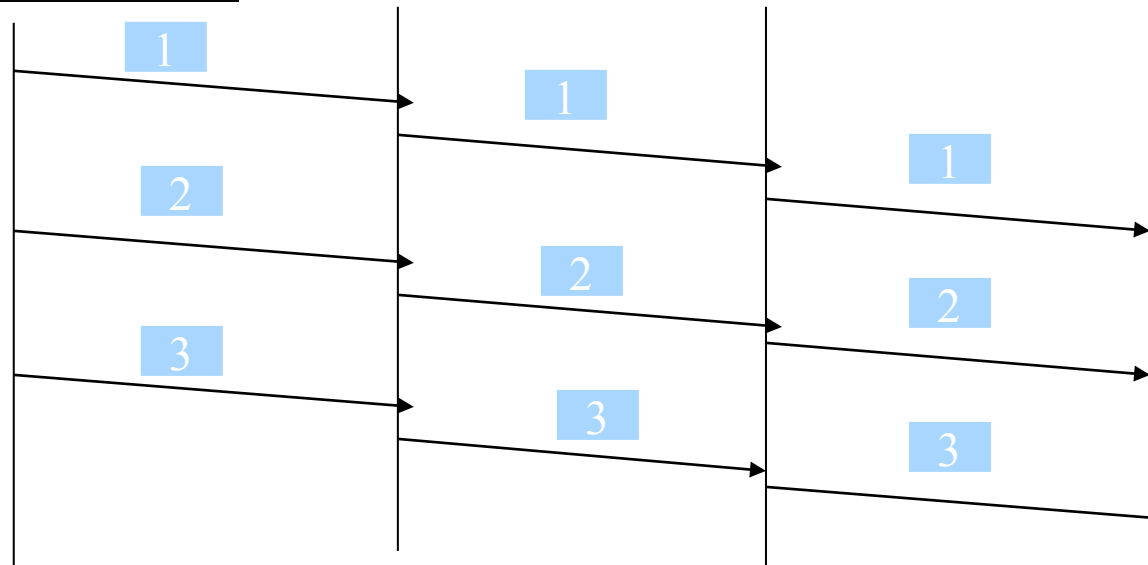
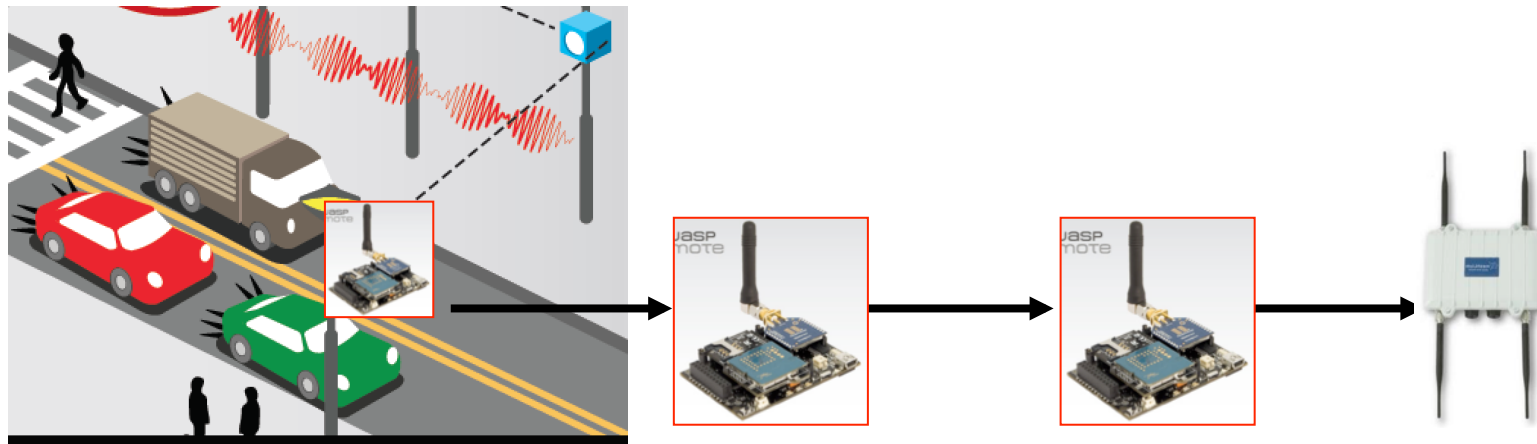


XBEE 802.15.4

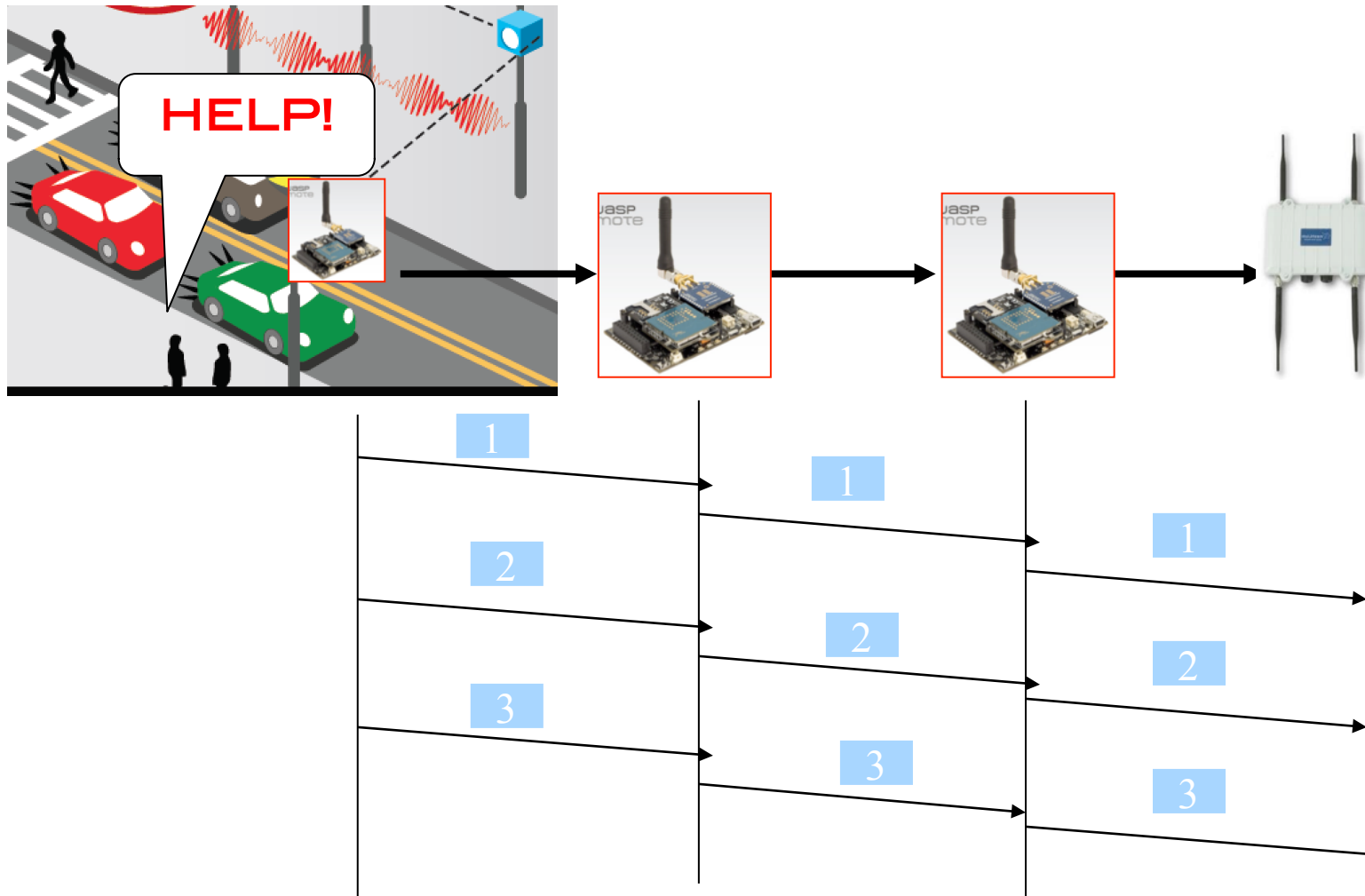


XBEE DIGIMESH

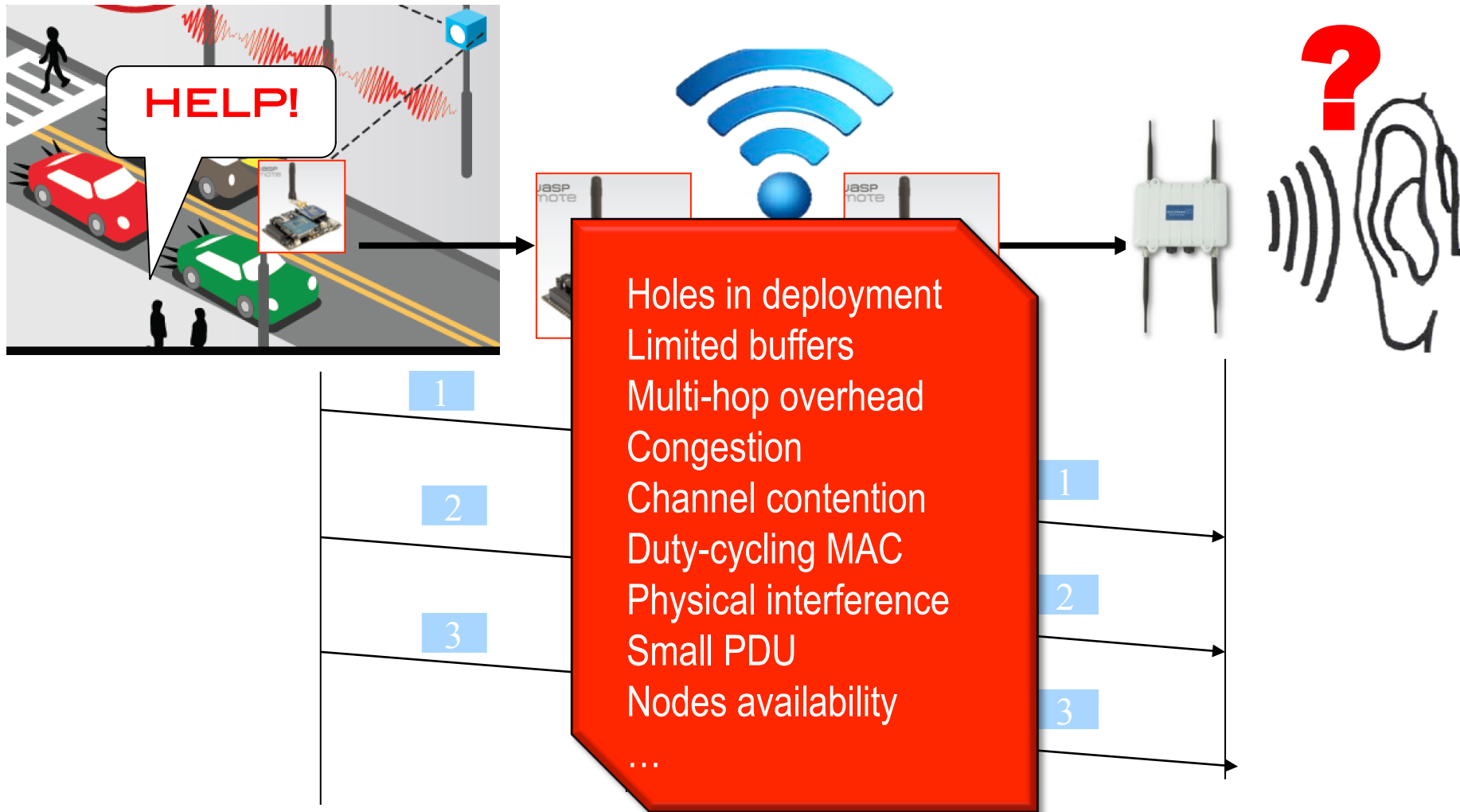
Multi-Hop Packet Forwarding?



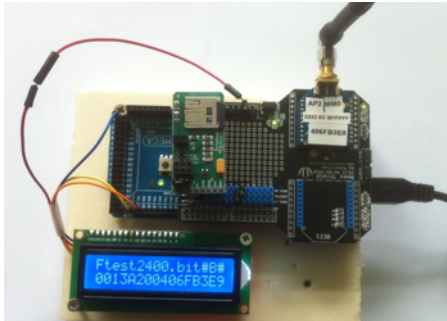
Multi-Hop Packet Forwarding?



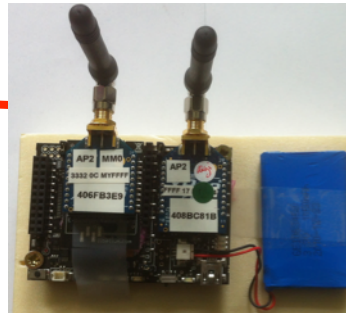
Multi-Hop Packet Forwarding?



Multi-hop audio

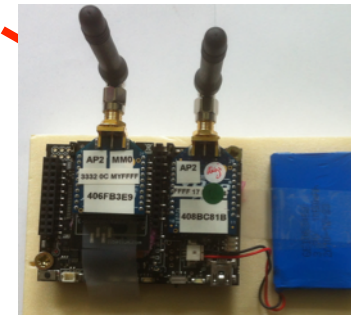


SENDS AN AUDIO FILE



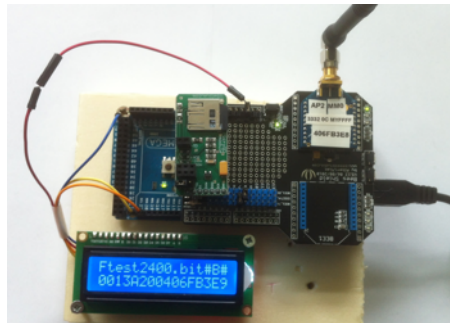
RELAY

RELAY

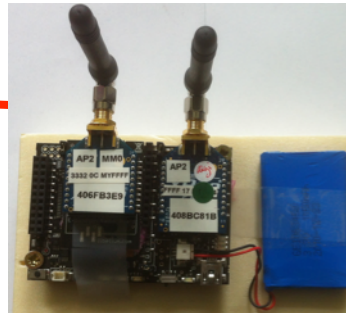


PLAY RECEIVED FILE

Multi-hop audio

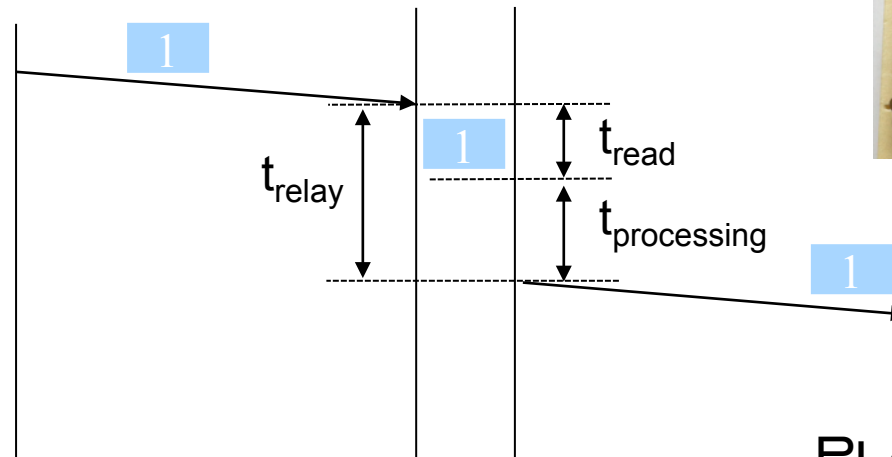
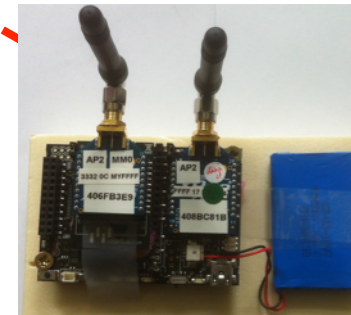


SENDS AN AUDIO FILE



RELAY

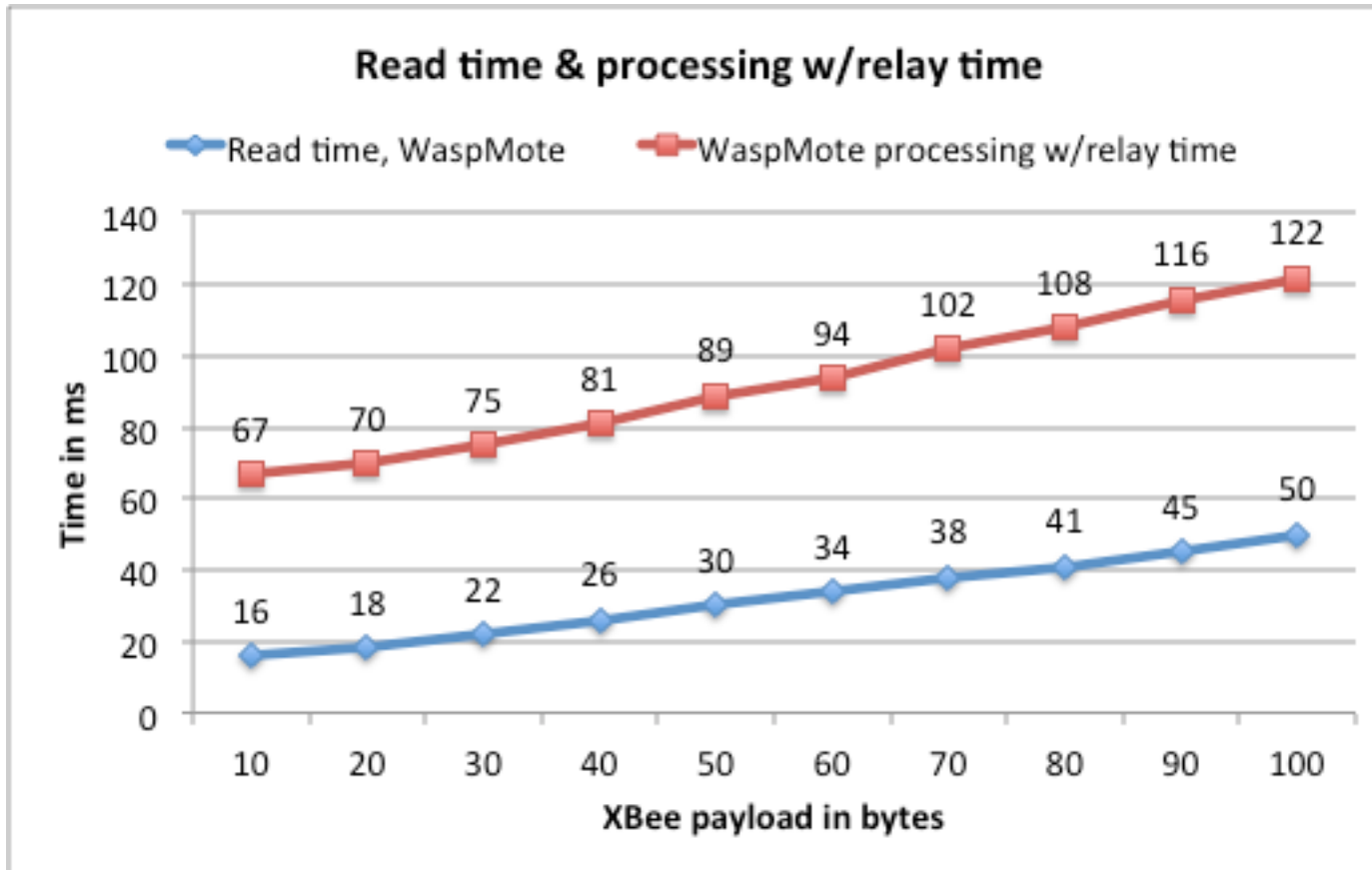
RELAY



PLAY RECEIVED FILE

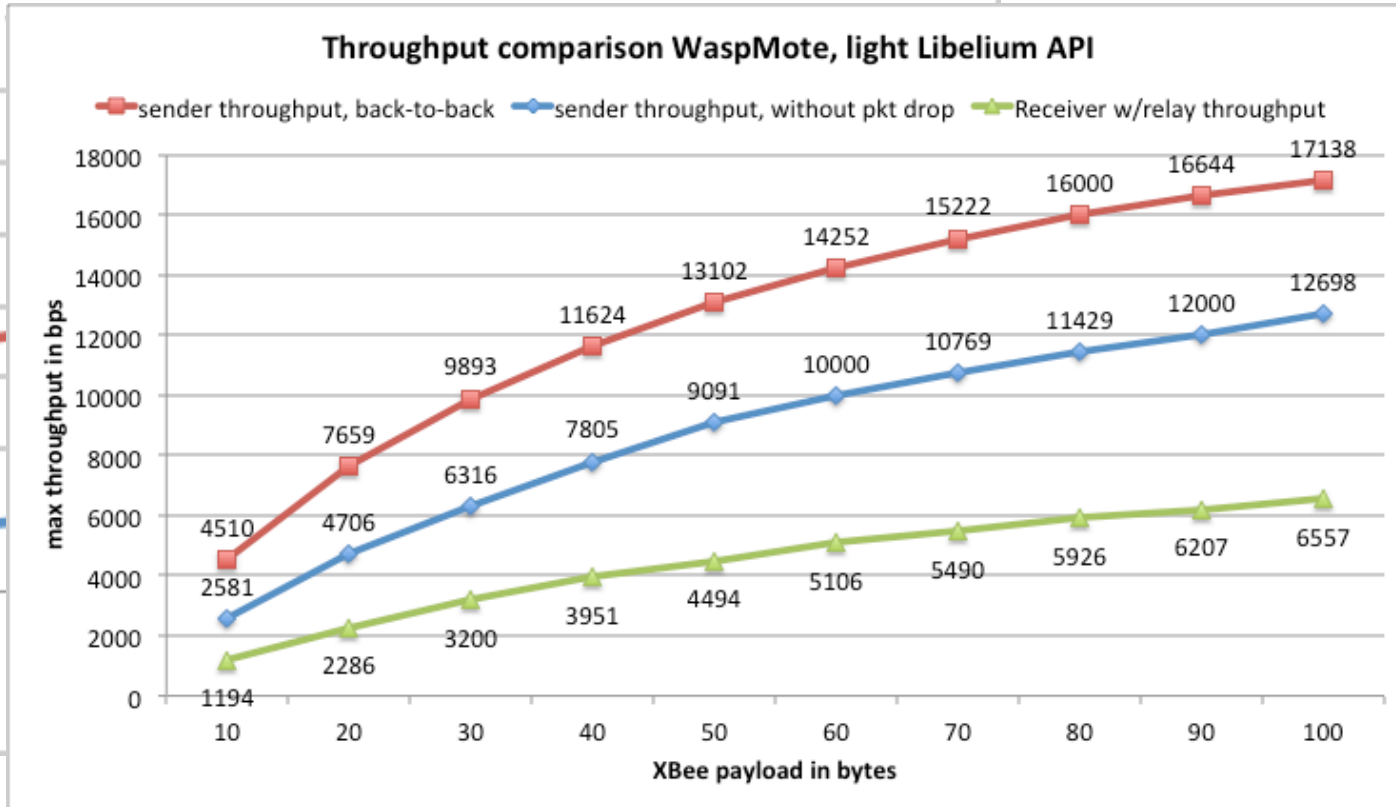
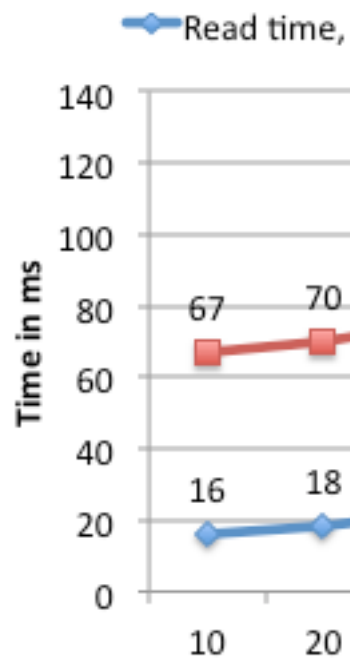


Multi-hop overheads



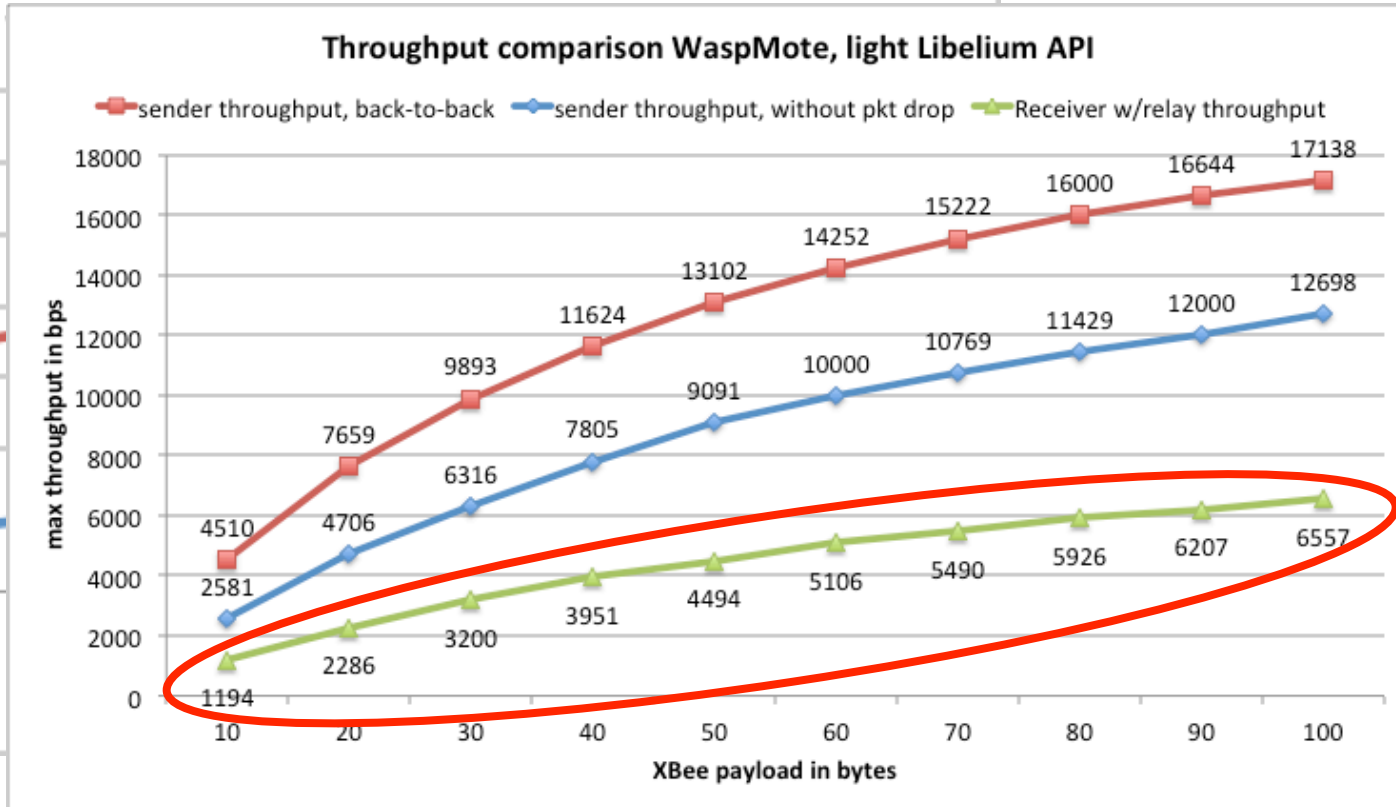
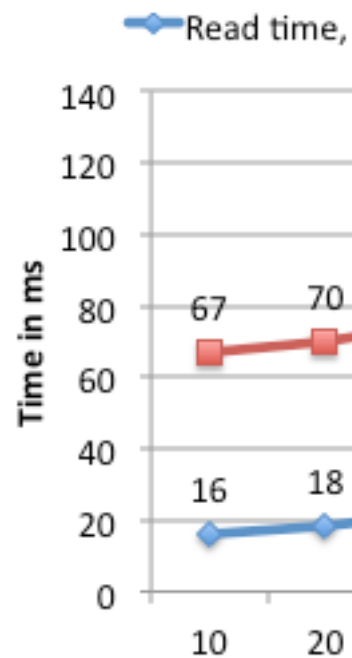
Multi-hop overheads

Read time & processing w/relay time



Multi-hop overheads

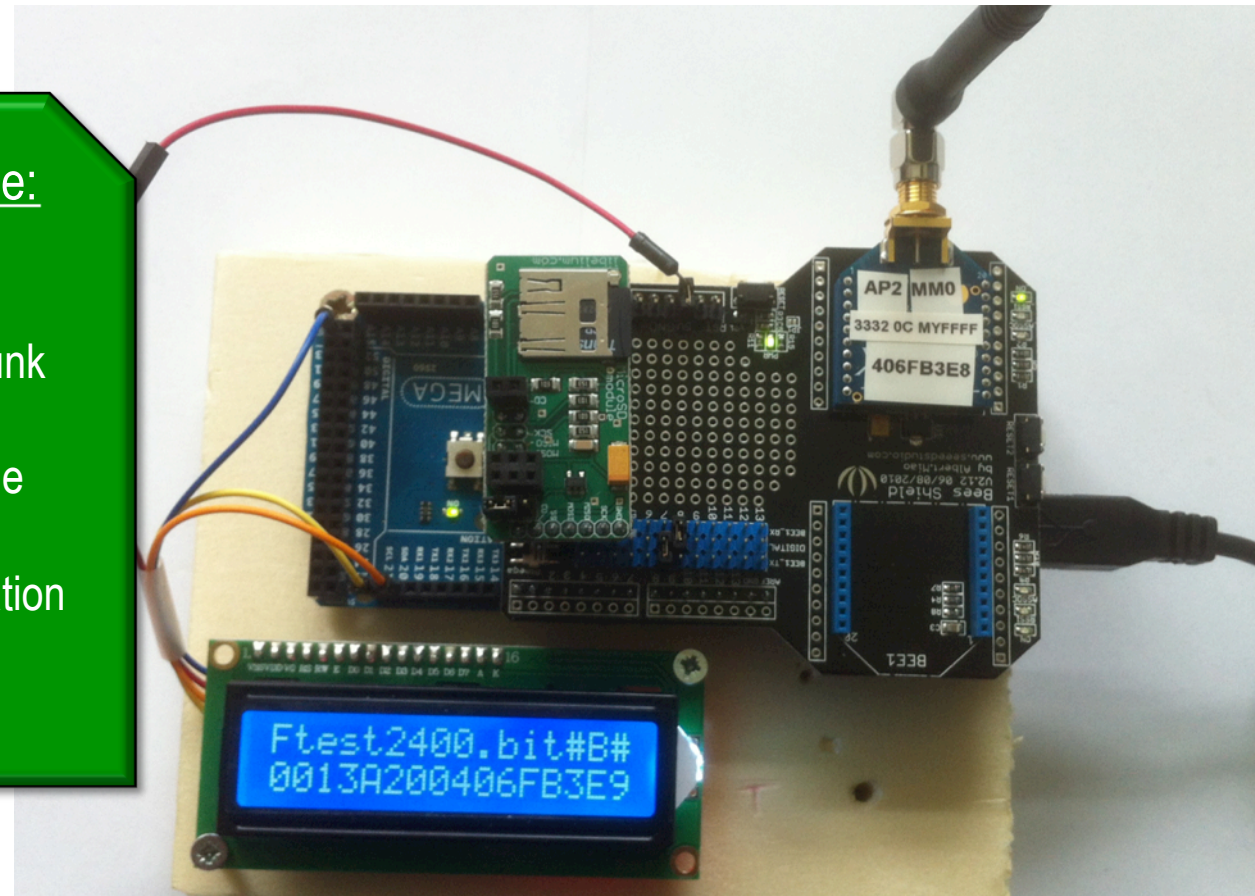
Read time & processing w/relay time



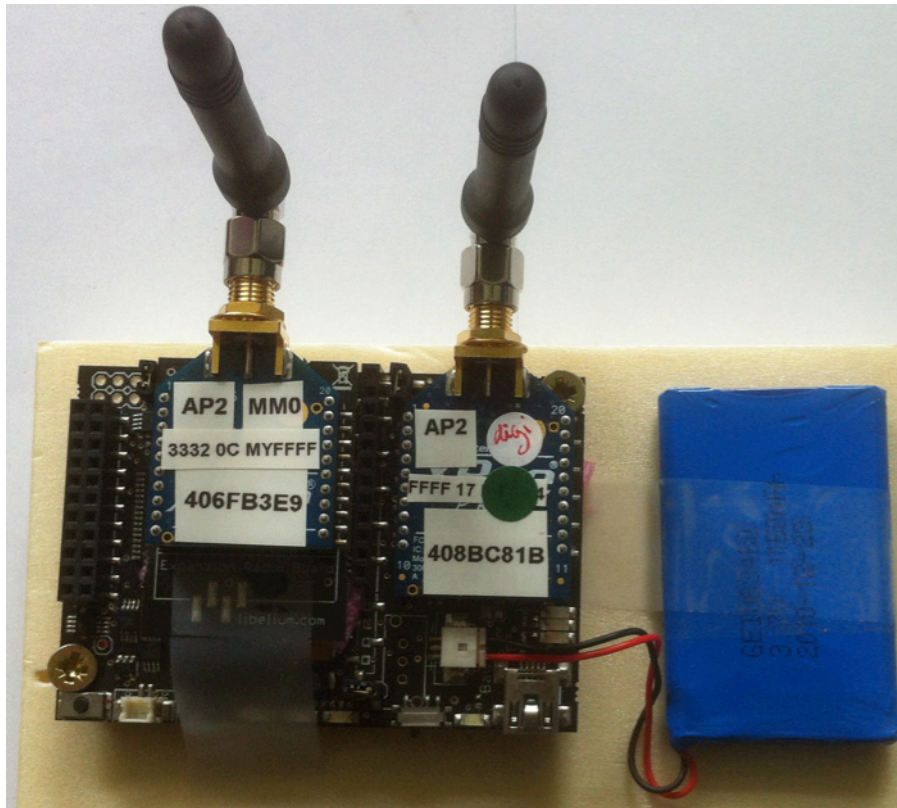
Sender node

Fully configurable:

- File to send
- Size of packet chunk
- Inter-packet delay
- Image/Binary mode
- Destination node
- Clock synchronization



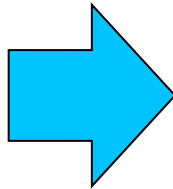
Relay nodes



LIBELIUM WASPMOTE

Fully configurable:

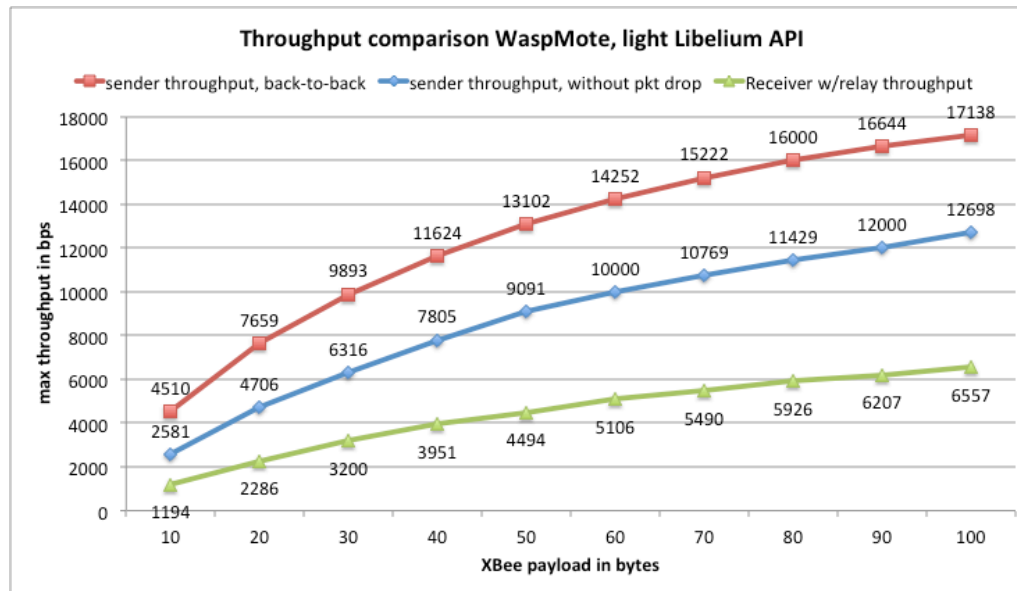
Destination node
Additional relay delay
Clock synchronization



LINUX PC/LAPTOP WITH
USB/SERIAL GATEWAY

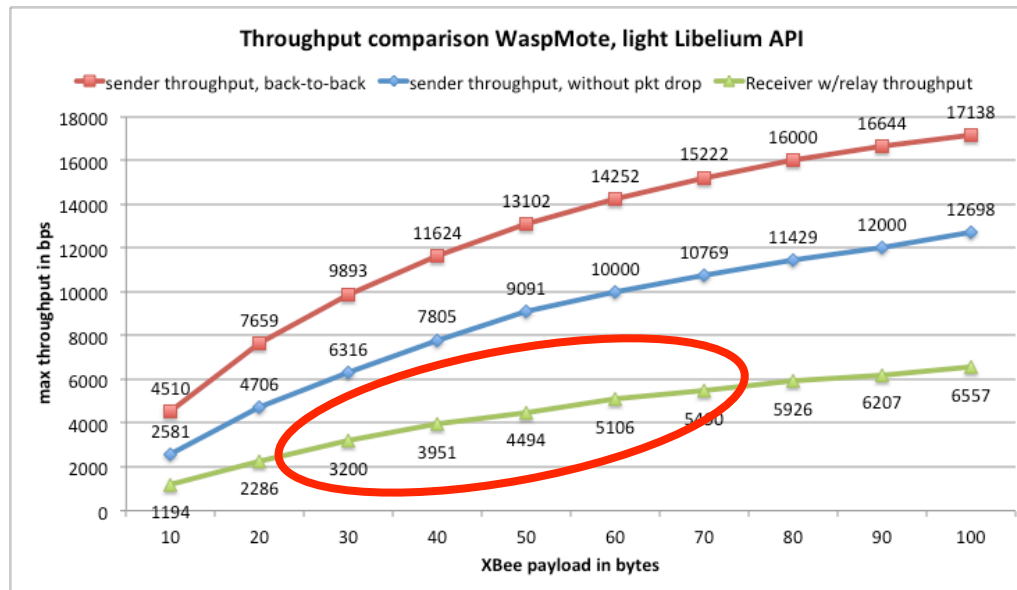


Audio encoding



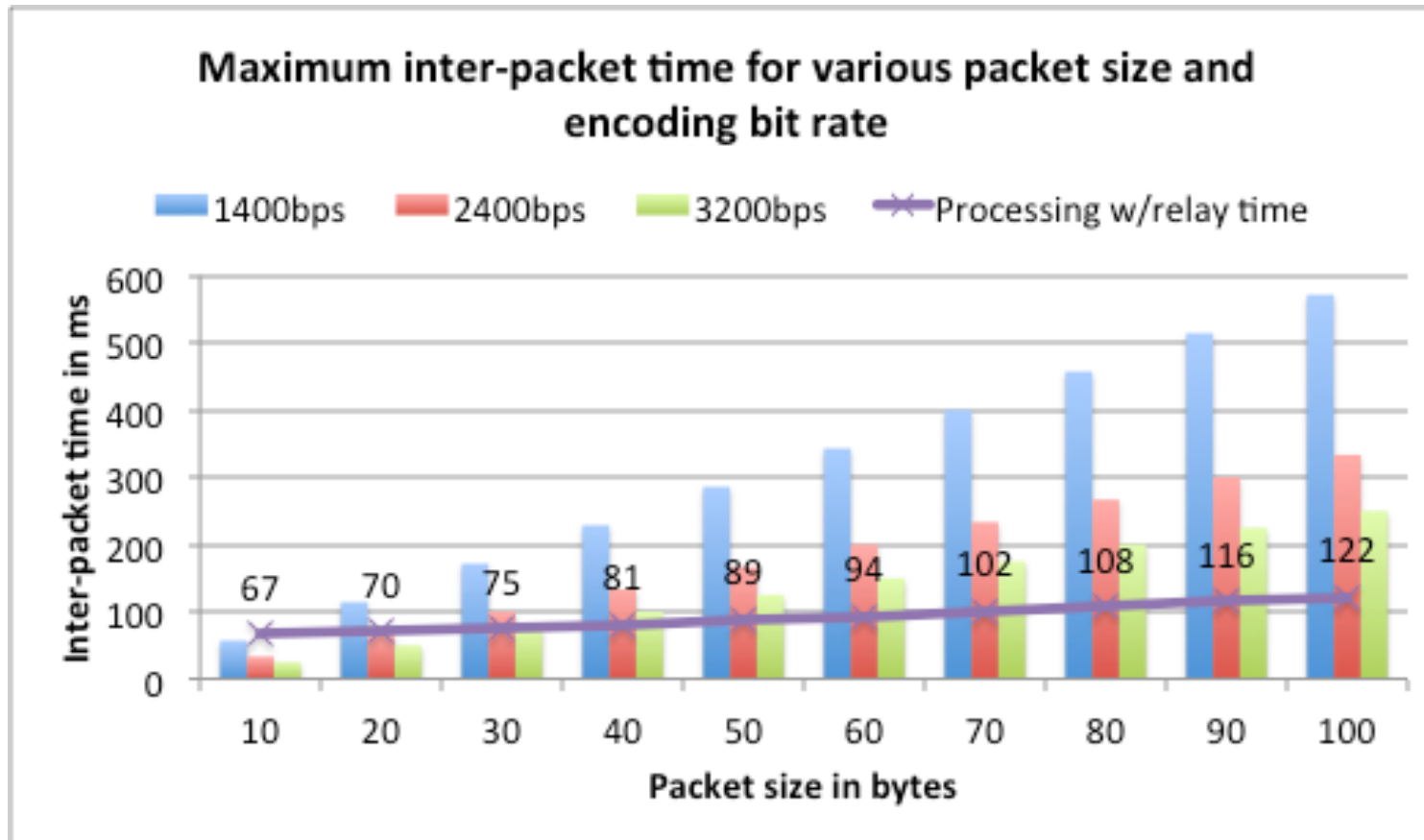
- Need a really low rate audio encoding scheme
- PCM is 64kbps, GSM 6.1 is 13kbps, can be lowered to 6kbps
- We use an open-source codec
 - codec2: <http://codec2.org>
 - Can be as low as 1400bps (1600, 2400 and 3400bps available)
 - All encoding/decoding tools are available in code source
 - Encoded file is robust against packet losses

Audio encoding



- Need a really low rate audio encoding scheme
- PCM is 64kbps, GSM 6.1 is 13kbps, can be lowered to 6kbps
- We use an open-source codec
 - codec2: <http://codec2.org>
 - Can be as low as 1400bps (1600, 2400 and 3400bps available)
 - All encoding/decoding tools are available in code source
 - Encoded file is robust against packet losses

Can we meet the constraints?







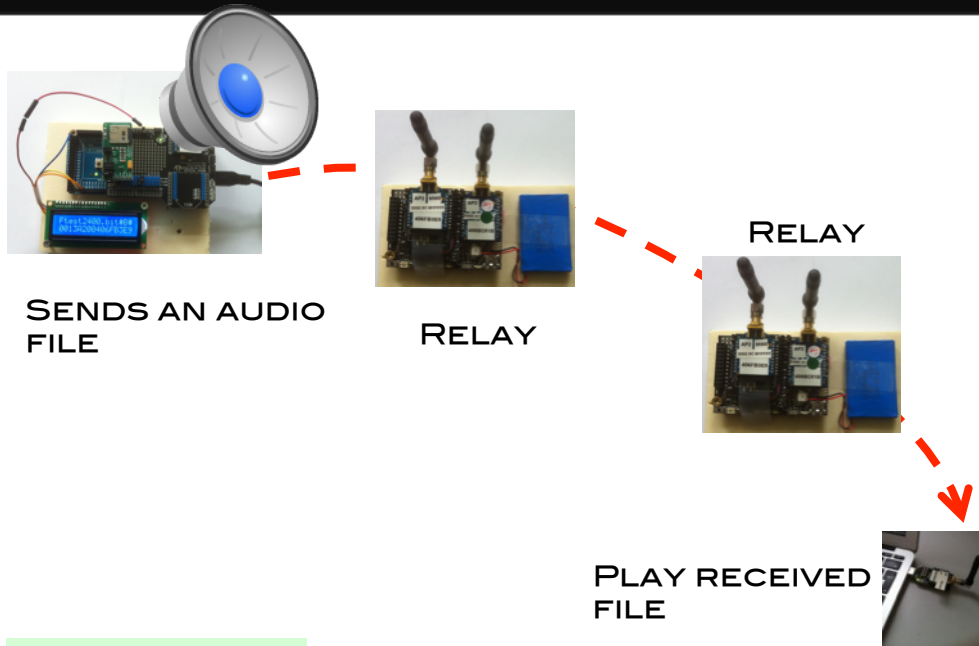


Test campaign – April 9th-10th 2013



the sounds of smart environments

Software for audio streaming



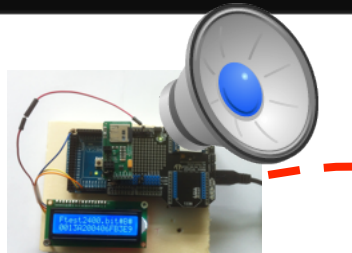
STORE & PLAY

```
> XBeeReceive -B test2400.bit
> c2dec 2400 -B test2400.bit - | play -t raw -r 8000 -s -2 -
```

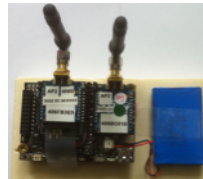
STREAMING

```
> XBeeReceive -B -stdout test2400.bit | bfr -b1k -m2% - | c2dec
2400 - - | play -t raw -r 8000 -s -2 -
```

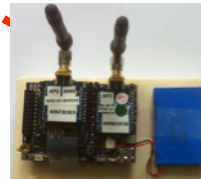
Software for audio streaming



SENDS AN AUDIO FILE



RELAY



RELAY

SAMPLE AUDIO: 13S
PCM = 104000B
CODEC2 AT 2400 IS 3900B



PLAY RECEIVED FILE

NEED TO REDUCE THE PACKET SIZE TO LIMIT IMPACT OF PACKET LOSSES

STORE & PLAY

```
> XBeeReceive -B test2400.bit
> c2dec 2400 -B test2400.bit - | play -t raw -r 8000 -s -2 -
```

STREAMING

```
> XBeeReceive -B -stdout test2400.bit | bfr -b1k -m2% - | c2dec
2400 - - | play -t raw -r 8000 -s -2 -
```

1-hop results

1-relay scenario									
bit rate	1400bps			2400bps			3200bps		
pkt size	40	50	60	40	50	60	40	50	60
n_{pkt}	59	47	39	101	81	67	134	108	90
t_{pkt}	105	110	120	105	110	120	105	110	120
n_{lost}	8	6	7	6	5	5	8	9	8
t_{pkt}	110	120	125	110	120	125	110	120	125
n_{lost}	1	0	0	0	2	2	3	1	3
t_s, s	6.5	5.6	4.8	11.1	9.7	8.3	14.7	14.4	11.2
t_{rcv}	6.9	6.4	5.2	11.6	10.1	8.8	15.4	15	11.7
t_{play}	4.7	4.5	3.7	8.4	8.2	6.1	13.1	12.8	9.8

“EAR-IT” at <http://web.univ-pau.fr/~cpham/SmartSantanderSample/>

2-hop results

2-relay scenario									
bit rate	1400bps			2400bps			3200bps		
pkt size	40	50	60	40	50	60	40	50	60
n_{pkt}	59	47	39	101	81	67	134	108	90
t_{pkt}	105	110	120	105	110	120	105	110	120
n_{lost}	9	7	7	7	7	7	8	8	10
t_{pkt}	110	120	125	110	120	125	110	120	125
n_{lost}	2	1	1	0	1	2	2	1	2
t_s, s	6.4	5.6	4.9	11.2	9.8	8.3	14.6	14.4	11.3
t_{rcv}	7.1	6.6	5.3	11.8	10.2	9	15.7	15.2	12
t_{play}	4.9	4.8	3.9	8.7	8.5	6.4	13.3	13	10.1

“EAR-IT” at <http://web.univ-pau.fr/~cpham/SmartSantanderSample/>

Conclusions

- Receiver throughput is low and a maximum of 8kbps can be achieved without packet losses
- Low bit rate codecs for voices can be streamed from source to gateway provided that contention on radio links is low
- Multi-source is challenging



the sounds of smart environment



Questions ?