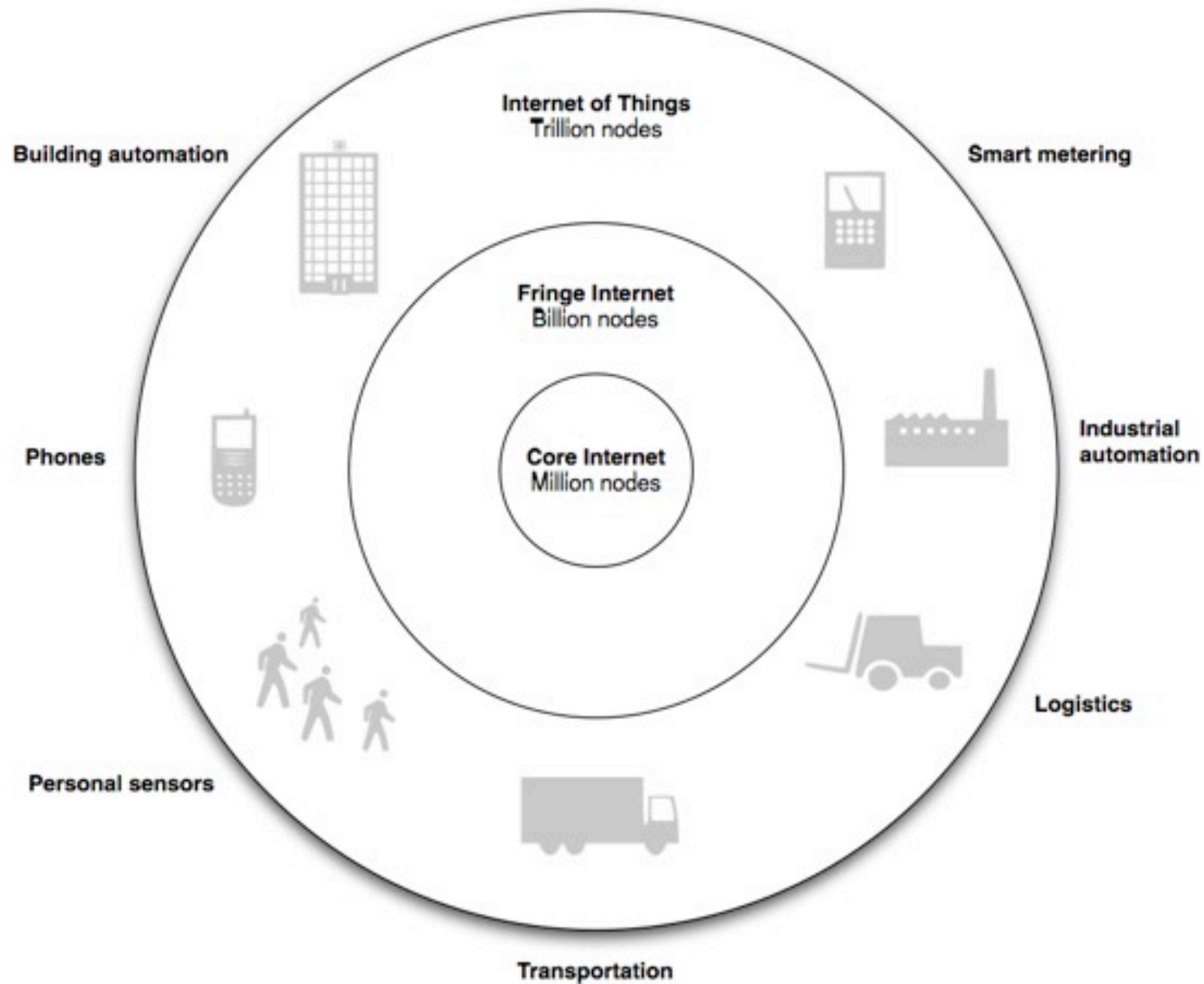


# Getting Started with IPv6 in Low-Power Wireless “Personal Area” Networks (6LoWPAN)

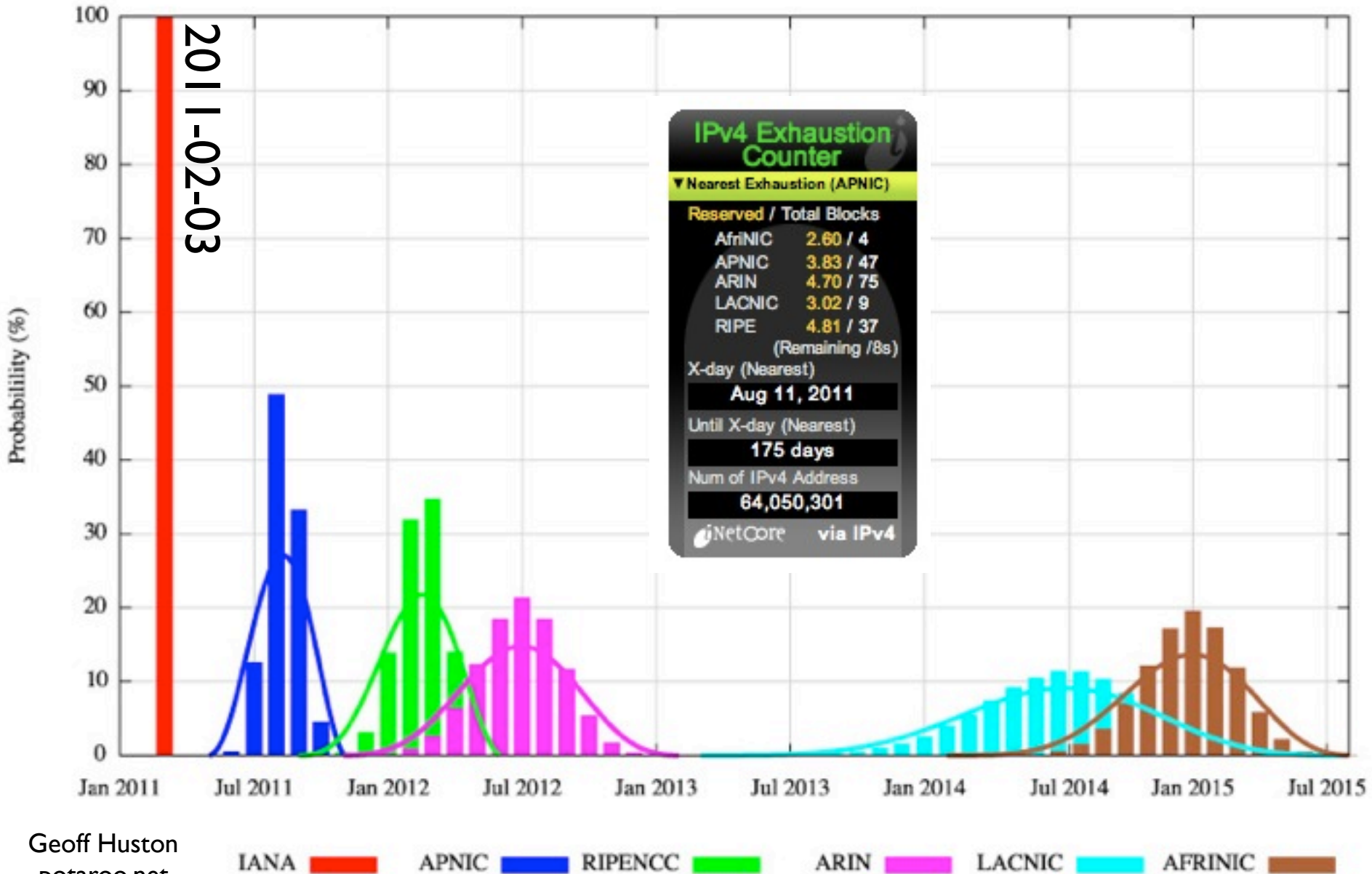
Carsten Bormann, Universität Bremen TZI  
IETF 6lowpan WG and CoRE WG Co-Chair

Presented at IAB Tutorial on *Interconnecting Smart Objects with the Internet*, Prague, Saturday, 2011-03-26

<http://www.iab.org/about/workshops/smartobjects/tutorial.html>



# IPv4 Registry Exhaustion Dates



IPv6 =  $3.4 \times 10^{38}$  addresses  
340282366920938463463374607431768211456

- ▶ There are only  $\sim 10^{25}$  grains of sand on the earth
- ▶ Let's settle for a billion ( $10^9$ ) objects on the net
- ▶ Danfoss (EU):  $0.4 \times 10^9$  thermostats so far
- ▶ Walmart (US):  $0.1 \times 10^9$  CFL light bulbs **per year**

# Wires are too expensive

- ▶ Electrical wall socket + installation = \$80
- ▶ Cat5 socket + installation = \$150
- ▶ 1 billion nodes = GDP of Kuwait

# Wireless?

Technology	Range	Speed	Power Use	Cost
WiFi	100 m	nn Mbit/s	high	\$\$\$
Bluetooth	10–100 m	n Mbit/s	medium	\$\$
802.15.4	10–100 m	0.n Mbit/s	low	\$

# Constrained node/networks

Internet of Things

IoT

Low-Power/Lossy

LLN

IP Smart Objects

IPSO

# Constrained nodes

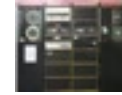
- ▶ **Node:** a few MHz, ~10 KiB RAM, ~100 KiB Flash/ROM
- ▶ Often battery operated — must sleep a lot  
( $\text{mW} \cdot (1.0 - (99.9\%)) = \mu\text{W}!$ )
  
- ▶ Moore's law will fix it?
- ▶ Moore's law will be used mostly
  - to make things cheaper,
  - more energy efficient!



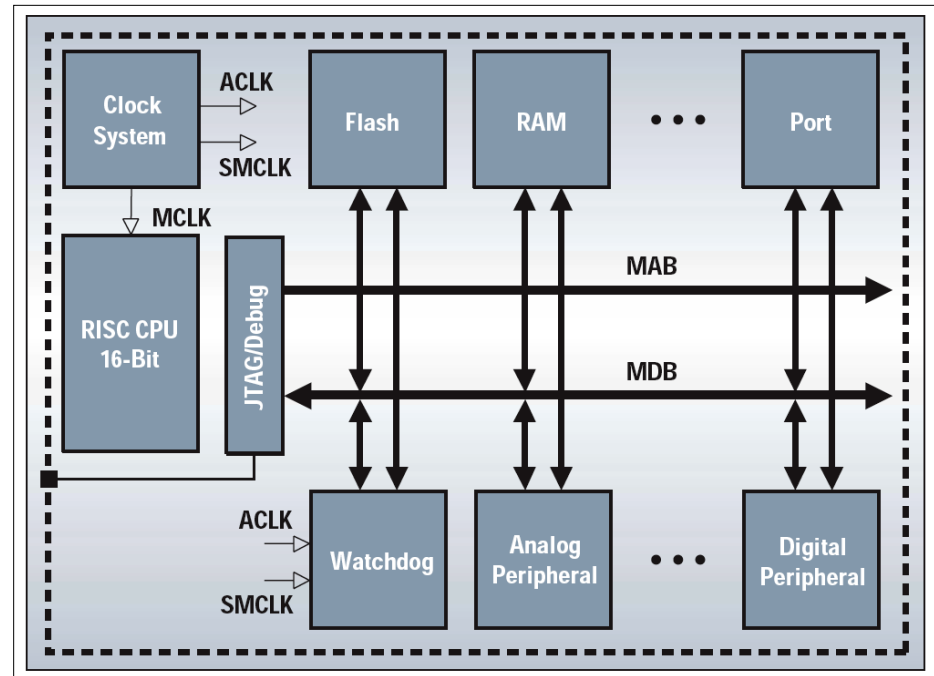


440 100 1000  
1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 13. 14. 15. 16. 17. 18. 19. 20. 21. 22. 23. 24. 25. 26. 27. 28. 29. 30. 31. 32. 33. 34. 35. 36. 37. 38. 39. 40. 41. 42. 43. 44. 45. 46. 47. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100. 101. 102. 103. 104. 105. 106. 107. 108. 109. 110. 111. 112. 113. 114. 115. 116. 117. 118. 119. 120. 121. 122. 123. 124. 125. 126. 127. 128. 129. 130. 131. 132. 133. 134. 135. 136. 137. 138. 139. 140. 141. 142. 143. 144. 145. 146. 147. 148. 149. 150. 151. 152. 153. 154. 155. 156. 157. 158. 159. 160. 161. 162. 163. 164. 165. 166. 167. 168. 169. 170. 171. 172. 173. 174. 175. 176. 177. 178. 179. 180. 181. 182. 183. 184. 185. 186. 187. 188. 189. 190. 191. 192. 193. 194. 195. 196. 197. 198. 199. 200. 201. 202. 203. 204. 205. 206. 207. 208. 209. 210. 211. 212. 213. 214. 215. 216. 217. 218. 219. 220. 221. 222. 223. 224. 225. 226. 227. 228. 229. 230. 231. 232. 233. 234. 235. 236. 237. 238. 239. 240. 241. 242. 243. 244. 245. 246. 247. 248. 249. 250. 251. 252. 253. 254. 255. 256. 257. 258. 259. 260. 261. 262. 263. 264. 265. 266. 267. 268. 269. 270. 271. 272. 273. 274. 275. 276. 277. 278. 279. 280. 281. 282. 283. 284. 285. 286. 287. 288. 289. 290. 291. 292. 293. 294. 295. 296. 297. 298. 299. 300. 301. 302. 303. 304. 305. 306. 307. 308. 309. 310. 311. 312. 313. 314. 315. 316. 317. 318. 319. 320. 321. 322. 323. 324. 325. 326. 327. 328. 329. 330. 331. 332. 333. 334. 335. 336. 337. 338. 339. 340. 341. 342. 343. 344. 345. 346. 347. 348. 349. 350. 351. 352. 353. 354. 355. 356. 357. 358. 359. 360. 361. 362. 363. 364. 365. 366. 367. 368. 369. 370. 371. 372. 373. 374. 375. 376. 377. 378. 379. 380. 381. 382. 383. 384. 385. 386. 387. 388. 389. 390. 391. 392. 393. 394. 395. 396. 397. 398. 399. 400. 401. 402. 403. 404. 405. 406. 407. 408. 409. 410. 411. 412. 413. 414. 415. 416. 417. 418. 419. 420. 421. 422. 423. 424. 425. 426. 427. 428. 429. 430. 431. 432. 433. 434. 435. 436. 437. 438. 439. 440. 441. 442. 443. 444. 445. 446. 447. 448. 449. 450. 451. 452. 453. 454. 455. 456. 457. 458. 459. 460. 461. 462. 463. 464. 465. 466. 467. 468. 469. 470. 471. 472. 473. 474. 475. 476. 477. 478. 479. 480. 481. 482. 483. 484. 485. 486. 487. 488. 489. 490. 491. 492. 493. 494. 495. 496. 497. 498. 499. 500. 501. 502. 503. 504. 505. 506. 507. 508. 509. 510. 511. 512. 513. 514. 515. 516. 517. 518. 519. 520. 521. 522. 523. 524. 525. 526. 527. 528. 529. 530. 531. 532. 533. 534. 535. 536. 537. 538. 539. 540. 541. 542. 543. 544. 545. 546. 547. 548. 549. 550. 551. 552. 553. 554. 555. 556. 557. 558. 559. 560. 561. 562. 563. 564. 565. 566. 567. 568. 569. 570. 571. 572. 573. 574. 575. 576. 577. 578. 579. 580. 581. 582. 583. 584. 585. 586. 587. 588. 589. 590. 591. 592. 593. 594. 595. 596. 597. 598. 599. 600. 601. 602. 603. 604. 605. 606. 607. 608. 609. 610. 611. 612. 613. 614. 615. 616. 617. 618. 619. 620. 621. 622. 623. 624. 625. 626. 627. 628. 629. 630. 631. 632. 633. 634. 635. 636. 637. 638. 639. 640. 641. 642. 643. 644. 645. 646. 647. 648. 649. 650. 651. 652. 653. 654. 655. 656. 657. 658. 659. 660. 661. 662. 663. 664. 665. 666. 667. 668. 669. 670. 671. 672. 673. 674. 675. 676. 677. 678. 679. 680. 681. 682. 683. 684. 685. 686. 687. 688. 689. 690. 691. 692. 693. 694. 695. 696. 697. 698. 699. 700. 701. 702. 703. 704. 705. 706. 707. 708. 709. 710. 711. 712. 713. 714. 715. 716. 717. 718. 719. 720. 721. 722. 723. 724. 725. 726. 727. 728. 729. 730. 731. 732. 733. 734. 735. 736. 737. 738. 739. 740. 741. 742. 743. 744. 745. 746. 747. 748. 749. 750. 751. 752. 753. 754. 755. 756. 757. 758. 759. 760. 761. 762. 763. 764. 765. 766. 767. 768. 769. 770. 771. 772. 773. 774. 775. 776. 777. 778. 779. 780. 781. 782. 783. 784. 785. 786. 787. 788. 789. 790. 791. 792. 793. 794. 795. 796. 797. 798. 799. 800. 801. 802. 803. 804. 805. 806. 807. 808. 809. 810. 811. 812. 813. 814. 815. 816. 817. 818. 819. 820. 821. 822. 823. 824. 825. 826. 827. 828. 829. 830. 831. 832. 833. 834. 835. 836. 837. 838. 839. 840. 841. 842. 843. 844. 845. 846. 847. 848. 849. 850. 851. 852. 853. 854. 855. 856. 857. 858. 859. 860. 861. 862. 863. 864. 865. 866. 867. 868. 869. 870. 871. 872. 873. 874. 875. 876. 877. 878. 879. 880. 881. 882. 883. 884. 885. 886. 887. 888. 889. 890. 891. 892. 893. 894. 895. 896. 897. 898. 899. 900. 901. 902. 903. 904. 905. 906. 907. 908. 909. 910. 911. 912. 913. 914. 915. 916. 917. 918. 919. 920. 921. 922. 923. 924. 925. 926. 927. 928. 929. 930. 931. 932. 933. 934. 935. 936. 937. 938. 939. 940. 941. 942. 943. 944. 945. 946. 947. 948. 949. 950. 951. 952. 953. 954. 955. 956. 957. 958. 959. 960. 961. 962. 963. 964. 965. 966. 967. 968. 969. 970. 971. 972. 973. 974. 975. 976. 977. 978. 979. 980. 981. 982. 983. 984. 985. 986. 987. 988. 989. 990. 991. 992. 993. 994. 995. 996. 997. 998. 999. 1000.

# Example: MSP430



- Texas Instruments mixed-signal uC
- 16-bit RISC
- ROM: 1-60 KiB
- RAM: Up to 10 KiB
- Analogue
  - 12 bit ADC & DAC
  - LCD driver
- Digital
  - USART x 2
  - DMA controller
  - Timers

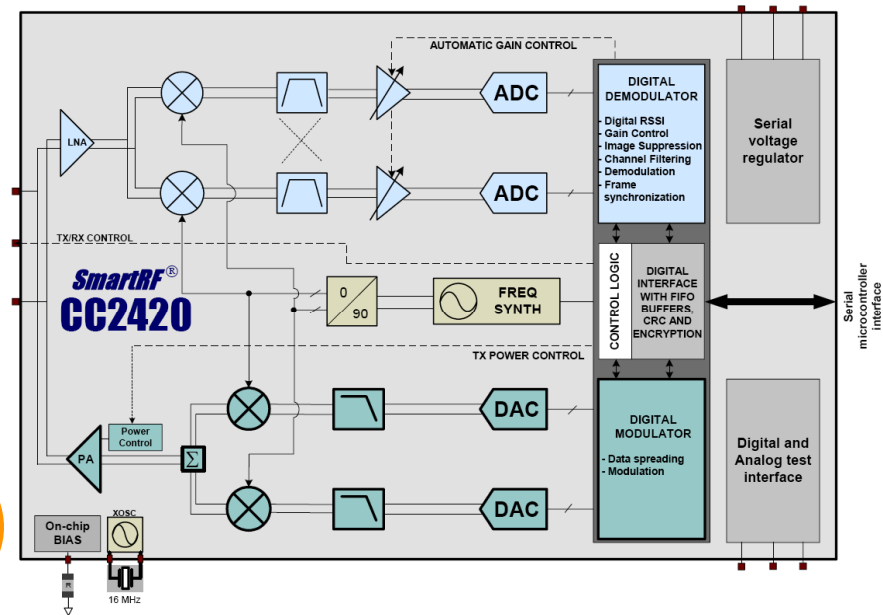


# Example: CC2420

- IEEE 802.15.4 compliant radio
- 2.4 GHz band using DSSS at 250 kbps
- Integrated voltage regulator
- Integrated digital baseband and MAC functions
  - Clear channel assessment
  - Energy detection (RSSI)
  - Synchronization
  - Framing
  - Encryption/authentication
  - Retransmission (CSMA)

cf. CC2431:  
Add an  
8051-style  
processor,  
8 KB RAM,  
128 KB Flash

| Sleep | Idle   | Tx            | Rx      |
|-------|--------|---------------|---------|
| 20 uA | 426 uA | 8.5 – 17.4 mA | 18.8 mA |



# Power Consumption

A simple approximation for power consumption:

$$P_{avg} = \frac{1}{T_F} \left\{ P_{Rx} T_{wk-up} + P_{Rx} (N_{Tx} T_{Tx-up} + N_{Rx} T_{Rx-up}) + P_{Tx} T_{Tx} + P_{Rx} T_{Rx} + P_{idle} T_{idle} + P_{sleep} T_{sleep} \right\}$$

- $T_{wk-up}$  = Time that takes to go from sleep state to awake state
- $T_{Tx-up}$  = Transmitter setup time, i.e. time it takes for the transmitter to be ready
- $T_{Tx}$  = Time in the  $Tx$  state
- $T_{Rx-up}$  = Receiver setup time, i.e. time it takes for the receiver to be ready
- $T_{Rx}$  = Time in the  $Rx$  state
- $T_{idle}$  = Time in the idle state
- $T_{sleep}$  = Time in the sleep state
- $N_{Tx}$  = Average number of times per frame that the transmitter is used
- $N_{Rx}$  = Average number of times per frame that the receiver is used
- $T_F$  = Duration of the time frame
- $P_{Tx}$  = Power used in the  $Tx$  state
- $P_{Rx}$  = Power used in the  $Rx$  state
- $P_{idle}$  = Power used in the idle state
- $P_{sleep}$  = Power used in the sleep state
- $P_{avg}$  = Power used in the sleep state
- = Average power used by the transceiver

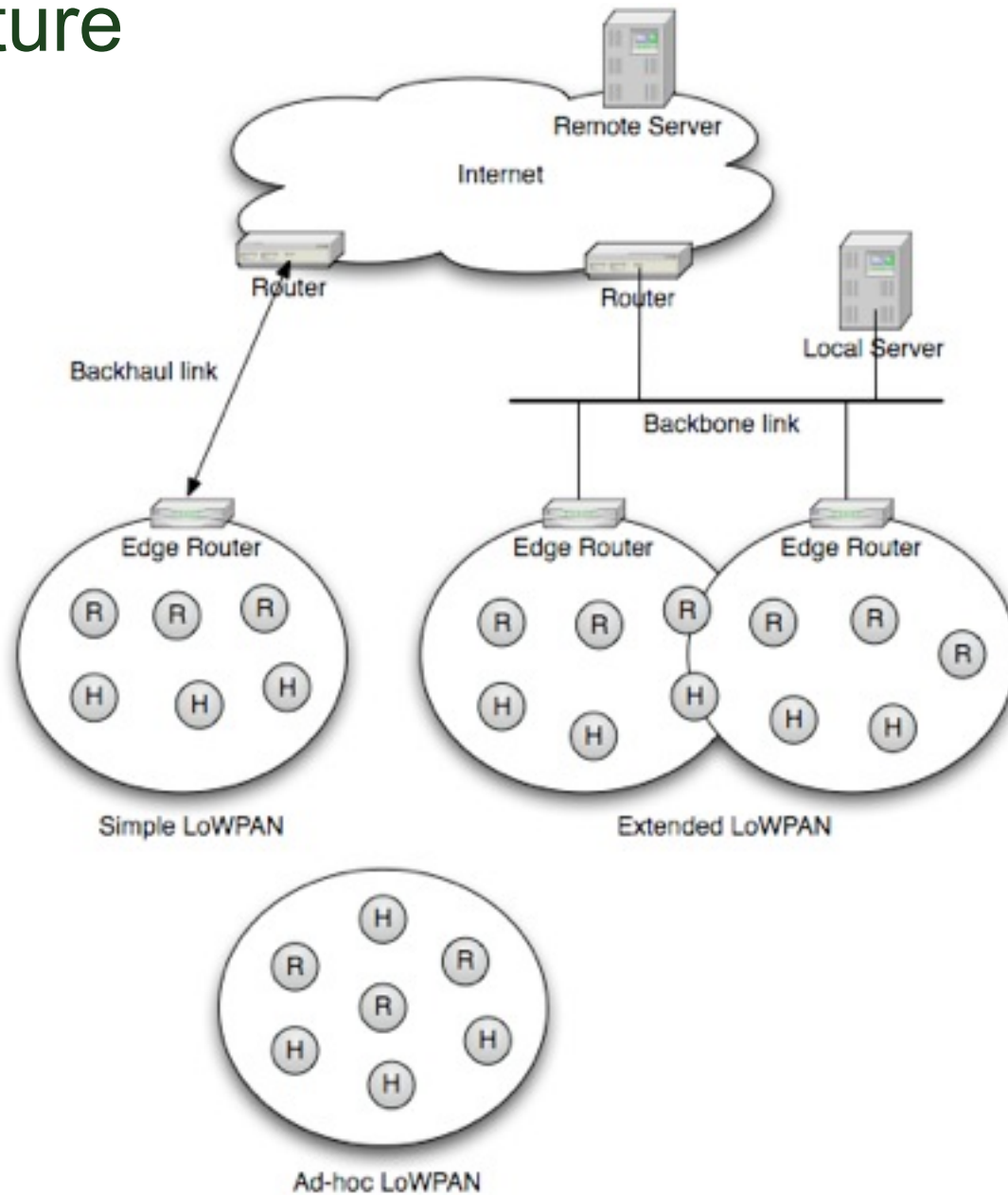
# Constrained networks

- ▶ **Node:** ... must sleep a lot ( $\mu\text{W}$ !)
  - vs. “always on”
- ▶ **Network:**  $\sim 100$  kbit/s, high loss, high link variability
- ▶ May be used in an unstable radio environment
- ▶ Physical layer packet size may be limited ( $\sim 100$  bytes)
- ▶ “LLN low power, lossy network”

# Constrained network example: IEEE 802.15.4

- ▶ popular low-power ( $\sim 1$  mW) radio
- ▶ 0.9 and 2.4 GHz bands
  - 868 MHz: Europe (1 % duty cycle, 20 kbit/s)
  - 900 MHz: US (40 kbit/s)
  - 2.4 GHz: World (256 kbit/s)
- ▶ up to 127-byte packets

# Architecture



# Constrained node/networks in the IETF

- ▶ IETF WGs to date:

| 6LoWPAN                                   | ROLL                                | CoRE   |
|---|-------------------------------------|--|
| INT area<br>(Internet)<br>L2/L3 interface | RTG area<br>(Routing)<br>L3 routing | APP area<br>(Applications)<br>L7 application |
| (now)                                     | JP Vasseur                          | Zach Shelby                                  |

- ▶ New: **LWIG** (INT area, Adam Dunkels  
Light-Weight Implementation Guidance)



# 6LoWPAN: IPv6 over Low-Power Area Networks (IEEE 802.15.4)

- ▶ IETF WG chartered in 2005 to define IPv6 over IEEE **802.15.4**
- ▶ Two initial deliverables approved 2007-05-01
  - RFC 4919: Problem statement (“Goals and Assumptions”)
  - **RFC 4944**: Format specification (“IPv6 over 802.15.4”)

# RFC 4944: make 802.15.4 look like an IPv6 link

2007

## ▶ Basic **Encapsulation**

- Efficient representation of packets < ~100 bytes
- First approach to **stateless** Header Compression

## ▶ **Fragmentation** (map 1280 byte MTU to < 128 bytes)

- Datagram tag/Datagram offset

## ▶ **Mesh forwarding**

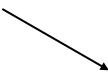
- Identify Originator/Final Destination

## ▶ Minimal use of complex MAC layer concepts

- cf. RFC 3819 “Advice for Internet Subnetwork Designers”

# The dispatch byte: replacement Ethertype

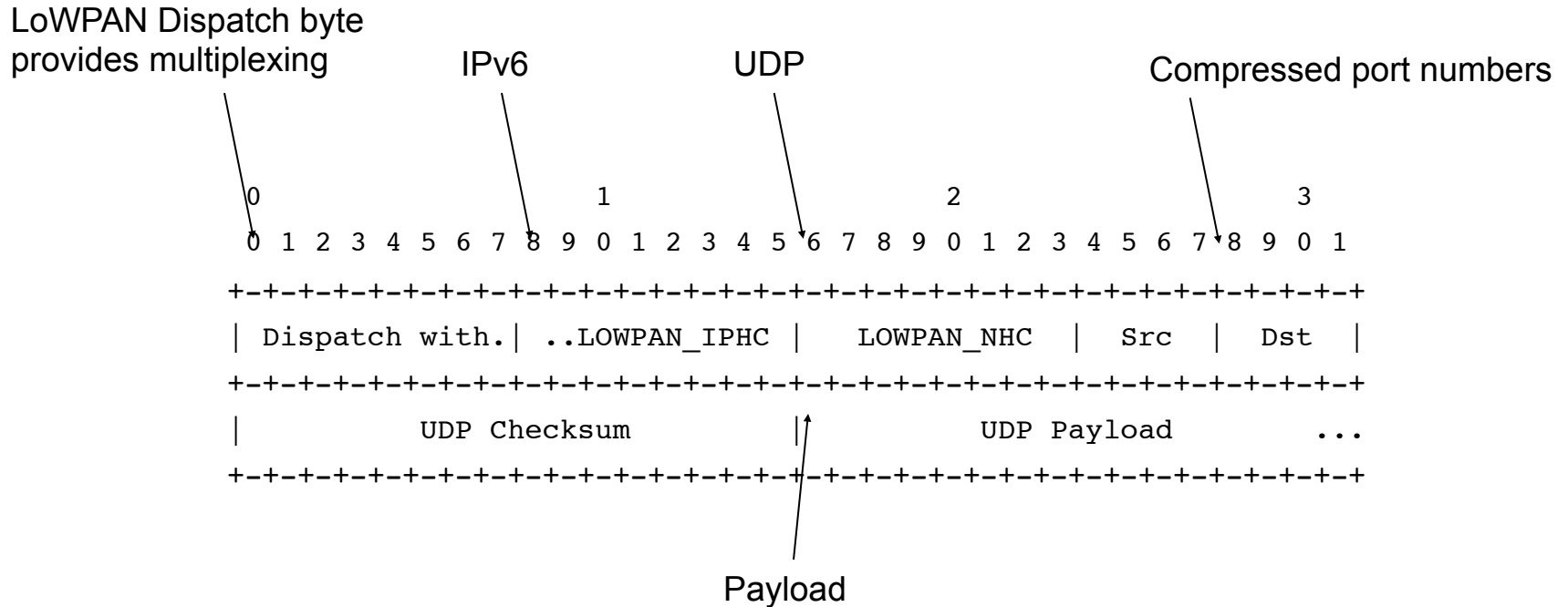
Space later taken by  
6LoWPAN-HC



| Pattern   | Header Type   |
|-----------|---|
| 00 xxxxxx | <b>NALP</b> - <b>Not a LoWPAN frame</b>                 |
| 01 000001 | <b>IPv6</b> - <b>Uncompressed IPv6 Addresses</b>        |
| 01 000010 | <b>LOWPAN_HC1</b> - <b>LOWPAN_HC1 compressed IPv6</b>   |
| 01 000011 | reserved - Reserved for future use                      |
| ...       | reserved - Reserved for future use                      |
| 01 001111 | reserved - Reserved for future use                      |
| 01 010000 | <b>LOWPAN_BC0</b> - <b>LOWPAN_BC0 broadcast</b>         |
| 01 010001 | reserved - Reserved for future use                      |
| ...       | reserved - Reserved for future use                      |
| 01 111110 | reserved - Reserved for future use                      |
| 01 111111 | <b>ESC</b> - <b>Additional Dispatch byte follows</b>    |
| 10 xxxxxx | <b>MESH</b> - <b>Mesh Header</b>                        |
| 11 000xxx | <b>FRAG1</b> - <b>Fragmentation Header (first)</b>      |
| 11 001000 | reserved - Reserved for future use                      |
| ...       | reserved - Reserved for future use                      |
| 11 011111 | reserved - Reserved for future use                      |
| 11 100xxx | <b>FRAGN</b> - <b>Fragmentation Header (subsequent)</b> |
| 11 101000 | reserved - Reserved for future use                      |
| ...       | reserved - Reserved for future use                      |
| 11 111111 | reserved - Reserved for future use                      |

# LoWPAN UDP/IPv6 Headers

6 Bytes!



using draft-ietf-6lowpan-hc format (see later)

# Fragmentation

LoWPAN Dispatch byte provides multiplexing

Initial Fragment

```
0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|1 1 0 0 0|  datagram_size  |  datagram_tag  |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

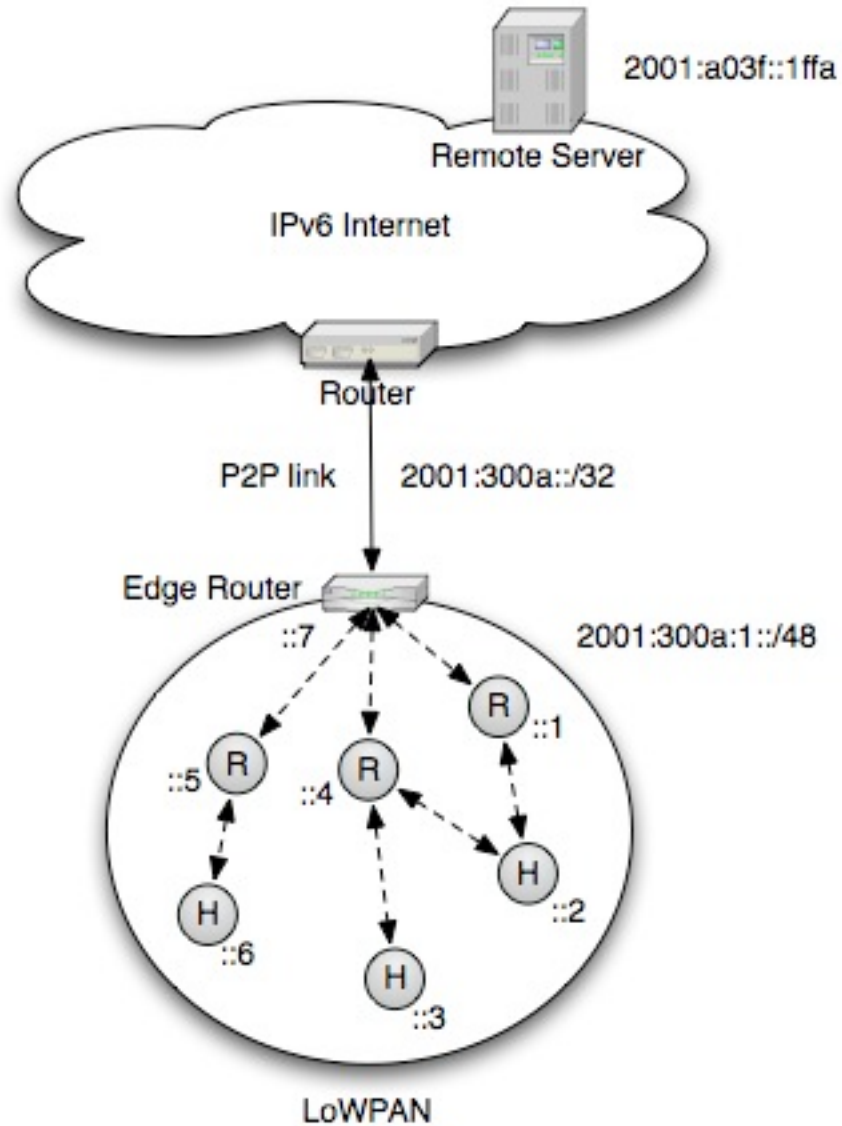
Following Fragments

```
0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|1 1 1 0 0|  datagram_size  |  datagram_tag  |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|datagram_offset|
+--+--+--+--+--+--+--+--+
```

# 6LoWPAN-ND

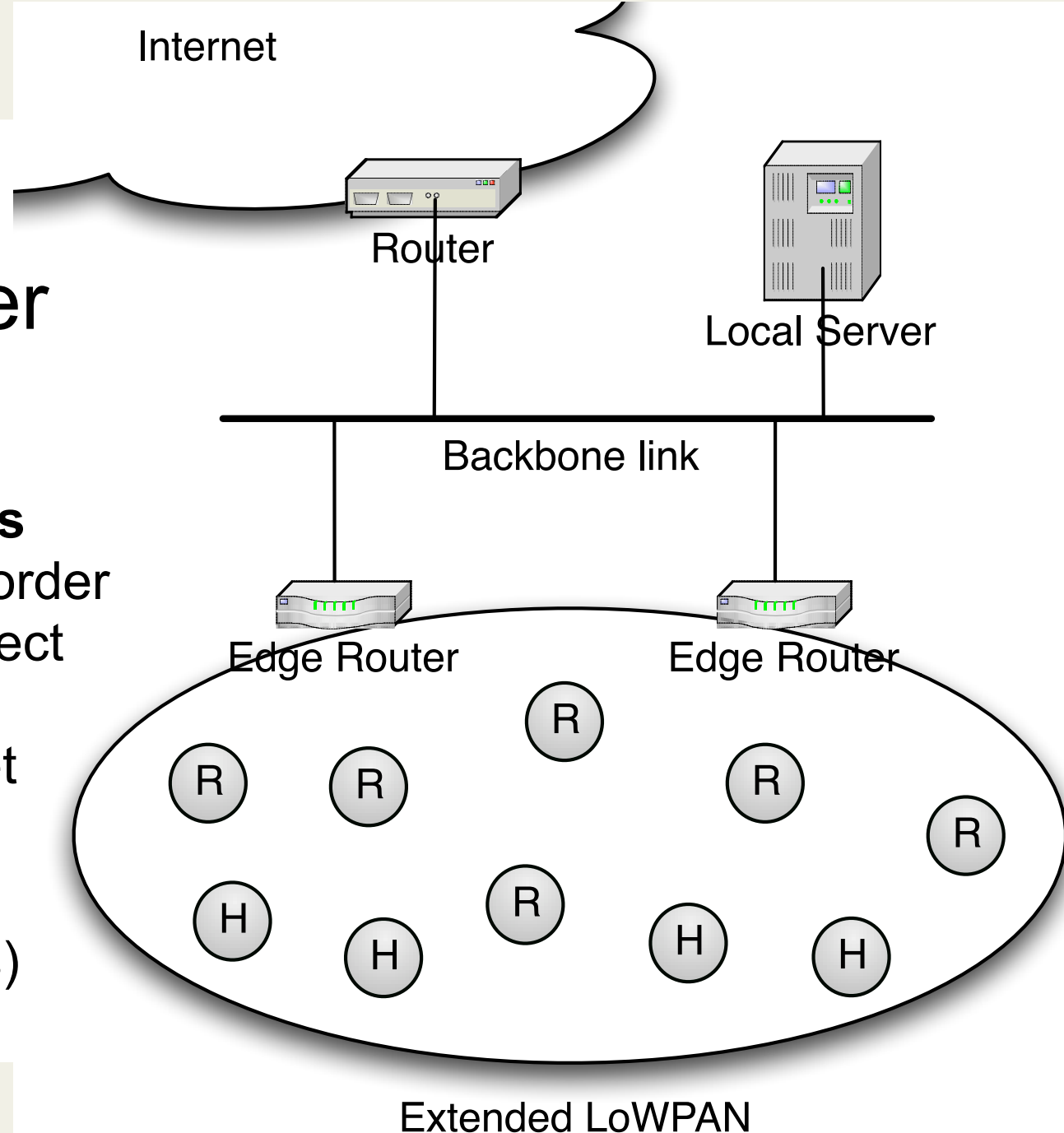
- ▶ Classic IPv6: link  $\equiv$  multicast domain
  - could be realized by **mesh-under** (L2 routing) protocol
  - can be substituted by less multicast-reliant ND
- ▶ RFC 5889: **ad-hoc link model**
  - Alternative: confine link to radio domain
  - multicast is local only
  - need **route-over** (L3 routing) protocol to build larger 6lowpan
- ▶ Both mesh-under and route-over covered by single architecture

# Addressing Example



# Route-Over

- ▶ One or more **Edge Routers** (6LoWPAN border routers) connect 6LoWPAN to global Internet
- ▶ 6LoWPAN comprises routers (6LRs) and hosts

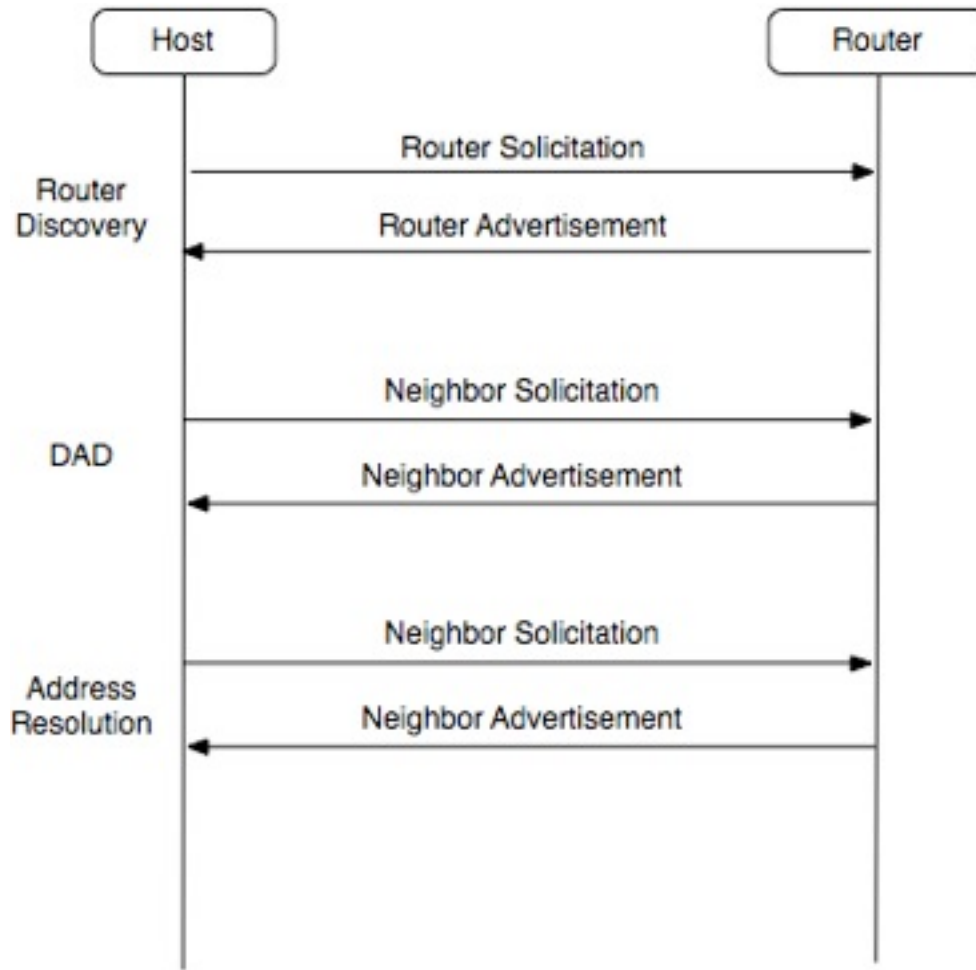




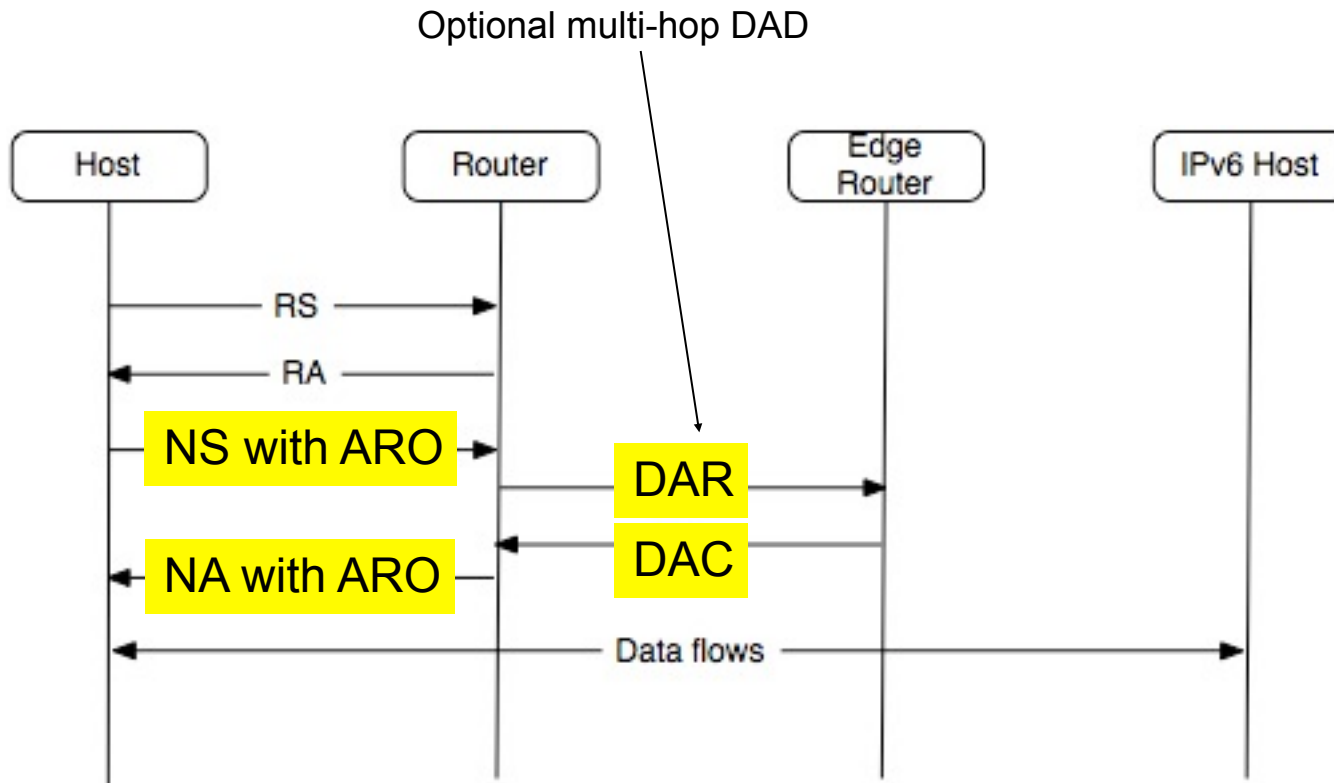
# Host-Router Interface

- ▶ Hosts only talk to routers
- ▶ Routers *may* redirect to hosts in mesh-under
  - no direct host-host communication in route-over (wouldn't be awake anyway...)

# Classic IPv6 Neighbor Discovery (4861)



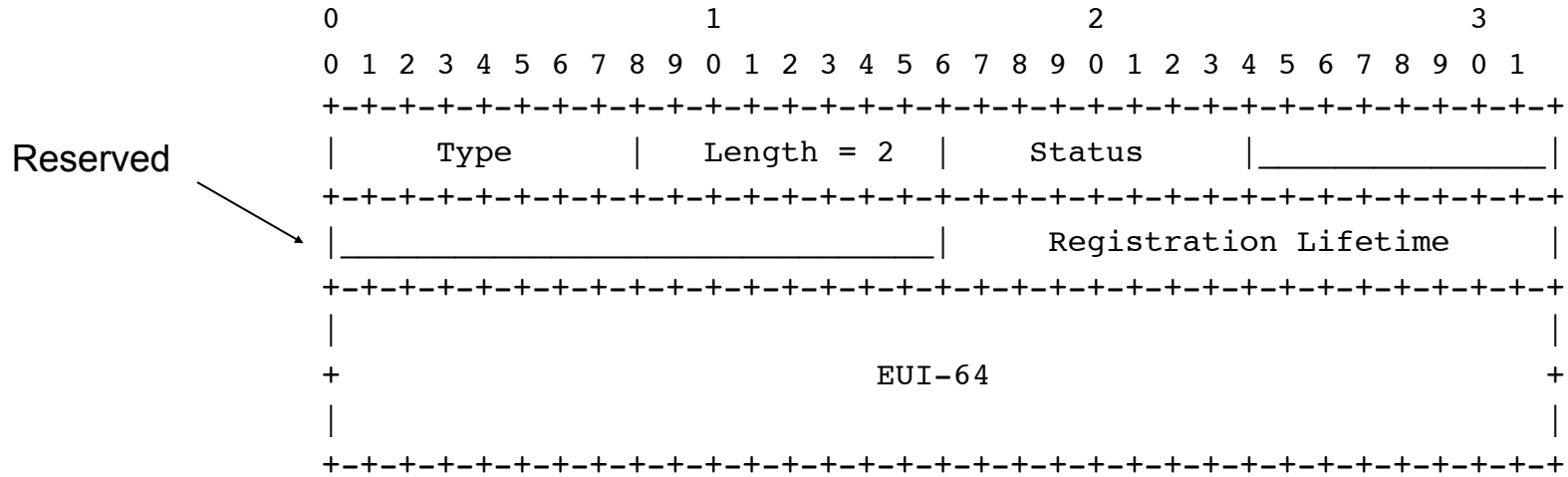
# Typical 6LoWPAN-ND Exchange



# 6LoWPAN-ND elements beyond 4861

- ▶ **ARO** (address registration option):
  - hosts register their addresses to routers (6LRs): NS/NR
  - 6LRs can check the address with edge router (6LBR):  
new ICMP messages **DAR/DAC**
  - replaces NS/NR use for address resolution (off-link model),  
but keeps NS/NR intact for NUD (neighbor unreachability  
detection)

# ARO Option



Registration Lifetime: 16-bit unsigned integer. Time in units of 60 s

# 6LoWPAN-ND elements beyond 4861

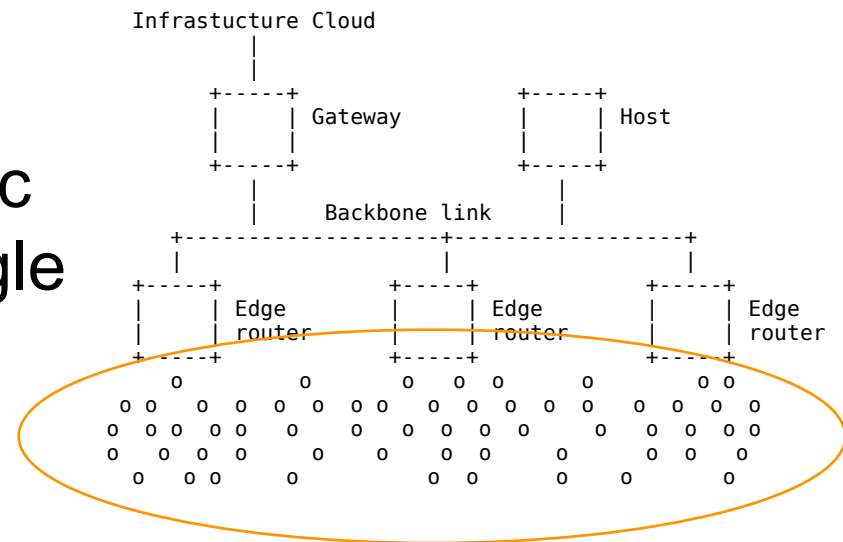
- ▶ **ABRO** (authoritative border router option)
  - distribute information about available 6LBRs (edge routers)
  
- ▶ **6CO** (6LoWPAN Context Option)
  - distribute **header compression** context in entire LoWPAN

# Header Compression

- ▶ Organized Layer Violation
  - compress L3+ headers on L3-L2 interface
- ▶ Traditional header compression (ROHC, RFC 3095 etc.) is **flow-based stateful**
  - exploit redundancies **between** packets
- ▶ RFC 4944 header compression is **stateless**
  - exploit **intra-packet** redundancies only
  - Can't compress global prefix

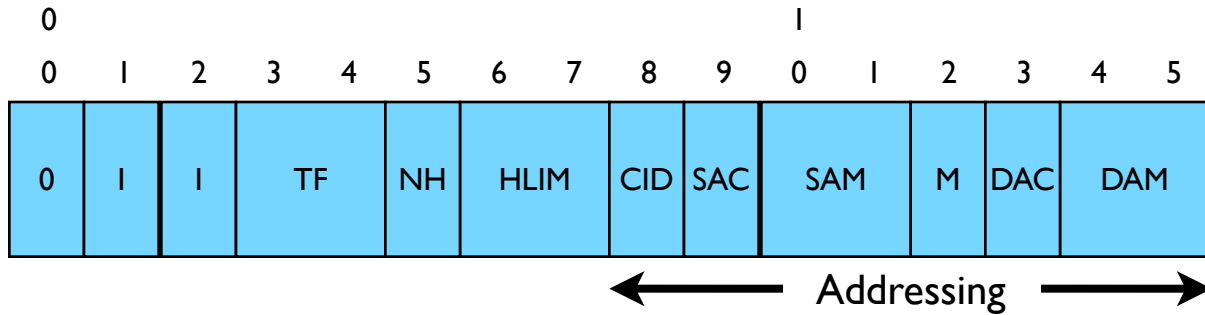
# 6LoWPAN-HC: Header Compression

- ▶ Is there a middle ground?
- ▶ draft-bormann-6lowpan-cbhc (2008-07): introduced a single **area context state** for an entire 6LoWPAN





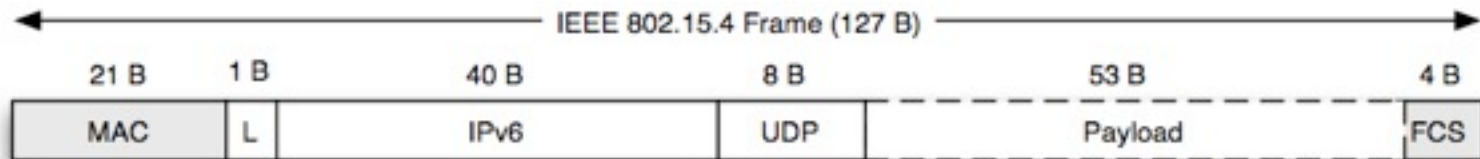
# IPv6 Header Compression



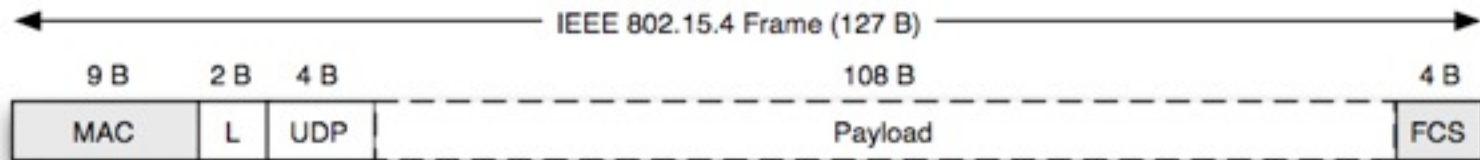
|      |        |                               |
|------|--------|-------------------------------|
| TF   | 2 bits | Traffic Class and Flow Label  |
| NH   | 1 bit  | Next Header                   |
| HLIM | 2 bits | Hop Limit                     |
| CID  | 1 bit  | Context Identifier Extension  |
| SAC  | 1 bit  | Source Address Context        |
| SAM  | 2 bits | Source Address Mode           |
| M    | 1 bit  | Multicast Address Compression |
| DAC  | 1 bit  | Destination Address Context   |
| DAM  | 2 bits | Destination Address Mode      |

# 6LoWPAN Headers

- Orthogonal header format for efficiency
- Stateless header compression



**Full UDP/IPv6 (64-bit addressing)**

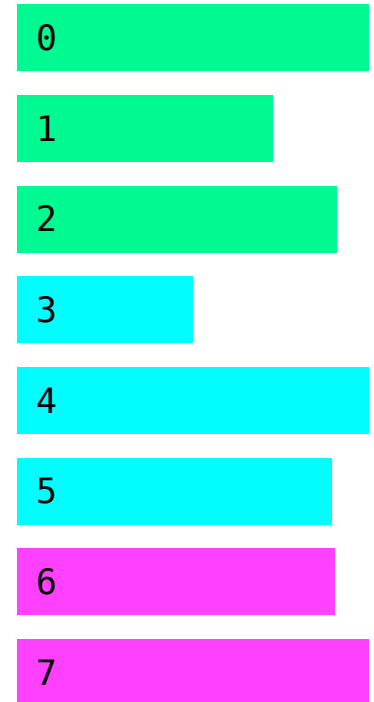


**Minimal UDP/6LoWPAN (16-bit addressing)**



# Area context state

- ▶ up to 16 contexts, each with
  - a prefix (up to 128 bits), given by value and length
- ▶ Used in **6LoWPAN-HC** compression methods:
  - carry the variable bits in packet
  - infer variable bits from L2 addresses
  - special method for multicast addresses
- ▶ **6LoWPAN-ND Context Option (6CO)**
  - distribute context throughout 6LoWPAN
  - defined as part of 6LoWPAN-ND



# LOWPAN-NHC: Next Header

Compression beyond IP header:

- ▶ UDP: Can compress ports (from 61616 to 61631)
- ▶ Extension headers:
  - IPv6 Hop-by-Hop Options Header
  - IPv6 Routing Header
  - IPv6 Fragment Header
  - IPv6 Destination Options Header
  - IPv6 Mobility Header [RFC3775]
  - IPv6 Header (for compressing Tunneling)

# New proposal: 6LoWPAN-GHC

- ▶ Generic compression of remaining headers and header-like payloads: ICMPv6, ND, RPL; DHCP; ...
- ▶ draft-bormann-6lowpan-ghc: simple LZ77 based on **bytecode**
  - **single-page** specification: simple
  - **stateless** (but can use 6LoWPAN-HC context)
- ▶ provides modest compression factors between 1.65 and 1.85 on realistic examples

| code byte | Action  | Argument            |
|-----------|---|---------------------|
| 0kkkkkk   | Append k = 0b0kkkkk bytes of data in the bytecode argument (k < 96)   | The k bytes of data |
| 0110iiii  | Append all bytes (possibly filling an incomplete byte with zero bits) from Context i  |                     |
| 0111iiii  | Append 8 bytes from Context i: i.e., the context value truncated/extended to 8 bytes, and then append 0000 00FF FE00 (i.e., 14 bytes total)                         |                     |
| 1000mm    | Append 0b0000mm-2 bytes of zeroes   |                     |
| 1001mm    | reserved  |                     |
| 101nsss   | sa += 0b0sss000, na += 0b000n000  |                     |
| 11renkkk  | n = na-0b00000mm-2; s = 0b0000kkk-sa-n; append n bytes from previously output bytes, starting s bytes to the left of the current output pointer; set sa = 0, na = 0 |                     |

# Example: ND Neighbor Solicitation

## ▶ Payload:

```

87 00 a7 68 00 00 00 00 fe 80 00 00 00 00 00
02 1c da ff fe 00 30 23 01 01 3b d3 00 00 00 00
1f 02 00 00 00 00 00 06 00 1c da ff fe 00 20 24

```

## Pseudoheader:

```

20 02 0d b8 00 00 00 00 00 00 ff fe 00 3b d3
fe 80 00 00 00 00 00 02 1c da ff fe 00 30 23
00 00 00 30 00 00 00 3a

```

copy: 04 87 00 a7 68

4 nulls: 82

ref(32): fe 80 00 00 00 00 00 02 1c da ff fe 00 30 23

-> ref 10lnssss 1 2/1lnnnkkk 6 0: b2 f0

copy: 04 01 01 3b d3

4 nulls: 82

copy: 02 1f 02

5 nulls: 83

copy: 02 06 00

ref(24): 1c da ff fe 00 -> ref 10lnssss 0 2/1lnnnkkk 3 3: a2 db

copy: 02 20 24

## Compressed:

```

04 87 00 a7 68 82 b2 f0 04 01 01 3b d3 82 02 1f
02 83 02 06 00 a2 db 02 20 24

```

Was 48 bytes; compressed to 26 bytes, **compression factor 1.85**

# 6LoWPAN status

- ▶ 6LoWPAN widely accepted as the way to run IP on 802.15.4
  - RFC 4944 published September 2007
  - Adoption in TinyOS, Contiki; Standards: ISA100, ZigBee SE 2.0, ...
- ▶ Recent interoperability events for both 6LoWPAN-ND/  
6LoWPAN-HC: **10+ implementations each**
- ▶ 6LoWPAN-HC (HC-15) in IESG (last step before RFC)
  - remaining issue: checksum omission (vs. draft-ietf-6man-udpzero-02)
- ▶ 6LoWPAN-ND (ND-15) went through second WG last call
  - remaining technical discussion next week before submitting to IESG



6LoWPAN =

RFC4944

– HC1/HC2

+ 6LoWPAN-HC

+ 6LoWPAN-ND