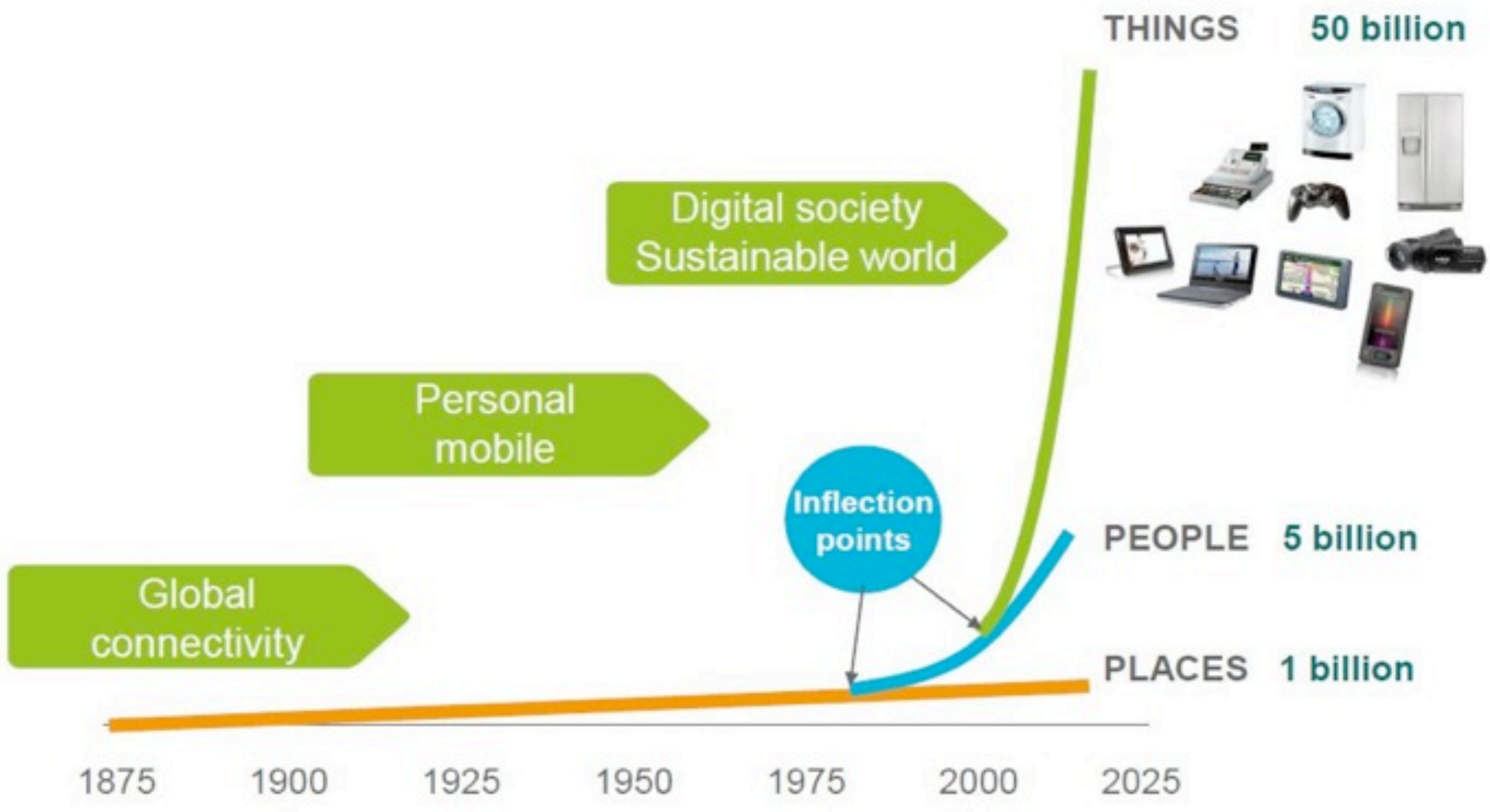
Constrained Node Networks

2014-03-05

Prof. Dr.-Ing. Carsten Bormann

TZI – Universität Bremen

Connecting:
Places → People → Things



Source: Ericsson

Scale up:

Number of nodes

(50 billion by 2020)

Scale down:

node

Scale down:

cost

complexity

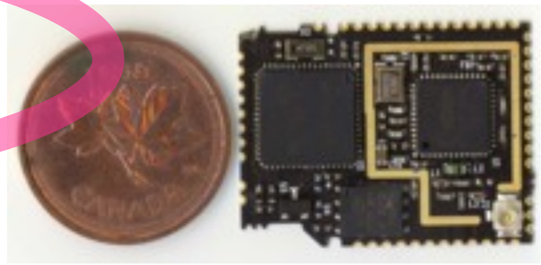
cent
kilobyte
megahertz

Constrained nodes: orders of magnitude

10/100 vs. 50/250

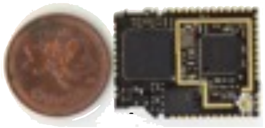
- There is not just a single class of “constrained node”

- Class 0: too small to securely run on the Internet
  - “too constrained”
- Class 1: ~10 KiB data, ~100 KiB code
  - “quite constrained”, “10/100”
- Class 2: ~50 KiB data, ~250 KiB code
  - “not so constrained”, “50/250”



- These classes are not clear-cut, but may structure the discussion and help avoid talking at cross-purposes

http://www.flickr.com/photos/blahflowers/3878202215/sizes/l/



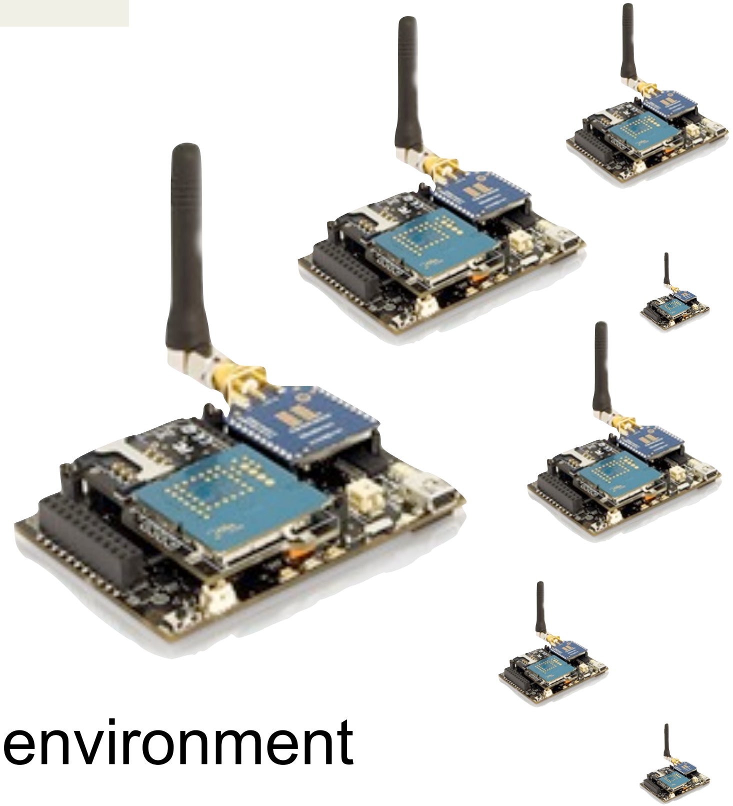
http://www.flickr.com/photos/blahflowers/3878202215/sizes/l/

in constrained node/networks, **Moore's law barely applies**

- In the low-power, low-cost area, gains from Moore's law are used
  - to save **power**
  - to save **cost**
- Performance, ROM, RAM grow **very** slowly

Constrained **networks**

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



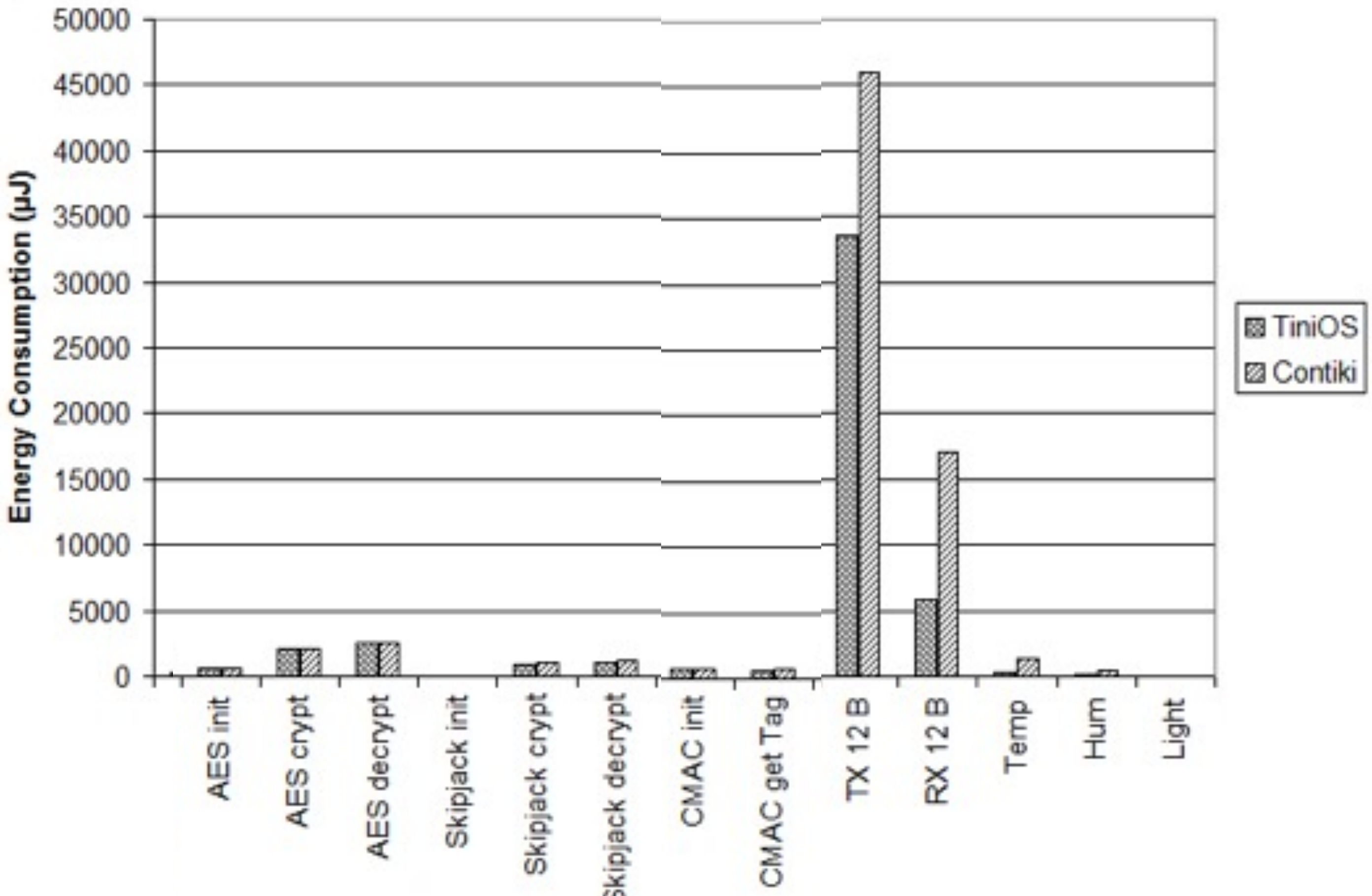
802.15.4 „ZigBee“
Bluetooth Smart
Z-Wave (G.9959)
DECT ULE

please re-calibrate your **complexity** meters

- **code** is expensive
  - “class 1” = 100 KiB, “class 2” = 250 KiB
- **state** is expensive
  - “class 1” = 10 KiB, “class 2” = 50 KiB
- **packets** are expensive
- **listening** is even more expensive
  - and multicast doesn’t work

Energy consumption on TelosB

Message exchange cost orders of magnitude more than symmetric crypto



C.B. Margi, B.T. de Oliveira, G.T. de Sousa, M.A. Simplicio Jr, P.S.L.M. Barreto, T.C.M.B. Carvalho, M. Näslund, R. Gold, ICCCN'2010 / IEEE WiMAN 2010]

Constrained Node Networks

Internet of Things	IoT
Wireless Embedded Internet	WEI
Low-Power/Lossy Networks	LLN
IP Smart Objects	IPSO

Constrained Node Network Cluster

INT	LWIG	Guidance
INT	6Lo	IP-over-foo
INT	6TiSCH	IP over TSCH
RTG	ROLL	Routing (RPL)
APP	CoRE	REST (CoAP)
SEC	DICE	Improving DTLS
OPS	________	________________

(2) The Application

CoAP

Constrained Node/Networks
➔ Compressed HTTP?

- Saves some bytes
- Retains all the complexity
  - lots of historical baggage
  - still needs TCP below
- Adds the CPU requirements for compression
- Limited gain
  - compression only takes you so far

"Make things
as simple as possible,
but not simpler.

Attributed to Albert Einstein

The **Co**nstrained **A**pplication **P**rotocol

CoAP

- implements HTTP's **REST** model
  - GET, PUT, DELETE, POST; media type model
- while avoiding most of the complexities of HTTP

- **Simple** protocol, datagram only (UDP, DTLS)
- 4-byte header, compact yet simple options encoding

- adds "observe", a lean notification architecture

CoAP Examples

- **GET** coap://temp1.25b006.floor1.example.com/temperature
  - ASCII string: `22.5`
  - could use JSON, e.g. as in draft-jennings-senml
- **PUT** coap://blue-lights.bu036.floor1.example.com/intensity
  - ASCII string: `70 %`
- **GET** coap://25b006.floor1.example.com/.well-known/core
  - `</temp>;n="TemperatureC",</light>;ct=41;n="LightLux"`
  - see RFC 6690 (CoRE link format)

More in draft-vanderstok-core-bc-05
see also draft-ietf-core-interfaces

Example Interchange

Option

Payload

C: CON + GET coap://server/resource

```
 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 | 0 |    0    |  GET = 0.01   |            MID=1234           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| +3 =3 |    6    |         "server" (6 Bytes) ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| +8=11 |    8    |        "resource" (8 Bytes) ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

S: ACK, ct=application/cbor, payload: {“hlo”:“World”}

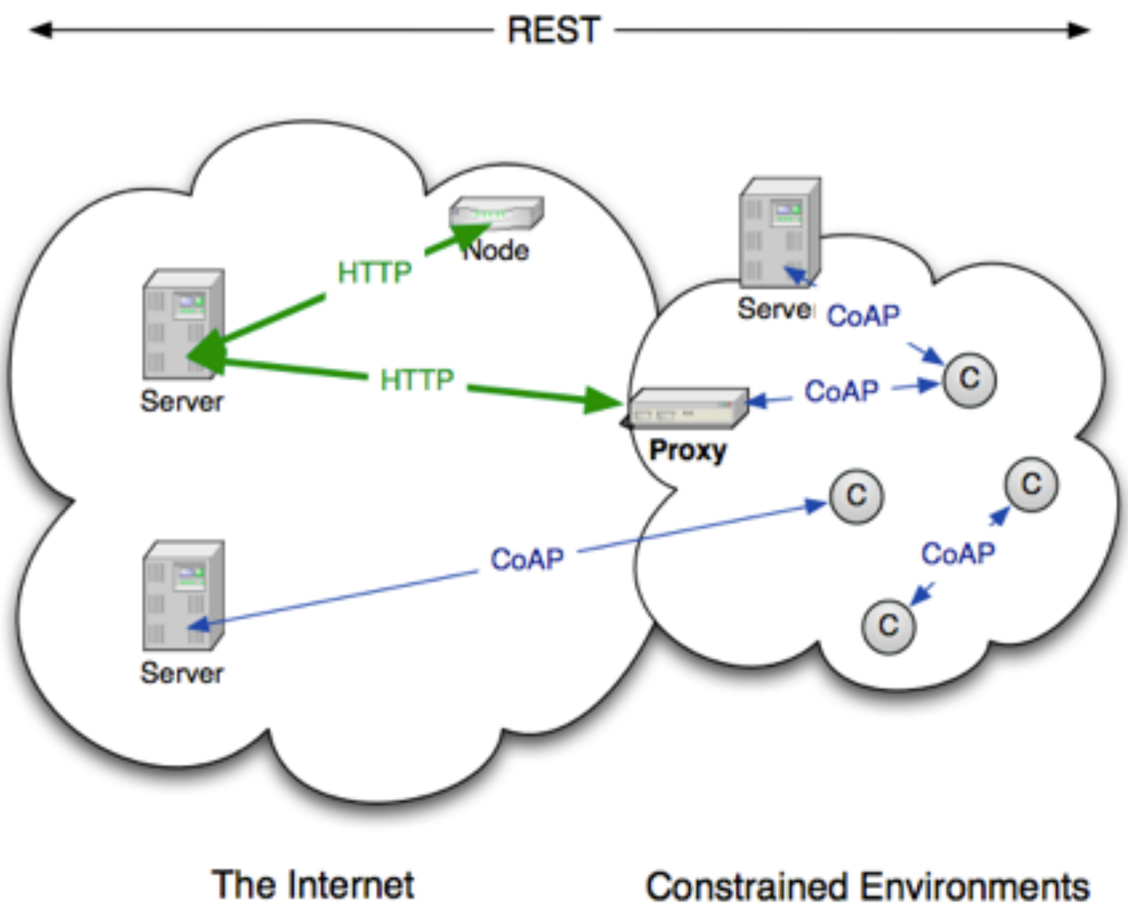
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 1 | 2 |    0    |Content = 2.05 |            MID=1234           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|+12=12 |    1    |      60       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|1|1|1|1|1|1|1| A1 63 h l o 65 W o r l d  (11 Bytes) ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Content-Format = 60 (application/cbor)

Payload Marker

Combining CoAP and HTTP

- CoAP is used in constrained environment
- CoAP and HTTP share proxy model based on REST
- Enables standard, application-independent proxy



TZi

Security is not optional!

- HTTP can use TLS (“SSL”)
- CoAP: Use **DTLS** 1.2
  - Add 6LoWPAN-**GHC** for efficiency
- Crypto: Move to **ECC**
  - **P-256** curve
  - **SHA-256**
  - **AES-128**
- To do:
  - Commissioning models (Mother/Duckling, Mothership, …)
  - **Authorization format and workflow**
  - Performance fixes (DICE)

128-bit security
(~ RSA 3072-bit)

CoAP

DTLS

SOLACE: Smart Object Lifecycle Architecture

- Processes for **usably secure** lifecycle (changes of ownership, authorization, privacy, …)

```
 _Manufactured           _SW update          _Decommissioned
/                       /                   /
|   _Installed          |   _ Application   |   _Removed &
|  /                    |  / reconfigured   |  /  replaced
|  |   _Commissioned    |  |                |  |
|  |  /                 |  |                |  |   _Reownership &
|  |  |    _Application |  |   _Application |  |  / recommissioned
|  |  |   /   running   |  |  / running     |  |  |
|  |  |   |             |  |  |             |  |  |             \\
+##+##+###+#############+##+##+#############+##+##+##############>>>
    \/  \______________/ \/  \_____________/ \___/         time //
    /           /         \          \          \
Bootstrapping  /      Maintenance &   \     Maintenance &
              /      re-bootstrapping  \   re-bootstrapping
        Operational                Operational
```

The lifecycle of a thing in the Internet of Things

[draft-garcia-core-security]