CoAP: Scaling the Web to billions of nodes

Prof. Dr.-Ing. Carsten Bormann, Universität Bremen TZI
IETF 6LoWPAN WG and CoRE WG Co-Chair

IoT International Forum Launch, 2011-11-24 Berlin
Things have always been “on the Internet”

- What we are trying now is a massive attempt at scaling:
  - Scaling down the (total) cost of a connected node
  - Scaling up the total number of connected nodes
For which applications did the Internet first scale massively?

- Remote Login
- E-Mail
- NetNews
- The Web
The elements of success of the Web

- **HTML**
  - uniform *representation* of documents
  - (now moving forward to HTML5 with CSS, JavaScript)
- **URIs**
  - uniform *referents* to data and services on the Web
- **HTTP**
  - universal *transfer protocol*
  - enables a distribution system of proxies and reverse proxies
Translating this to M2M

- **HTML**
  - uniform *representation* of documents
  - (now moving forward to HTML5 with CSS, JavaScript)

- **URIs**
  - uniform *referents* to data and services on the Web

- **HTTP**
  - universal *transfer protocol*
  - enables a distribution system of proxies and reverse proxies
Constrained node/networks

Internet of Things  IoT
Low-Power/Lossy      LLN
IP Smart Objects     IPSO
Internet-Connected Objects  ICO
Constrained nodes

- **Node**: a few MHz, ~10 KiB RAM, ~100 KiB Flash/ROM
- Often battery operated — must sleep a lot
  \( (\text{mW} \times (1.0 - (99.9 \%))) = \mu\text{W}) \)

- Moore’s law will fix it?
- Moore’s law will be used mostly
  - to make things cheaper,
  - more energy efficient!
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
Constrained networks

- **Node**: ... must sleep a lot (\(\mu W\!\!\!)\)
  - 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 \((\sim 100\) bytes)\)

- “LLN low power, lossy network”
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 Constrained Application Protocol (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-07.txt

- **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 draft-ietf-core-link-format-09.txt

More in draft-vanderstok-core-bc-05.txt
Example Interchange

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

S: ACK, ct=text/plain, payload: Hello World
Combining CoAP and HTTP

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

CoAP Server → Proxy → HTTP Client

CON GET /light

ACK max-age=30s 2.05 Content "<light>..."

HTTP GET /light

200 OK "<light>..."

cache /light

cache valid

HTTP GET /light

200 OK "<light>..."
Constrained RESTful Environments (CoRE) Working Group

Drafts:

• Constrained Application Protocol  draft-ietf-core-coap
• Observing Resources in CoAP  draft-ietf-core-observe
• Blockwise Transfers in CoAP  draft-ietf-core-block
• CoRE Link Format  draft-ietf-core-link-format
Self-Describing Nodes: Discovery via CoRE Link Format

REQ: GET /.well-known/core

RES: 2.05 Content
</sensors>;ct=40;rt="index";rt="Sensor Index",
</sensors/temp>;rt="TemperatureC";if="sensor",
</sensors/light>;ct=41;rt="LightLux";if="sensor",
<http://www.example.com/sensors/t123>;anchor="/sensors/temp"
;rel="describedby",
</t>;anchor="/sensors/temp";rel="alternate"
CoAP: Industry uptake

- Tens of implementations show up to interop events
  - including a number of open-source projects: libcoap (generic C and TinyOS), Erbium (Contiki), Copper (GUI)
  - ETSI will hold a formal interop on March 24/25, 2012, in Paris
- ZigBee/HomePlug Smart Energy Profile 2.0 is using HTTP and considering adding CoAP
- CoAP is integral part of ETSI “M2M” TS 102 921
The Web of Things

M2M Applications

Asset Management

Facility Management

Security Monitoring

Energy Management

IP Smart Objects

Work together

Developers

CoRE
Web Linking
JOSE
TLS
SenML
HOMENET
LWIG

IETF

ETSI

OASIS

W3C

Broadband Forum

oma

BACnet™

ZigBee™