CoRE
Internet-Draft
Intended status: Informational
Expires: April 28, 2011

P. van der Stok
Philips Research
K. Lynn
Consultant

October 25, 2010

CoAP Utilization for Building Control
draft-vanderstok-core-bc-02

Abstract

This draft describes an example use of the RESTful CoAP protocol for building automation and control (BAC) applications such as HVAC and lighting. A few basic design assumptions are stated first, then URI structure is utilized to define group as well as unicast scope for RESTful operations. RFC 3986 defines the URI components as (1) a scheme, (2) an authority, used here to locate the building, area, or node under control, (3) a path, used here to locate the resource under control, and (4) a query part (fragments are not supported in CoAP.) Next, it is shown that DNS can be used to locate URIs on the scale necessary in large commercial BAC deployments. Finally, a method is proposed for mapping URIs onto legacy BAC resources, e.g., to facilitate application-layer gateways.

This proposal supports the view that (1) building control is likely to move in steps toward all-IP control networks based on the legacy efforts provided by DALI, LON, BACnet, ZigBee, and other standards, and (2) service discovery is complimentary to resource discovery and facilitates control network scaling.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 28, 2011.

Table of Contents

1. Introduction

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

In addition, the following conventions are used in this document.

The term "service" often means different things to different communities and even different things to the same community. In building control protocol standards, service is often used to refer to a function in the RPC sense. In this context, we generally substitute the term "function". In the IETF community, service may often refer to an abstract capability such as "datagram delivery". In this submission we use the term service, in the sense defined by "DNS-based Service Discovery" [I-D.cheshire-dnsext-dns-sd], as equivalent to a CoAP end-point.

A CoAP end-point is identified by the authority part of a URI. We refer to this end-point (which is resolved to an {IP address, port} tuple) as a "node". By "device" we generally mean the physical object handled by the installer. While a device may host more than one service, for simplicity we assume here that a given device may only host a single CoAP node.

In examples below involving URIs, the authority is preceded by double slashes "//" and path is preceded by a single slash "/". The examples may make use of fully qualified or partial domain names and the difference should be clear from the context.

1.2. Motivation

The CoAP protocol [I-D.ietf-core-coap] aims at providing a user application protocol architecture that is targeted to a network of nodes with a low resource provision such as memory, CPU capacity, and energy. In general, IT application manufacturers strive to provide the highest possible functionality and quality for a given price. In contrast, the building controls market is highly price sensitive and manufacturers tend to compete by delivering a given functionality and quality for the lowest price. In the first market a decreasing memory price leads to more software functionality, while in the second market it leads to a lower Bill of Material (BOM).

The vast majority of nodes in a typical building control application are resource constrained, making the standardization of a lightweight

application protocol like CoAP a necessary requirement for IP to penetrate the device market. This approach is further indicated by the low energy consumption requirement of battery-less nodes. Low resource budget implies low throughput and small packet size as for [IEEE.802.15.4]. Reduction of the packet size is obtained by using the header reduction of 6LoWPAN [RFC4944] and encouraging small payloads.

Several legacy building control standards (e.g [BACnet], [DALI], [KNX], [LON], [ZigBee], etc.) have been developed based on years of accumulated knowledge and industry cooperation. These standards generally specify a data model, functional interfaces, packet formats, and sometimes the physical medium for data objects and function invocation. Many of these industry standards also specify lower-level functionality such as proprietary transport protocols, necessitating expensive stateful gateways for these standards to interoperate. Many more recent building control network include IP-based standards for transport (at least to interconnect islands of functionality) and other functions such as naming and discovery. CoAP will be successful in the building control market to the extent that it can represent a given standard's data objects and provide functions, e.g. resource discovery, that these standards depend on.

From the above the basic syntax assumptions can be summarized as:

- Generate small payloads.

- Compatible with legacy standards (e.g LON, BACnet, DALI, ZigBee Device Objects).

- Service/resource discovery in agreement with legacy standards and naming conventions.

This submission aims at an approach in which the payload contains messages with a syntax defined by legacy control standards. Accordingly, the syntax of the service/resource discovery messages is related to the chosen legacy control standard. The intention is a progressive approach to all-IP in building control. In a first stage standard IETF based protocols (e.g CoAP, DNS-SD) are used for transport of control messages and discovery messages expressed in a legacy syntax. This approach enables the reuse of controllers based on the semantics of the chosen control standard. In a later stage a complete redesign of the controllers can be envisaged guided by the accumulated experience with all-IP control.

Two concepts, hierarchy and group, are of prime importance in building control, particularly in lighting and HVAC. Many control messages or events are multicast from one device to a group of

devices (e.g. from a light switch to all lights in a room). The scope of a multicast command or discovery message determines the group of nodes that is targeted. A group scope may be defined as link-local, or as a tree maintained by IP-multicast or an overlay that corresponds to the logical structure of a building or campus, and is independent of the underlying network structure. Techniques for group communication are discussed in [I-D.rahman-core-groupcomm].

As described in "Commercial Building Applications Requirements" [I-D.martocci-6lowapp-building-applications] it is typical practice to aggregate building control at the room, area, and supervisory levels. Furthermore, networks for different subsystems (lights, HVAC, etc.) or based on different legacy standards have historically been isolated from each other in so-called "silos". RESTful web services [Fielding] represent one possible way to expose functionality and normalize data representations between silos in order to facilitate higher order applications such as campus-wide energy management.

Consequently, additional protocol oriented assumptions are:

- Nodes may be addressed by more than one group.

- Resources addressed by a group must be uniformly named across all targeted nodes.

For clarity, this I-D limits itself to two types of applications: (1) M2M control applications running within a building area without any human intervention after commissioning of a given network segment and (2) maintenance oriented applications where data are collected from node in several building areas by nodes inside or outside the building, and humans may intervene to change control settings.

2. URI structure

This I-D considers three elements of the URI: scheme, authority, and path, as defined in "Uniform Resource Identifier (URI): Generic Syntax" [RFC3986]. The authority is defined within the context of standard DNS host naming, while the path is valid in relation to a fully qualified domain name (FQDN) plus optional port (and protocol is implicit, based on scheme). An example based on RFC 3986 is: foo://host.example.com:8042/over/there?name=ferret#nose, where "foo" is the scheme, "host.example.com:8042" is the authority, "/over/there" is the path, "name=ferret" is the query, and "nose" is the fragment. Fragments are not supported in CoAP.

2.1. Scheme part

The default scheme specified in this submission is "coap". We assume syntactic compatibility with the "http" scheme specification [RFC2616], namely that the host part of the authority may be represented either as a literal IP address or as a fully qualified domain name. While scheme is irrelevant from the perspective of the service, it is used in service discovery to identify the protocol used to access the service.

TBD: we have yet to fully explore the utility of a separate scheme (e.g., "coapm") to support group communication models as described in [I-D.rahman-core-groupcomm].

2.2. Authority part

The authority part is either a literal IP address or a DNS name comprised of a local part, specifying an individual node or group of nodes, and a global part specifying the (sub)domain that may reflect the logical hierarchical structure of the building control network. The result is said to be a fully qualified domain name (FQDN) which is globally unique down to the group or node level. An optional port number may be included in the authority following a single colon ":" if the service port is other than the default CoAP value.

The CoAP spec [I-D.ietf-core-coap] states "When a CoAP server is hosted by a 6LoWPAN node, it SHOULD support a port in the 61616-61631 compressed UDP port space defined in [RFC4944]. The specific port number in use will be communicated in a URI and/or by some other discovery mechanism." As shown below, DNS-SD [I-D.cheshire-dnsext-dns-sd] is a viable technique for discovering dynamic host and port assignments for a given service. However, the use of dynamic ports in URIs is likely to lead to brittle (non-persistent) identifiers as it is conventional to treat different ports as representing different authorities and there is no assurance that a coap server will consistently acquire the same dynamic port.

A building can be unambiguously addressed by it GPS coordinates or more functionally by its zip or postal code. For example the Dutch Internet provider, KPN, assigns to each subscriber a host name based on its postcode. Analogously, an example authority for a building may be given by: //bldg.zipcode-localnr.Country/ or more concretely an imaginary address in the Netherlands as: //bldg.5533BA-125a.nl/. The "bldg" prefix can specify the target node within the building. Arriving at the node identified by //bldg.5533BA-125a.nl, the receiving service can parse the path portion of the URI and perform the requested method on the specified resource.

Buildings have a logical internal structure dependent on their size and function. This ranges from a single hall without any structure to a complex building with wings, floors, offices and possibly a structure within individual rooms. The naming of the building control equipment and the actual control strategy are intimately linked to the building structure. It is therefore natural to name the equipment based on their location within the building. Consequently, the local part of the URI identifying a piece of equipment is expressed in the building structure. An example is: //light-27.floor-1.west-wing...

This proposal assumes a basic level of cooperation between the IT and building management infrastructure, namely the ability of the former to delegate DNS subdomains to the latter. This allows the building controls installer to implement an appropriate naming scheme with the required granularity. For institutional real estate such as a college or corporate campus, the authority might be based on the organization's domain, e.g. //node-or-group.floor.wing.bldg.campus.example.com/. In cases where subdomain delegation is not an option, structure can still be represented in a "flat" namespace, subject to the 63 octet limit for a DNS sub-string: //group1-floor2-west-bldg3-campus.example.com.

Most communication is device to device (M2M) within the building. Often a device needs to communicate to all devices of a given type within a given area of the building. For example a thermostat may access all radiator actuators in a zone. A light switch located at room 25b006 of floor one, expressed as: //switch0.25b006.floor1.5533BA-125a.nl/, might specify a command to light1 within the same room with //light1.25b006.floor1.5533BA-125a.nl/. This approach seems to lead to rather verbose URI strings in the packet, contrary to the small packet assumption. However, the design of CoAP is such that the authority portion of the URI need not be transmitted in requests sent to origin servers. The question arises as to whether the syntax of the authority part needs to be standardized for building control. Given the examples later in the text, this appears more to be the concern of the building owner or the installer.

2.3. Path part

Every network addressable resource is completely identified by a URI scheme://authority/path. The path part of the URI specifies the resource within a given node. The representation of object types and their associated attributes are typically subjects for standardization. There is no widely accepted standard for uniformly naming building control device structure in a URI. A vigorous effort is undertaken by the oBIX working group of OASIS [oBIX], but its

current impact is limited.

When a GET method with an URI like "//t-sensor1.25b006.floor1.example.com/temperature" is sent, it represents an a priori understanding that the node with name t-sensor1 exists, is of a given standard type (e.g BACnet temperature sensor), and that this standard type has the readable attribute: temperature. When commands are sent to a group of nodes it MUST be the case that the targeted resource has the same path on all targeted nodes. Therefore, it is necessary to establish at least a local uniform path naming convention to achieve this. One approach is to include the name of the standard, e.g BACnet, as the first element in the path and then employ the standard's chosen data scheme (in the case of BACnet, /bacnet/device/object/property). Perhaps a better alternative is to build on the concepts presented in [I-D.ietf-core-link-format] and identify resources of a given type in terms of the "/.well-known/core" prefix.

3. Group Naming and Addressing

Given a network configuration and associated prefixes, the network operator needs to define an appropriate set of groups which can be mapped to the building areas. Knowledge about the hierarchical structure of the building areas may assist in defining a network architecture which encourages an efficient group communication implementation. IP-multicasting over the group is a possible approach for building control, although proxy-based methods may prove to be more appropriate in some deployments [I-D.rahman-core-groupcomm].

Example groups become:

URI authority	Targeted group
//all.bldg6...	"all nodes in building 6"
//all.west.bldg6...	"all nodes in west wing, building 6"
//all.floor1.west.bldg6...	"all nodes on floor 1, west wing, ..."
//all.bu036.floor1.west.bldg6...	"all nodes in office bu036, ..."

The granularity of this example is for illustration rather than a recommendation. Experience will dictate the appropriate hierarchy for a given structure as well as the appropriate number of groups per subdomain. Note that in this example, the group name "all" is used to identify the group of all nodes in each subdomain. In practice, "all" could name an address record in each of the DNS zones shown above and would bind to a different multicast address [RFC3596] in each zone. Highly granular multicast scopes are only possible using

IPv6. The multicast address allocation strategy is beyond the scope of this I-D, but various alternatives have been proposed [RFC3306][RFC3307][RFC3956]. Some techniques in this proposal, e.g. service discovery as described below, can be accomplished with a single coap-specific multicast address as long as the desired scope is building-wide.

To illustrate the concept of multiple group names within a building, consider the definition, as done with [DALI], of scenes within the context of a floor or a single office. For example, the setting of all blue lights in office bu036 of floor 1 can be realized by multicasting a message to the group "//blue-lights.bu036.floor1". Each group is associated with an IP address. Consequently, when the application specifies the sending of an "on" message to all blue lights in the office, the message is multicast to the associated IP address. The uri-authority option [I-D.ietf-core-coap] need not be sent as part of the message. However to identify the resource that is addressed, a short version of the resource path can be inserted as an option as explained in [I-D.ietf-core-link-format].

The binding of a group FQDN to multicast address (i.e., creation of the AAAA record in the DNS zone server) happens during the commissioning process. (TBD: How do we associate this name with MLD's notion of a group?) Resolution of the group name to a multicast address happens at restart of a source or receiver node. A multicast address and associated group name in this context are assumed to be long-lived. It can happen that during operation the membership of the group changes (less or more lights) but its address is not altered and neither its name. In the limit, the group can degrade to a single controller that represents a non-networked subsystem replacing the original networked group of nodes. Group membership may be managed by a protocol such as Multicast Listener Discovery [RFC5790].

A group defines a set of nodes. All resources on a given node are referenced by the multicast address(es) to which the node belongs. A given node might belong to a number of groups. For example the node belonging to the "blue-lights" group in a given corridor might also belong to the groups: "whole building", "given wing", "given floor", "given corridor", and "lights in given corridor". Assuming that belonging to a group has as only consequence for the group member that it should accept packets for an additional IP address, the granularity of the domain names may have an impact on the complexity of the DNS server but not necessarily on the low-resource destinations or sources. Assuming that resolution of addresses only happens at node start-up, the complexity of the DNS server need not affect the responsiveness of the nodes.

In summary, the authority portion of the URI is used to identify a node (group) and the resulting DNS name is bound to a unicast (multicast) address. Naming is building or organization dependent, must be flexible, and does not require standardization efforts but SHOULD conform to some uniform convention.

4. Discovery

4.1. DNS-Based Service Discovery

DNS-Based Service Discovery (DNS-SD) defines a conventional way to configure DNS PTR, SRV, and TXT records to facilitate discovery of services such as CoAP nodes within a subdomain, using the existing DNS infrastrucure. This section gives a cursory overview of DNS-SD; see [I-D.cheshire-dnsext-dns-sd] for a detailed description.

A DNS-SD service is specified by a name of the form Instance.ServiceType.Domain, where the service type for CoAP nodes is "_coap._udp". The identifier "_udp" is required by the SRV record definition [RFC2782] and "_coap" identifies the protocol on top of udp. For each CoAP end-point in the zone, a PTR record with the name _coap._udp is defined and each of these refers to SRV and TXT records having the Instance.ServiceType.Domain name.

DNS-SD also supports one level of subtype, which could be used to locate coap services based on object model, for example: _bacnet._sub._coap._udp, _dali._sub._coap._udp, or _zigbee._sub._coap._udp. The maximum length of the type and subtype fields is 14 octets, therefore this could be extended to type-function as _dali-light, _dali-switch, etc.

The Domain part of the service name is identical to the DNS (sub)domain part of the authority in URIs that identify the resources on this node or group and may identify a building zone as in the examples above.

The Instance part of the service name is defined as part of the commissioning process. It must be unique within the (sub)domain. The complete service name uniquely identifies both a SRV and TXT record in the DNS zone. The granularity of a service name MAY be that of a host or group, or it could represent a particular resource within a coap node. The SRV record contains the host (AAAA record) name and port of the service. In the case where a service name identifies a particular resource, the path part of the URI must be placed in the TXT record.

4.2. Service vs Resource Discovery

While service discovery is concerned with finding the IP address, port, and protocol of a named service, resource discovery is a fine-grained discovery of resource URIs within a web service. [I-D.ietf-core-link-format] specifies a resource discovery pattern, such that sending a confirmable GET message for the /.well-known/core resource returns a set of links that identify all resources present on this node that are exposed as functions.

Assuming the ability to multicast the GET over the local link, the coap resource discovery can be used to populate the DNS-SD database in a semi-automated fashion. CoAP resource descriptions can be imported into DNS-SD for exposure to service discovery by using the n= attribute as the basis for a unique "Instance" name, defaulting to "_coap._udp" for the ServiceType, and using some means to establish which domain the service should be registered in (TBD). The DNS TXT record can be populated by importing the other resource description attributes as they share the same key=value format specified in Section 6 of [I-D.cheshire-dnsext-dns-sd].

4.3. Browsing for Services

CoAP nodes in a given subdomain may be enumerated by sending a DNS query to the authoritative server for that zone for PTR records named _coap._udp. A list of names for SRV records matching that ServiceType.Domain is returned. Each SRV record contains the port and host name of a CoAP node. The IP address of the node is obtained by resolving the host name. DNS-SD also specifies an optional TXT record, having the same name as the SRV record, which can contain "key=value" attributes. This can be used to store information about the device, e.g., schema=DALI, type=switch. The format of the TXT record can be standardized by the various control standards bodies as they adopt CoAP.

TO DO: How to handle changes in building control network configuration.

5. Legacy Structure in CoAP

In the text above it is shown how information to locate services and devices can be stored in a DNS zone registry. An installation tool can populate the registry with the resource information gleaned by the coap GET query to /.well-known/core. Applications can then query the registry to find the address, port, and path for targeted services/resources. Given the returned information, an application that acts on devices of a given legacy standard can invoke the legacy

service using coap methods. Assume a short URI-reference /dl and the setting of a value of 200 in the DALI device, dt is the number of the dali type stored in the TXT record, and ct=52 is the proposed Internet media type.

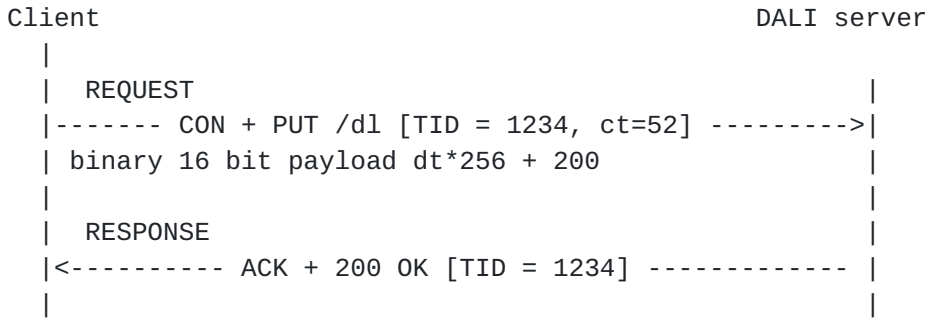

Figure 1: Sending a DALI setting with coap to DALI device

In the example the format of the payload is determined by the legacy standard. The short URI /dl on this IP address is obtained from the TXT record for this service, e.g., sh="/dl". The value dt is entered (e.g. dt="200") as the number identifying the dali type of the dali compatible resource.

6. Conclusions

This I-D explains how naming in building control is based on a hierarchical structure of the building areas. It is shown that DNS subdomain delegation and naming can be used to express this hierarchy in the authority portion of the URI, down to the group or node level. The hierarchical naming scheme need not be standardized, but rather can be designed to suit the application. However, it is recommended that the scheme be employed consistently throughout the delegated subdomain(s).

The authority portion of the URI is resolved by the client, using conventional DNS, into the unicast or multicast IP address of the targeted node(s). Taking advantage of the CoAP design [I-D.ietf-core-coap], the uri-authority option need not be transmitted in requests to origin servers and thus there is no performance penalty for using descriptive naming schemes. The coap design allows sending a short url to distinguish between resources on a given node, resulting in very compact identifiers.

DNS-SD [I-D.cheshire-dnsext-dns-sd] can be used to scale up service

discovery beyond the local link. DNS-SD can be used to enumerate instances of a given service type within a given sub-domain. This affords additional flexibility, such as the ability to discover dynamic port assignments for coap node, locate coap nodes by subtype, or bind service names for particular coap URIs.

A targeted resource is specified by the path portion of the URI. Again, this I-D does not mandate a universal naming standard for resources but uses examples to show how resources could be named for various legacy standards. An obvious requirement for resources that are accessed by multicast is that they MUST all share the same path, including short uri if used. It is shown that it is possible to transport legacy commands (e.g. expressed in BACnet, LON, DALI, ZigBee, etc.) inside a CoAP message body. This necessitates the definition of an additional IANA mime code, and the mapping of legacy specific discovery semantics to CoAP resource discovery messages or DNS-SD lookups.

7. Security considerations

TBD: The detailed CoAP security analysis needs to encompass scenarios for building control applications.

Based on the programming model presented in this I-D, security scenarios for building control need to be stated. Appropriate methods to counteract the proposed threats may be based on the work done elsewhere, for example in the ZigBee over IP context.

Multicast messages are, by their nature, transmitted via UDP. Any privacy applied to such messages must be block oriented and based on group keys shared by all targeted nodes. The CoRE security analysis must be broadened to include multicast scenarios.

8. IANA considerations

This I-D proposes the following additions to the Media type identifiers in conformance with the proposals done in [I-D.ietf-core-coap].

Internet media type	Code
/application/legacy	52

9. Acknowledgements

This I-D has benefited from conversations with and comments from Andrew Tokmakoff, Emmanuel Frimout, Jamie Mc Cormack, Oscar Garcia, Dee Denteneer, Joop Talstra, Zach Shelby, Jerald Martocci, Matthieu Vial, Jerome Hamel, and Nicolas Riou.

10. Changelog

- Removed all references to multicast and multicast scope, given draft of rahman group communication.

- Adapted examples to coap-2 and core-link drafts.

- transport short URL for destination recognition.

- Elaborated legacy discovery under DNS-SD.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[RFC3306] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6 Multicast Addresses", RFC 3306, August 2002.

[RFC3307] Haberman, B., "Allocation Guidelines for IPv6 Multicast Addresses", RFC 3307, August 2002.

[RFC3596] Thomson, S., Huitema, C., Ksinant, V., and M. Souissi, "DNS Extensions to Support IP Version 6", RFC 3596, October 2003.

[RFC3956] Savola, P. and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", RFC 3956, November 2004.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[RFC5785] Nottingham, M. and E. Hammer-Lahav, "Defining Well-Known Uniform Resource Identifiers (URIs)", RFC 5785, April 2010.

[RFC5790] Liu, H., Cao, W., and H. Asaeda, "Lightweight Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Version 2 (MLDv2) Protocols", RFC 5790, February 2010.

11.2. Informative References

[I-D.cheshire-dnsext-dns-sd]
Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", draft-cheshire-dnsext-dns-sd-06 (work in progress), March 2010.

[I-D.cheshire-dnsext-multicastdns]
Cheshire, S. and M. Krochmal, "Multicast DNS", draft-cheshire-dnsext-multicastdns-11 (work in progress), March 2010.

[I-D.ietf-core-coap]
Shelby, Z., Frank, B., and D. Sturek, "Constrained Application Protocol (CoAP)", draft-ietf-core-coap-02 (work in progress), September 2010.

[I-D.ietf-core-link-format]
Shelby, Z., "CoRE Link Format", draft-ietf-core-link-format-00 (work in progress), October 2010.

[I-D.martocci-6lowapp-building-applications]
Martocci, J., Schoofs, A., and P. Stok, "Commercial Building Applications Requirements", draft-martocci-6lowapp-building-applications-01 (work in progress), July 2010.

[I-D.rahman-core-groupcomm]
Rahman, A., "Group Communication for CoAP", draft-rahman-core-groupcomm-00 (work in progress),

October 2010.

[BACnet] Bender, J. and M. Newman, "BACnet/IP", Web http://www.bacnet.org/Tutorial/BACnetIP/index.html.

[ZigBee] Tolle, G., "A UDP/IP Adaptation of the ZigBee Application Protocol", draft-tolle-cap-00 (work in progress), October 2008.

[LON] "LONTalk protocol specification, version 3", 1994.

[DALI] "DALI Manual", Web http://www.dali-ag.org/c/manual_gb.pdf, 2001.

[KNX] Kastner, W., Neugschwandtner, G., and M. Koegler, "AN OPEN APPROACH TO EIB/KNX SOFTWARE DEVELOPMENT", Web http://www.auto.tuwien.ac.at/~gneugsch/ fet05-openapproach-preprint.pdf, 2005.

[IEEE.802.15.4] "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)", IEEE Std 802.15.4-2006, June 2006, <http://standards.ieee.org/getieee802/802.15.html>.

[oBIX] "oBIX working group", Web http://www.obix.org, 2003.

[Fielding] Fielding, R., "Architectural Styles and the Design of Network-based Software Architectures, Second Edition", Doctoral dissertation , University of California, Irvine , Web http://www.ics.uci.edu/~fielding/pubs/dissertation/top.html, 2000.

Authors' Addresses

Peter van der Stok
Philips Research
High Tech Campus
Eindhoven, 5656 AA
The Netherlands

Email: peter.van.der.stok@philips.com

Kerry Lynn
Consultant

Phone: +1 978 460 4253
Email: kerlyn@ieee.org